Doing data

Management and documentation using EpiData, REDCap and statistical software

Bøggild, Henrik

DOING DATA

MANAGEMENT AND DOCUMENTATION USING EPIDATA, REDCAP AND STATISTICAL SOFTWARE

Henrik Bøggild

Public Health and Epidemiology Group
Department of Health Science and Technology
Aalborg University

Doing data: Management and documentation using EpiData, REDCap and statistical software

Henrik Bøggild, Aalborg University, Aalborg, Denmark

ISBN: 978-87-971732-4-X

Edition 0.9: September 9, 2021. This version still has minor faults and spelling errors.

Website for latest version of the book and material: http://homes.hst.aau.dk/boggild/

1 Contents

2 Preface

The ideas laid out in this book grew over several years of teaching the principles of Data Management and Documentation (DMD) at the Master of Public Health and at the PhD-program at Aalborg University.

Much inspiration came from the book "Take good care of your data", [1] in which Svend Juul introduced the idea of the audit trail. I had also focus on other aspect of data management part, though, and other inspiration came from Jens Lauritsen from the EpiData Foundation, who share my interest in these matters. With the arrival of a new flavour of the EpiData program and the introduction of REDCap at the University, I decided to write up my teaching in this text.

At the same time, focus has grown on keeping up with new legislation in relation to the safe handling of data presupposed by the EU General Data Protection Regulation (GDPR).

I came across the quote below in another relevant book on data management[2]. The quote has been nailed to my wall above the computer for guidance over the years, but it deserves re-quoting, as it depicts the whole idea of this book. Sadly, the text is still relevant, 30 years later.

> *Coding and data entry in particular are the Cinderellas of survey method, attracting little academic interest or concern compared with sampling, interviewing and tests of significance. Yet a survey, like the proverbial chain, is probably as good as its weakest link. And if enough care, thought and time are not devoted to these aspects of the study the validity and usefulness of the whole operation are jeopardised. (. . .) we have no magical alternatives to the painstaking and methodical attention to detail which are needed for this part of the study. We can merely record the way we did it and the checks we built into our system. To do it well you need to be obsessional.*
>
> (Cartwright & Seale, 1990, p. 70)[3]

I plan to update the text regularly. Suggestions, comments and information on detected errors are most welcome; write me at boggild[at]hst.aau.dk

[1] Juul, S. Take good care of your data. Århus: School of Public Health, Department of Epidemiology, Aarhus University, 2011

[2] Bennett, S., Myatt, M., Jolley, D., & Radalowicz, A. Data Management for Surveys and Trials. The EpiData Association., 2001

[3] Cartwright, A., & Seale, C. The natural history of a survey: an account of the methodological issues encountered in a study of life before death. London: King Edward's Hospital Fund, 1990

2.1 The layout of the book

I've divided the text into 4 sections, the first describing the general principles and ideas of DMD, the second is showing how EpiData can be used to follow the principles in practical data handling, the third is written to likewise introduce the DMD principles in relation to the use of REDCap and the fourth section is describing how DMD principles is also to be used when analysing data in statistical programs.

The first part can be read without using the programs, but to better understand the content of remaining three, it would be relevant to have access to the corresponding programs. In the first section I end some of the chapters with a few recommendations that sum up the chapter.

At the same time I find that learning DMD is much easier when you have data to work on. Therefore, the text in part II-IV has been written for the readers to be able to work through examples in the text using their own data, or if you prefer, I have also included pdf-files, datasets and sample EpiData-, STATA-, SAS- and SPSS-program files for the examples, so that you can use them for self-studying.

The questionnaire and sample files are available at the website http://homes.hst.aau.dk/boggild/.

2.2 EpiData

The EpiData programs used in section II can be downloaded for free from www.epidata.dk – you will need both EpiData Manager and EpiData Entry Client (currently version 4.6.0.6) for use with this text. Specific versions of the programs have been developed to run on computers with Windows, Mac, and Linux operating systems; figures in this text are grabbed from the Windows version, but they have been verified also for a Linux-version and many of my students have been using the Mac-version.

2.3 REDCap

Research Electronic Data Capture (REDCap), used in Part III, is a secure web application for building and managing surveys and databases and it allows the user to create and design project instruments (for instance questionnaires or keying forms) of various kinds. REDCap documents access to individual files and records in the dataset, which is in accordance with GDPR. REDCap is developed at Vanderbilt University and is utilized by thousands of members of the REDCap Consortium.

To use REDCap, your organization will have to be a member of the consortium, which is free for non-profit organizations, but the set-up has some infrastructure requirements (PHP web server, MySQL databases server etc.) as well as dedicated IT-support and administration, so this is only relevant in for instance hospital– or university settings and you will not be able to have access without your organization being a member of the Consortium. See https://www.project-redcap.org/. The administrator of REDCap can install different applications, so your possibilities may differ from those at Aalborg University.

REDCap includes a Mobile App, so that data can be collected using IPad, IPhone or Android tablets.

The DMD principles also apply when using REDCap, and in section III I show how DMD works in this application —if you have access to a REDCap installation.

2.4 Statistical programs

After collecting, preparing and documenting data, you will need access to a statistical program for conducting your analyses, preparing tables and figures, and most often this also involves further data management. EpiData comes with a small EpiData Analysis program, and ”R” is another free program. In many research environments SPSS, STATA, and SAS are used but they are not free. All of them can be worked upon using written programming (called program/syntax/DO files for instance), which is of utmost importance for documentation (see chapter 35.2). Your choice of statistical program will probably be governed by the possibilities in your institution and your own experiences, and I have shown DMD principles for SPSS, STATA and SAS with examples. It will also work with many other statistical programs.

Both EpiData and REDCap can export data for use in these programs and others, eventually through a comma-separated (.csv) file (see chapter 33) and by that the Audit Trail (see chapter 4) will be able to work across different programs.

Regardless of the statistical program used, documentation of the work should be made using written programs, which is the reason I disencourage using Excel for DMD, as Excel is not able to produce documentation automatically.

2.5 Other programs for working with DMD

A multitude of other programs exists that can be used for collecting data and document work in accordance with the Audit Trail principles. Some are free like EpiInfo or CSPro; others are sold, like SurveyXact. I have tried them, but do not have deep experience.

2.6 Notation and examples

Throughout the text, I have been using screen-shots from version 4.4 of the EpiData programs and version 6.15 of REDCap, they may differ a little from the actual versions of the programs.

In the text, I use font in violet for commands to be chosen, text to be written and sharp marks for showing keys on the keyboard. Thus, <SHIFT>+<CTRL> means that the keys SHIFT and CTRL on the keyboard should be pressed at the same time, <A> mean the key A.

Part I

Principles of Data Management and Documentation

3 Why focus on DMD?

Everyone working with the collection of data will have their focus on the coming analysis of data. What interesting findings can we report from the information, that we have been collecting?

However, after having read this text on DMD you will be aware that you should instead start by focussing on how you will secure that the data are as good as they can be *before* you start analysing.

Regardless of the source of data, being paper questionnaires, written paper forms from patient sheets, data fed automatically from a device into the computer, respondents filling out questionnaire forms on the Internet, or even from access to register data collected for administrative purposes, faults are bound to happen in every step before you end up analysing them – and you will need to find and handle them.

The reasons for this can be argued in relation to both your own sake and for the sake of society.

3.1 For your own sake

Think of what you did this time last week. Do you remember the details—or are they only vague, partly forgotten fragments? Then think whether you will be able to remember all your own clever thoughts, decisions and practical work related to the handling of your data next month or even 5 years from now when someone stumbles across your data and wants to make relevant use of them. If you are like most other researchers, you won't remember any details.

So you will need ways of documenting your thoughts and decisions in relation to the data in a way allowing both yourself and others to follow the line of thoughts, but in a way that is not consuming too much time to produce. This way you and others will be able to learn what the data are, how they were constructed and handled. The documentation will make it possible to reuse work, to communicate with colleagues and for easing your coming work with the statistical analysis.

But it is also a method for safeguarding your data, should they be partly destroyed by theft, melt-down of the computer or by a faulty hard disks, so that data and analyses can be re-created from the back-up without the need to start all over.

Finally, these principles of DMD will offer protection against accusations of fraud, because you will be able to document what you have been thinking and doing, making it open for others to see how you handled problems encountered during data preparation and how you did your analyses.

3.2 Legislation and Society

For the sake of society, DMD will partly protect against bad science, as the researchers will be more likely to report true findings instead of invalid ones when data are of better quality.

It will likewise protect participants and patients, as they should not be using time to participate in research that is not giving the truest results or being exposed to potential side effects of treatments based on knowledge that was the result of wrong data or analyses.

Finally, the researcher will be able to comply with the national and international codes of conduct in relation to data protection and use, see as an example The Singapore Statement on Research Integrity[4] that has accountability as a main principle.

National guidelines are also of relevance, and for instance, the Danish Code of Conduct for Research Integrity[5] offer guidelines on good scientific practice, implying that society anticipates that researchers are able to document their work.

More formally, data handling is in most countries legally regulated. For instance, Danish researchers are obliged to follow legislation and guidelines from the Danish Data Protection Agency (Datatilsynet) that in parallel with other European countries have implemented the European Union principles of the General Data Protection Regulations (GDPR) in May 2018. The principles covers the use, handling and access to data, and it is part of those principles that changes to data should be documented.

But first and foremost, think of the results you read in any scientific article. You may be able to critically scrutinize the way the author got to his result by reading the *material and method* sections of the paper - but what if the data being analysed were wrong? What if the answers to the items in the questionnaire were wrongly entered or if an error in the syntax changed the result of the statistical analyses? The results would be wrong, maybe even pure garbage – but the worst thing is that you would have *no* way of knowing!

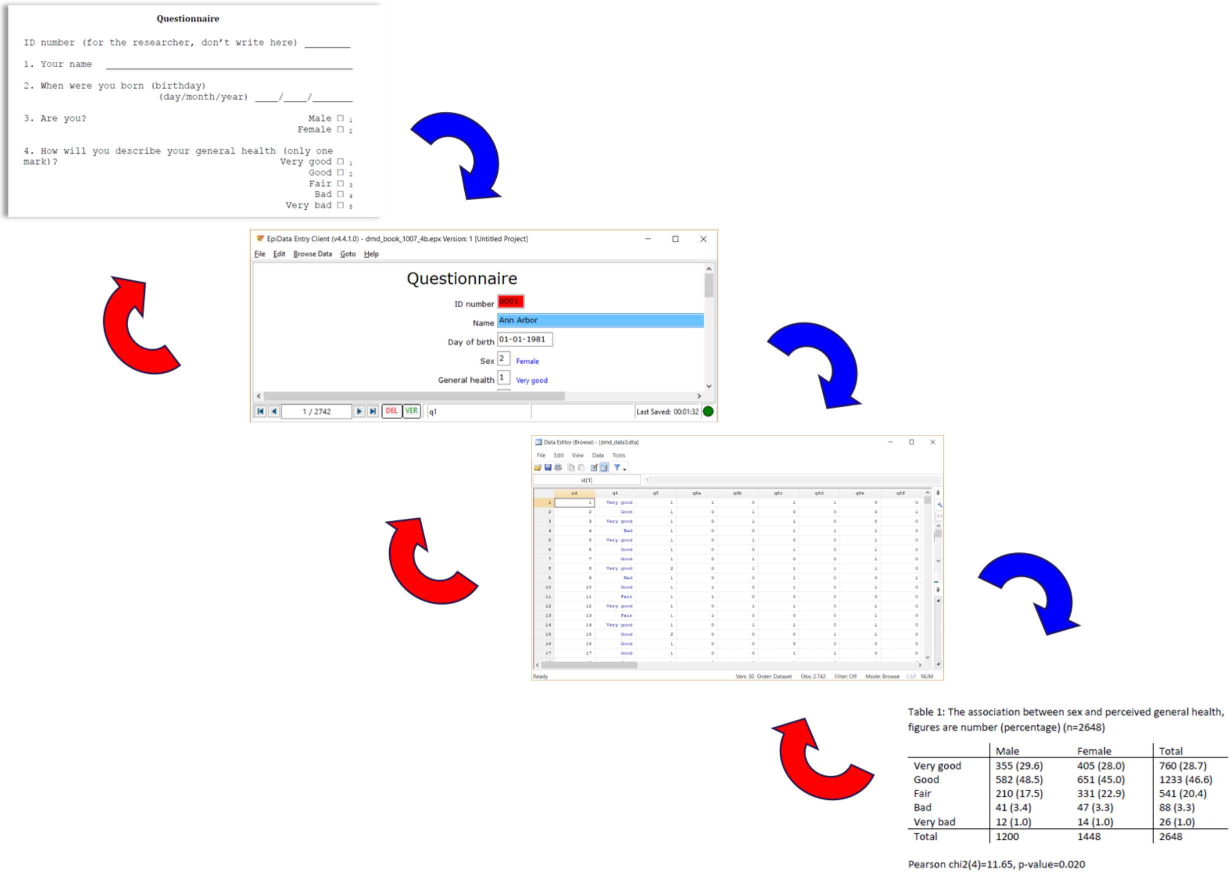
You and Society depends on the authors for doing their best to have reliable and valid data underlying their results and by that a focus on DMD is also crucial for the research community.

[4] 2nd World Conference on Research Integrity. (n.d.). Retrieved from http://www.singaporestatement.org/

[5] Ministry of Higher Education and Science. Danish Code of Conduct for Research Integrity. Copenhagen, 2014.

4 The concepts of audit trailing, data management and documentation

The idea of audit trailing is to be able to follow every step you have made, from the result, back to the analyses in the dataset, further back to the keying in of the collected data and to the individual questionnaire information. This resembles the work that an auditor do with an account, going from the financial results, back to the balance sheet and further back to the individual voucher. This is illustrated in the figure as the red arrows.



Questionnaire

ID number (for the researcher, don't write here) ________

1. Your name ______________________________

2. When were you born (birthday)
(day/month/year) ____/____/_______

3. Are you?
Male □ 1
Female □ 2

4. How will you describe your general health (only one mark)?
Very good □ 1
Good □ 2
Fair □ 3
Bad □ 4
Very bad □ 5

EpiData Entry Client (v4.4.1.0) - dmd_book_1007_4b.epx Version: 1 [Untitled Project]

Questionnaire

ID number 0001

Name Ann Arbor

Day of birth 01-01-1981

Sex 2 Female

General health 1 Very good

1 / 2742

q1

Last Saved: 00:01:32

Data Editor (Browse) - [dmd_data3.dta]

id[1] 1

	id	q4	q5	q6a	q6b	q6c	q6d	q6e	q6f
1	1	Very good	1	1	0	1	1	0	0
2	2	Good	1	0	1	0	0	0	1
3	3	Very good	1	0	1	1	0	0	0
4	4	Bad	1	0	0	1	1	1	0
5	5	Very good	1	0	1	0	0	1	0
6	6	Good	1	0	0	1	0	1	0
7	7	Good	1	0	1	0	0	1	0
8	8	Very good	2	0	1	1	1	1	0
9	9	Bad	1	0	0	1	0	0	1
10	10	Good	1	1	0	1	0	1	0
11	11	Fair	1	1	0	0	0	0	0
12	12	Very good	1	0	1	1	0	1	0
13	13	Fair	1	1	0	0	0	1	0
14	14	Very good	1	0	1	1	0	1	0
15	15	Good	2	0	0	0	1	1	0
16	16	Good	1	0	0	0	0	0	0
17	17	Good	1	0	0	1	1	0	0

Ready Vars: 30 Order: Dataset Obs: 2,742 Filter: Off Mode: Browse CAP NUM

Table 1: The association between sex and perceived general health, figures are number (percentage) (n=2648)

	Male	Female	Total
Very good	355 (29.6)	405 (28.0)	760 (28.7)
Good	582 (48.5)	651 (45.0)	1233 (46.6)
Fair	210 (17.5)	331 (22.9)	541 (20.4)
Bad	41 (3.4)	47 (3.3)	88 (3.3)
Very bad	12 (1.0)	14 (1.0)	26 (1.0)
Total	1200	1448	2648

Pearson chi2(4)=11.65, p-value=0.020

Figure 1 The audit trail principle. We work in the direction of the blue arrows, but should be able to audit in the direction of the red arrows

But researchers work in the opposite direction (blue arrows), from the collection of data, over making them analysable in a statistical program (for instance by keying them into EpiData), then analysing them in a statistical program like STATA, and finally reporting results in tables and figures for others to read. The focus for the researcher and the reader is on the table showing our interesting results.

This principle of auditing will work only if all steps in the chain from the data form to the resulting table are kept open, visible and repeatable. Therefore, if you change a wrongly keyed cipher in the spreadsheet of the statistical program when you recognize a mistake - it will not be repeatable because you would have no way of knowing what was changed, from what original (wrong) cipher – and why. It will not be visible; no one will be able to see that an error was changed. It might even be that you changed the wrong value, thus introducing more faults. Instead, the change should be documented either in a note or as a step in the program of a syntax file, so that you and others will be able to document the changes being made to the original – but faulty – data set. This makes it accountable for you and for others.

Using these principles will also prevent the situation where you lose weeks or even years of work due to the breakdown of a hard disk. Having documented your work will make it possible to re-create the work from scratch - even years from now with relatively little work, as long as you have the original data files and the documentation of the changes made.

The principles are relevant to every type of quantitative data sources – from interviews using pre-formatted questions, to survey questionnaires, to information extracted from patient files or laboratory sheets – but also when using secondary data provided from registers or from an electronic device through some sort of interface, as these data may turn up to have faults that need to be found and changed in a repeatable, visible and documented way. I will however use questionnaire data to show the principle in this text.

4.1 The Audit Trail

Like an accountant that follow the stream of money from the balance sheet back to the account book and further back to individual vouchers, being able to identify the missing dime in the balance sheet, DMD should be planned for the researcher and others to find their way from the table or figure in the paper back to the statistical analysis and the syntax that made them, and further back to the individual data constituting the background for the published work.

To do so, one must plan, but also document on the fly – and in the words of Cartwright and Seale above – be obsessive about it. On the other hand, choosing to work in accordance with an audit trail is relatively easy using the computer programs and principles given in this text - and some practice!

For the auditing to work, all corrections, decisions, and changes in data should be documented during data entry (in EpiData or REDCap), in building the dataset and derived variables, and during analysing data in the statistical software (using SPSS, STATA, SAS, or other software programs).

When working with data, always have the need of the accountant in the back of your head. Help the accountant by commenting your choices and changes of data – by doing so, you will also help yourself when you read the documentation even years from now!

4.2 The audit trail in collaboration

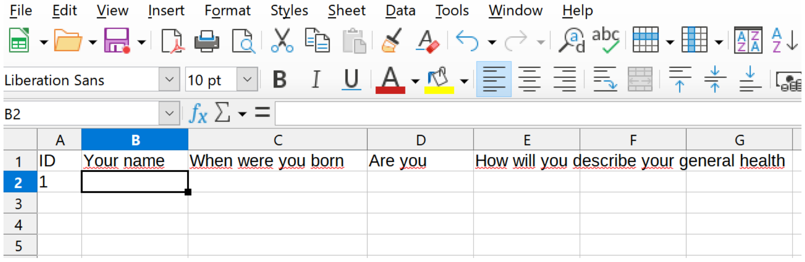
Who will be keying your data into the computer - and who will help you with those fancy statistical procedures? It might be yourself, that's cheap and you know what you want. But even under these circumstances, you should work according to these principles to ensure that data is as good as they get.

More often, you will be collaborating with others and they should be observing the principles, too. It might be that others are providing data as well, or your time might be allocated for other tasks meaning that you must persuade or pay others to do the hard and boring work of making data accessible for analysis through keying data in EpiData or REDCap. This makes it even more relevant to plan the entry in detail and to formulate instructions for the person keying in or delivering the data (I will refer to this person as *the keyer*), to secure that the data end up being the way you want them. This implies that you must prepare instruction on how to handle the data for your keyers to work in accordance with your thoughts and to instruct them to document all doubts and the decisions they were forced to make when you were not around.

5 Database, variable and records

Data for our analyses reside in datasets. We use ciphers to represent different types of information in order to be able to work with statistical software to describe and analyse the information.

These data can be regarded as constituting a database structure like the one used by Excel or other spreadsheet programs, where rows constitutes the records, columns represents the variables and the cells contains the values for that particular variable for that particular record. Most statistical programs can present the data in the same way and it is of course possible to key in the data using the spreadsheet, although I strongly encourage against it, as it will give more errors during data entry.



	A	B	C	D	E	F	G
1	ID	Your name	When were you born	Are you	How will you describe your general health		
2	1						
3							
4							
5							

Figure 2 Database structure with records (horizontal lines) and variables (vertical rows)

5.1 Measuring scales

All information in the variables of a dataset can be defined as belonging to one of three different measuring scales (almost).

```
3. Are you?                     Male □
                              Female □

4. How will you describe your
general health (only one mark)?
                           Very good □
                                Good □
                                Fair □
                                 Bad □
                            Very bad □
```

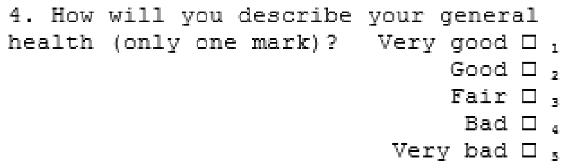
Figure 3 Two types of measuring scales - a nominal (sex) and a ordinal scale (self-reported health with 5 categories)

Nevertheless, they will often include the same numbers in the cells, but the scale type is both determining how the information might be described (using proportions, median and interquartile or mean and standard deviations)[6], how it might be shown in figures (scatter-plots or histograms) and how it should be analysed (using t-tests, Mann-Whitney or chi-square-tests, for instance)

Nominal scales just represent named information. Sex is represented on a nominal scale, males might be represented as "1" and females "2", but that is just representation – and it could have been named by the opposite numbering (1=female, 2=male) without problems. Each number just represents a named group.

Even when using numbers like a postal code, it is still just names of postal areas or towns, and nothing is implied on the ordering of the information with the actual numbers.

Ordinal (or rank) information imply that there is some kind of order in the data—some groups are more or less than other, something is being better or worse. The respondents can for instance choose to evaluate self-reported health in relation to a series of wordings: very good, good, fair, bad, very bad; We will think that having a good health is better than having a fair health and this again is better than having a bad health, but we can't be sure if the distance between those three answers is the same - even when using numbers to describe the answer categories.



```
4. How will you describe your general
health (only one mark)?   Very good □ 1
                               Good □ 2
                               Fair □ 3
                                Bad □ 4
                           Very bad □ 5
```

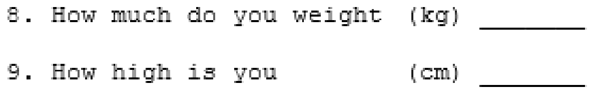
Figure 4 Number coding of answers on a ordinal scale

We would represent the information by numbers 1, 2, 3, 4 and 5 – but might use the numbers opposite. But using the sequence 3, 4, 1, 5, 2 for the five sentences would seem odd, suggesting some kind of inherent order in the answers.

Some measurement scales look like continuous information (like APGAR-scores, that sum evaluation of skin colour, pulse rate, reflexes to stimulation, activity, and respiration in newborn babies on a scale numbering 0-10). In this case, although using numbers from 1-10, the distance between numbers are unknown, as we do not know whether APGAR 10/10 (that is the APGAR score 10 minutes after birth) is the same distance from 8/10 as this is from 6/10. The ruler can be regarded as being made of rubber! This means that APGAR scores should be regarded and handled as

[6] All information consisting of ciphers might be described by e.g. mean and SD – but in many cases (and always when being nominal or ordinal scales) this would be formally wrong and not advisable

ordinal data. The same goes even for VAS (Visual Analogue Scales), that in appearance resembles a ruler, but it has repeatable been shown that this ruler is also made of rubber.



```
8. How much do you weight  (kg) _______

9. How high is you         (cm) _______
```

Figure 5 Ratio-interval scale questions

Ratio-interval scales are measuring on non-rubber rulers, the distance between numbers is the same in all parts of the scale. This would go for height, weight, blood pressure and other biological values. A further division might be made between interval scales and ratio scales, according to the existence of an absolute zero of the measurement, as for instance for length. On a ratio scale, something can be twice or half something else. In this text, this distinction is not used, however, as it has no implication for DMD.

The focus on identifying the type of measuring scale is that it forces us to reflect on the type of information presented, and it will steer us toward a relevant statistical presentation of the data when reaching the analytic phase of the project.

Besides these three different types of data, **string or text** variables can hold alphanumeric information consisting of letters and numbers as "F10.1". Even if numbers are present, they will normally not allow us to calculate anything from the information, but the information in the string variable can on the other hand be used to allow the respondent to deliver information, and for the researcher to collect information, that can't be coded in advance.

Dates are a special form of variables holding information on the day, month, year and time. Statistical programs often have a fixed date as an internal zero-point and will calculate the date in relation to this (if the internal zero-point is 01/01/1960, the number of days are calculated in relation to January 1st, 1960, so that June 30th, 2017 is represented as 21,000 days from this date, and days before that date are simply negative). Regardless of this, dates may be shown (represented) as DD/MM/YYYY or MM/DD/YYYY and this should be specified in the documentation of the data. 01/12/2020 in the European way of representing dates are different from 01/12/2020 in the American (around 323 days!)

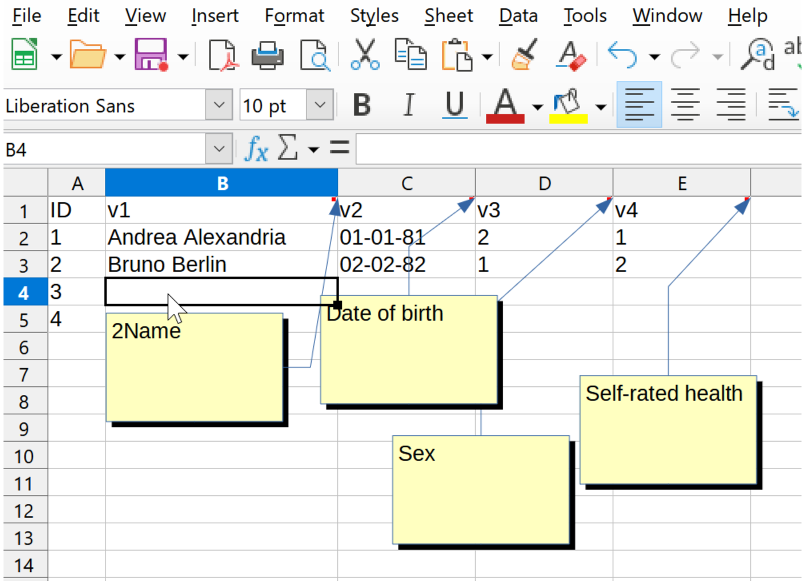
5.2 Variables

The information relating to one question or item might be collected in a variable. We will have to give the variable a name. It can either be what the variable is including “agegroup”(or “age_group” as most programs won't work with names that have spaces in them) - and often also gives problems when moving between programs). Or we might simply give it a short non-informative name, that is easy to write – and might relate to for instance the question number in the questionnaire ("q1" for the first variable, "q2" for the second and so on).

The variable must be declared as a number (integer or floating), an alphanumeric or a date-variable. For most variables, an integer representing for instance age in years, or the coding of an answer as 5 for "very bad" (see figure 4) will work.

The variable will have a defined length, making it possible for the variable to store the desired information, if for instance having a variable containing "weight" of a person, it should be able to handle values up to 200 kg – or whatever highest number is expected. This would demand a length of the variable of "3" (that is hundreds, tens, and ones kilograms), allowing numbers from 0 to 999. We should also chose whether we would need to have fractions of the number, for instance to the nearest 100 grams, in that case the length should allow for a decimal point to handle 75.6 kg.

If we have chosen to name our variable "q1", at some time we might no longer remember what information the variable was holding, and it would be relevant to give the variable a *variable label*, describing the *content* of the variable. Most programs allow the user to label variables. This is not for the sake of the program but merely a help for the user – and it should be used from the beginning in order to document the information stored with the variable.



	A	B	C	D	E
1	ID	v1	v2	v3	v4
2	1	Andrea Alexandria	01-01-81	2	1
3	2	Bruno Berlin	02-02-82	1	2
4	3				
5	4				
6					
7					
8					
9					
10					
11					
12					
13					
14					

Figure 6 Labels for variables. Naming of variables are short

5.3 Values

Each variable will be able to hold one of a number of possible values for the given record. It might be the age of that person, a number representing the answer given ("5" for representing "very bad"), or a short text in a string-variable, as the response to the question given. The respondent might have stated: "the reason that I did not go to the shop was that the door was locked" and this should be recorded just as it it stated by the respondent.

The type of value was defined in the declaration given to the variable.

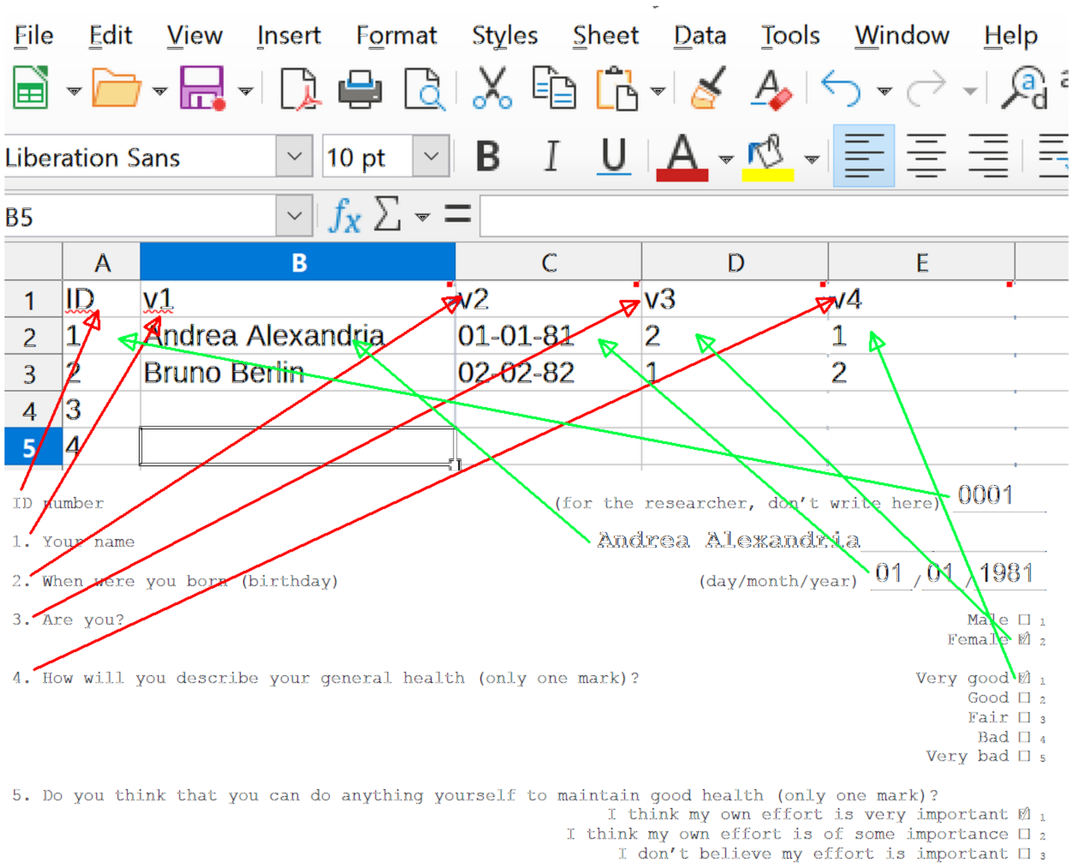
If the scale is measured on a ratio-interval scale we normally just record the number as the answer (the weight in kg), if it is a nominal or nominal information the number should be recorded, but it would be advisable to tag each value with a *value label*, translating the number with the value being represented, combining the number "5" with the label "very bad".

String variables are used to allow the respondents to deliver an answer, normally further management is needed and the answers needs to be recoded for analyses at a later time. The information can be keyed in without recoding, often *the keyer* will simply write the text, but even then it could be either capitalized, written in only small letters or with a capitalized first letter. "John Doe", "john doe" and "JOHN DOE" is representing the same information, but will be interpreted different by the computer. So when using string information, it is advisable using for instance only small or capitalized letters.

5.4 Records

The record keep all information on the subject – that is the actual content (value) of each variable for this subject.

The relation between questionnaire and coding might look like this for a record identified by number 1 in a survey questionnaire. The question number is used as variable name, the (abbreviated) information that was asked is used as variable label, the answers are typed in the cells and for nominal and ordinal variables, the numbers are used with a label of the values (not shown in figure).



	A	B	C	D	E
1	ID	v1	v2	v3	v4
2	1	Andrea Alexandria	01-01-81	2	1
3	2	Bruno Berlin	02-02-82	1	2
4	3				
5	4				

ID number (for the researcher, don't write here) 0001

1. Your name Andrea Alexandria

2. When were you born (birthday) (day/month/year) 01 / 01 / 1981

3. Are you?
Male □ 1
Female ☑ 2

4. How will you describe your general health (only one mark)?
Very good ☑ 1
Good □ 2
Fair □ 3
Bad □ 4
Very bad □ 5

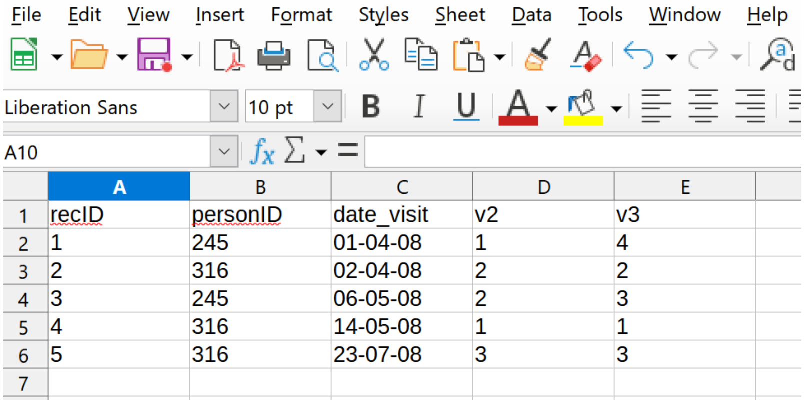
5. Do you think that you can do anything yourself to maintain good health (only one mark)?
I think my own effort is very important ☑ 1
I think my own effort is of some importance □ 2
I don't believe my effort is important □ 3

Figure 7 The relation between answers in the questionnaire and information in the data set (red arrows show the identification of variables, the green arrows how the content of the questionnaire is represented. Text from the questionnaire is used for labelling the variable and the label for values come from the answering categories

Normally a record holds information relating to different variables for the same person, but a record can represent other content. For instance, the Danish National Patient Register holds information relating to the discharge and one person might have zero (if never hospitalized), one (if having only one hospitalization) or many records. For each hospitalization, the variables might be the same or different (for instance if the person has been on different wards or different diagnoses for the different visits) for the same person.

Regardless of structure of the database, each *record* should have a unique identifier of that record. The record might also include other identification, for instance for the person that has been hospitalized, this would also be a unique identification number of the *person*, preferably we should be able to identify the person (unique CPR-number or name and address). The information identifying the person should, however, be kept apart from the rest of the data as soon as possible in order not to be able to identify the person during analyses. This will often be done by dividing the database into a key-file including only record ID and person ID, and a de-identified database with the record ID and the rest of the data (see chapter 6.7), but without the person ID. By having a record ID it is possible to reconstruct the data if necessary by merging the two dataset.

In the example below, 5 records are included, each identified with the recID, but the 1. and 3. are both related to the same person (same personID), seen at different dates, and the 2., 4. and 5. are all related to another person.



	A	B	C	D	E
1	recID	personID	date_visit	v2	v3
2	1	245	01-04-08	1	4
3	2	316	02-04-08	2	2
4	3	245	06-05-08	2	3
5	4	316	14-05-08	1	1
6	5	316	23-07-08	3	3
7					

Figure 8 Example of database with identification of both record and person.

5.5 Database structure

Variables and records are structured in a data base, the simplest being two-dimensional, for instance represented in rows and columns in a spreadsheet (see figure 2)

The records are normally shown horizontally as records and variables will then be shown by columns.

Databases can be large, holding millions of records and variables. Very often, it is easier to divide information into several databases and then link them using an identifier. This way redundant information (same information, that is repeated several times) can be kept at a minimum.

But in order to analyse data, the database structure should normally be rectangular, either during the setting up of data - or by using the statistical program to fetch data from different sources. This will not be covered in this text.

Like labels for variables and values, *dataset labels* can also be defined for the dataset (what is the data, when, and where was it collected).

This ends in six issues to consider in relation to preparing the database structure, variables and values to be decided and collected in a final documentation of the dataset:

Know the structure of the database before setting it up

Put labels on all datasets

Abbreviate variable names ("q3") to ease the statistical analyses later on

Always put variable labels to the variable and identify the source

Label values for all nominal and ordinal variables

Pre-code questionnaires and sheets

Some of this work will be handled by using EpiData or REDCap to set up the database (see chapter 14 and 27)

6 General principles for setting up the dataset

6.1 Sketching the data structure

Normally the data base is structured with a row for each respondent and columns for variables, but if data is collected at several instances (for instance the same questionnaire used one year apart) or using different tools (a questionnaire and a separate chart for measuring height, weight and blood pressure) it might be more rational to have several databases with information identifying records that belong together. This way redundant information can be minimized, as long as an ID-number identify those records in different datasets, that is related.

It should be determined how the data structure for the project will be in order to be able to plan for the collection of data.

6.2 What constitutes a variable?

In most instances, it is reasonably easy to decide what should be defined as variables. In principle, everything that is codable should end up in a variable.

But some questions are not straightforward. If for instance, a question has several possibilities for marking an answer, having just one variable will call for special coding to separate those crossing out the first, third and last answer, from those marking the second and third only. In that case, it is much easier to define a variable for each box (variable 6A, 6B, 6C etc. in the example used below) and then just register whether this particular box is crossed out or not. From that every combination can easily be recoded afterwards.

6.3 Structure for coding answers

Use the same structure everywhere. If you chose using 1 for coding "no" and 2 for coding "yes" in a question, this should be used throughout. This will also make labelling much easier (see chapter 15.2.2).

The codes can be printed on the data form to make keying in data easier (see figure 4)). This does nearly never interferes with the ticking off by the respondent and makes it easy for the keyer to determine what to record.

6.4 Missing values

Missing information for a variable should be coded to prevent being insecure whether the answer was in fact missing or whether it was mistakenly not entered. This is done by designating a number for this – and again, use the same principle throughout.

At least code whether the information is missing, but you should at least consider distinguishing between:

- Not stated – when a variable should have a value but hasn't been answered. Use one code for this, for instance, 9 if the valid answers are 1-5, 99 if valid values are including 9 etc.
- Irrelevant – the respondent should not answer, for instance by being filtered (males not answering questions on being pregnant). This could be coded by another number, for instance, 8 or 98.

Some would prefer using negative numbers like -8, -9, -98, -99 etc. in order to distinguish missing values from valid answers. In statistical programs missing values will not be included in the calculation if properly marked, but it is normally not of any concern which numbers are used as long as they are not valid answers.

In STATA missing values are represented with a dot and eventually a letter (., .a, .b, .c etc.), which is not easily entered, as alphanumeric information is not readily keyed into for instance EpiData. I advice using a number and then recode it at a later time (STATA has a special function for that (mvdecode q1, mv(9)) that will change for instance a 9 used for missing values in variable q1 into the ".").

6.5 Preventing errors by using dedicated entry programs

Data could be entered directly into the spreadsheet of Excel, SPSS, STATA, R, SAS or another statistical program, but this induces errors. In some instances, errors will just be random and produce higher uncertainty, but errors can easily turn up to be a source of bias, for instance if some errors are more likely than others.

When I use a numeric keypad and type numbers without looking for a long time, my "personal" error is to type 4 instead of 6 and the opposite. I do that more than confusing for instance 8 and 2. This means that I'll produce more errors in relation to typing 4 and 6 than when typing other numbers—and this will lead to systematic error.

To avoid this, dedicated programs can minimize the possibility of entering errors by using filters, frames, helping texts, checks, validation texts etc. In section two and three I'll show how the principles can be used with EpiData and REDCap.

Furthermore, entering data should be seen as an integrated part of the data management part and I would suggest considering double entering data in order to prevent keying errors.

Even if your data come from other registries or are scanned from paper forms and read by an OCR programs, faults will be inevitable. By planning DMD also for these data, to control them systematically and to document them afterwards will help to secure that data are as good as they can be.

The GIGO-problem of all research is that if you put garbage into your statistical program you will, without doubt, get garbage out—and what's even more frightening; you won't be able to tell!

6.6 Identification of respondent

During entry, it is necessary to a certain point to keep information on the identity of the respondent, both in order to be able to go back for controlling the questionnaire if suspicion of errors occur, to be able to link information across databases and eventually to identify the respondent with the purpose of approach him, if e.g. a blood test shows abnormal values.

But on the other hand we will refrain from keeping personal identity as it violate GDPR, so we will often "name" the respondent with a unique number and then at an early point removing the personal identification and keep a pseudo identification by assigning a unique ID. The relation between the respondent ID and the name of the respondent should be kept in a separate file (see below).

When keying data from the questionnaire or other sources, the most crucial information is the identification, without proper ID it will not be possible to return to the previous step in the trail. This means that the ID must be in the file and calls for ways to ensure that the ID is correct. I suggest to include two different variables for the same ID, one encountered when starting to key information relating to the respondent and a second instance when leaving. This will prompt the "keyer" to secure that the information is correct. It is also possible to having the entry program comparing the content of these to variables.

6.7 Working with un-identifiable data

When moving into analysing data knowing the identity of the respondents is not necessary. This means that identification should be removed from the data used for analysis, but preserving a key file that enables linking information on the record with the identification.

This is done by ensuring an identifier for each record, often just by giving the consecutive and unique ID-number assigned to the respondent in each row, although it might also be constructed of several variables (an ID-number of the respondent and a date, if the same person is present several times in the dataset, for instance for several visits at the General Practitioner).

The ID-number of the respondent should then be included in the *key* file together with the identification information (name, address, CPR-number etc.) and this key file should then be kept unavailable in a safe place. In the file used for analyses, the identification information is removed, but the ID-number is maintained. This file will be without identification, but the ID-number can be used to look up the identity either by flicking through the key file or by merging the key file and the analytic file.

6.8 Non-participating participants

In most instances, not all approached respondents will participate. It would be relevant to keep information of these non-participants, give them an ID-number and to keep them in the files, for instance by keeping the date of birth, sex, and dates for sending out information. It might help to describe non-participation at a later stage and should be planned from the beginning of the study.

This suggest five things to consider in relation to the database structure, variables and values:

Determine your data structure

Convert all information to relevant variables and define the possible values

Identify possibilities of missing information and plan for their coding

Plan for keeping all participants in the data

Plan for putting data together and at the same time un-identify the dataset

7 Error identification and correction

7.1 Searching for errors

Errors are found in any part of the process from collecting data until writing up the report on the study. They will emerge both as a result of careful and planned description of the data and accidentally during analysis.

Errors should be found and changed. But it should be done in a way that respects the principles described above to secure documentation.

Use time and effort to prevent and identify errors, first by securing that the datafile is made in a way that reduces the possibility for errors to occur, secondly to re-enter all or part of the dataset and identifying and documenting both the actual errors and the prevalence of error, and to correct the errors in a new file (see chapter 37).

When the data set is ready, it should be described by making tables of frequencies (for nominal and ordinal scales) and distributions (for ratio-interval scales) and graphical description, to make sure that the values correspond to the defined possibilities and that outliers in ratio-interval scales are identified and controlled.

Description of calculated variables (eg. "age" as the subtraction of the date of birth from the date of visit) will also identify errors in either variable.

Lastly, two-way cross-tabs of relevant variables will further identify groups, that should not be present (males with former pregnancies) or are unrealistic (an 18 year old with 4 children).

Not everything turning up in these searches are errors, but it will be necessary to carefully looking through the original material (questionnaires) in order to determine this. If it is an error, it should be corrected, otherwise it might be necessary to delete the information. Either way, changes should be documented.

7.2 Correcting errors

Every change of values should be documented, either by writing a note in the documentation and/or preferably in a program-file in the statistical programs, that ensure that the actual change is also documented. This is described in detail in part IV (see chapter 37).

7.3 Double entry and validation

Even when being obsessed by the idea of creating a thorough and right data set, entering data will become boring and you lose concentration.

By re-entering *part* of your dataset, you will have the opportunity to estimate the proportion of errors and decide whether you will be satisfied with this.

By re-entering *all* data and comparing the two files, it is possible to identify those situations, where errors occurred. Using this comparison, you will be able to look up the right value in the original source (questionnaire or the like) for those different cells. Setting up a third and final data set as a copy of one of the two and changing the errors in this will create an accurate data set.

Using EpiData for this is described in chapter 22.2

Six things to consider in relation to error search and correction:

Plan for doing error correction - it takes time.

Write a codebook from the keyed data

Look for invalid information in frequency tables and histograms

Look for outliers in ratio-interval scales using box-plot and descriptives

Cross-tabulation might reveal further errors

If keying data plan for double entry of all or at least a proportion of data

8 Modifying data

The dataset will never be completely ready for analysis when keying is made, as new groupings of answers will be necessary, variables as BMI should be calculated from height and weight and the need of creating new variables containing combined answers from several original variables will be planned or emerge during work. Variables will eventually have to be recoded in order to analysing them, for instance when dichotomizing a variable for logistic regression analysis.

I advise to refrain from overwriting existing information, often a recoding will go wrong, and you will need the original variable to compare with the new in cross-tabulations to ensure that calculations were actually made as planned.

Although it might seem tempting to create the variables on the fly as you need them, it will be hard to keep overview. Instead, all modification should be kept in one dedicated part of the syntax file, so you will be able to easily find them again.

All these modifications of data should be documented so errors can be tracked back. This is further elaborated in part IV.

8.1 Control changes to the dataset

Always control the modification made by comparing the "old" and the "new" variable in order to secure that the modification is as planned. This can be done by cross-tabulations for nominal or ordinal variables or by descriptions of a ratio-interval scale variable, stratified by the new ordinal variable. Control that coding is ending up as anticipated, here for instance that collapsing a variable with 5 possibilities for answers end up in the right two groups.

old (horizontal)					
and new (vertical) variable	very good (1)	good (2)	neutral (3)	bad (4)	very bad (5)
good	226	220	210	0	0
bad	0	0	0	248	158

9 Receiving and combining dataset

Data might be keyed by different persons, it might be collected in different forms, or you might have others giving you access to their data. This imply that you have to control the data of others and then eventually combine them in one dataset.

9.1 Receiving data from others

Receiving a dataset from another source will ease your work, but it still leaves the work of managing and documenting data. Having to rely on others initial work increase the need for control and documentation so you are comfortable that data is correct.

You should start by asking for a copy of the available documentation from the donor and further conduct frequency-tables to learn the data. If data is not labelled they may be hard to use, but often the documentation will include information on this, and it may be used for setting up variable labels and value labels in the received data set if not provided by the donor.

Further, it will be relevant to compare analyses from this dataset with those made from the original data to ensure that data act as anticipated.

9.2 Working with collection of data from the Internet

It is often possible to harvest data from the Internet, for instance by using REDCap or SurveyXact to collect survey information.

Apart from the evident methodological problem in maintaining confidence that the sample is representative, these data are often complete.

But even when care has been made while setting up the SurveyXact picture, they will occasionally be in another format than anticipated, will often be named after principles that are not corresponding to the intended name and might have the variables of different types than anticipated (integers instead of float). This must be changed during data management and much work can be anticipated if combining with other sources.

If using both a paper and an Internet version of the survey for collection of data, time and resources for tidying up the SurveyXact file before putting it together with the data from the paper data should be foreseen.

It should be carefully controlled by making comparison of the original and the merged file.

9.3 Appending and merging data

It is necessary to plan the combination of data from several datasets in order to be sure that data behave as planned.

The different records might either be combined after each other ("end to end"), this would be the solution if different people had keyed in questionnaires from different respondents. The same respondent would not be in both samples. This is called concatenation of datasets (see figure below, left). It is imperative that the variables in both dataset have the same name and is of the same type if the appending is to be successful.

The other possibility is that the *same* person has variables in several datasets, for example, if information from questionnaires were in one dataset, and clinical information was found in another dataset. In this case, datasets should be merged “side to side” and the two or more records needs to be identified by a unique ID-number.

If merged “side to side”, one or both data sets might include records, that are not present in the other data set (see figure below, where ID number 4 is not present in the data with variable V3 and V4, while ID number 5 is not present with V1 and V2). In this case, one might choose to include only records with ID in both dataset (inner merge), or might include those in the parent (left merge), the other (right merge) or both (outer merge) dataset. In the figure at right an outer merge is depicted.

The resulting dataset might look very differently according to the choices. This calls for carefully planning of the combination of data.

Eventually, also this combination of datasets should be carefully controlled, eventually by marking each record according to the source of the record.

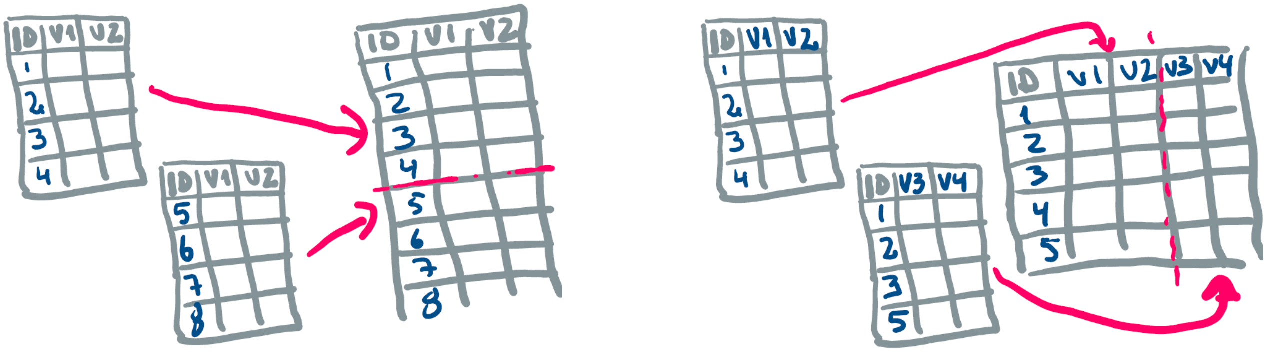


Figure 9 Merging two databases, end-to-end (left) or side-to-side (right), ID is the key variable

10 Documentation

The data should be documented in order to give a full description for other users to read at another time.

Documentation should be created at three levels:

- Study and dataset – how did the material originate, who was responsible, when did collection take place, how many were asked, what was the number of participants etc. This corresponds to the material section of the paper, that is to be written later.
- Variable – a systematic description of each variable in the dataset. It should encompass name of the variable, label with description of the question, type (width and number of decimals (in order to ensure secure transfer between different programs))
- Values – value and value labels for each variable, related to the type of variable (nominal and ordinal should have labels for each value; for ratio-interval scaled answers should have measuring unit in the variable label).

This information will normally constitute a codebook - and it can be derived from both EpiData, REDCap and most statistical programs (see chapter 23 (EpiData), 30 (REDCap) and 39.2 (statistical programs).

The codebook should be started already during planning, and will be appended during data collection and analyses.

All changes to data, including the merging and calculation of new variables should be documented as well, both outside the dataset (for instance by saving the reports produced by EpiData) and inside the dataset by remembering to set up labels for newly created variables and values.

You would also need documentation of the different syntax files produced for analyses (see chapter 35)

11 Archiving data

Data that have been collected for scientific purposes will eventually need to be disposed of in accordance with the legislation. But before you delete the data, think of the possibilities to reconstruct work in accordance with the Audit Trail principle. Will you be able to go from tables and back to data in a reasonable detail?

According to Danish legislation, data will need to be destroyed, anonymized or archived at the end of the period, typically after 5 or 10 years. On the other hand, the need for being able to document research will demand that log files (and by that also data) should be kept for a number of years after the publication. This means that normally you should be able to document your work for up to 10 years - and maybe more.

Finally, it might also be relevant to return to the collected data at a later time in order to re-analyse or re-collect information on the original cohort and in that case the analytic file as well as the key should be kept with permission.

Destroying data can easily be done by deleting all files, afterwards a program, that can erase the hard disk should be used so that re-establishment of the deleted files is no longer possible. Backup files should also be deleted.

But data might instead be anonymized completely by destroying only the key-file (6.7) and eventually other material making identification possible. If data are very detailed it might also include deletion of certain variables to ensure that respondents are not identifiable. Evaluate whether any of the respondents would be able to identify themselves from the data if you gave them access. But in most instances it is better to keep anonymized data (if permitted) and the program-files, than destroying all data.

11.1 National archives (Statens arkiver)

In Denmark it is possible to archive data in the National Archives (Statens Arkiver, SA) if they are regarded as relevant. In that case, data with complete documentation might be handed over to SA for storing. The achieved data might be stored with the key file identification, making it possible to re-use data for other research purposes.

EpiData (see part II) can export data in the DDI format used by SA. The archiving at SA will substitute for deleting data (you should then of course delete all back-up in your own computer).

11.2 Using REDCap for archiving

REDCap (part III) is used for capturing research data and can also archive the key file apart from the data itself in a secure manner. The program has the possibility to document access to files (who was been looking or changing data at what time?), which is necessary in order to comply with GDPR regulation when using identifiable data, see 34

You might use EpiData, REDCap or other dedicated programs to easily put the principles in this part in effect when working with your own data. This is described in the next parts of the book.

12 Structuring your computer and securing backup

First, your will have to rely on your work with the computer, and should set it up to aid your work. The developers of Windows and Mac operating systems seem to think that all files will be found using the search function of the computer, and consequently, all files are placed together in the "document" section. This is not a rational way to work with databases and projects. Below are suggestions for setting up your computer in general.

12.1 Use folders

Develop and use a structure for your files in folders. You might have one folder per project, eventually folders in folders for keeping things together. This will make it much easier to find those files that belong to the same project – and it will make backup much easier. I would discourage having files on the desktop, they tend to disappear. Put them in the right folder from the beginning.

Keep codebook, epx-files, dta-files and other files belonging to the same project together in the same project folder.

You might also consider dividing your hard disk into several logical drives, keeping for instance drive C: for all programs and then have the project folders in drive D:

This will make it even more easy to backup, as you will just copy drive D: to a different location or a USB or DVD drive.

12.1.1 Default folder for a program

Programs will often by default place new files in the same folder or a sub-folder for the program itself – or it might place them in a "system" folder, for instance using "documents" or an application folder. This might make it difficult to find your files again, especially when trying to back up the files.

Instead, use the ability of most programs to define the folder for saving new files, and change it according to the choices of the folder structure discussed above. You should then secure that your program places un-anonymized information (see below) on your encrypted hard disk.

12.2 Backup regularly

I'm sure you have heard it before; backup your files. Develop and use a structure for backing up your files. This can be done using dedicated programs, that will automate the process – or you might just write in your calendar that you should copy the folders or the drive to another location on that day.

Using automated software is easy – but remember to control it occasionally; it might stop working for some reason.

How often you should backup your data is up to yourself to decide, but consider how many hours of work you will accept spending reconstructing files that were not backed-up – if you will accept to redo one week of work, then backup should be done every week, if you would accept redoing two days of work, then backup every second day. The important thing is that you should make a habit of it and use a short amount of time to protect yourself from data loss.

Your computer department may have a back-up of your files, this is normally working, but I have encountered a situation where the data was not available after all when I needed it, and therefore I have made my own backup schedule.

The first full backup of your folders or D-drive may take some time, but after that, you can settle with copying only those files that have been changed since the last backup. This is called incremental backup and you would need to keep the set with the initial and all incremental backups in order to be able to reconstruct the folders - until you make a new full backup. This might be done a couple of times a year.

Backup sets should be controlled - they may contain errors and in that case they would not be of much help. Keeping several sets of full backups would be advisable.

12.3 Storing backup data

Data from your backup may be located at a hard disk, on a memory (USB) sticks, or on CD/DVD. If the media is containing data that are not anonymized (for instance during keying in data), these media should be locked up in a secure physical place that only you have access to.

Remember that media may also fail, and I suggest to implement a structure with two independent media (two hard disks or two USB-sticks) that are used in turns. I would keep the initial full and the set of incremental backups on the same media, if doing a backup every week, one media would be used every second week, if it was then damaged, I would be able to reconstruct the content from the other, loosing up to a week of work

12.4 New or reused files

You might also consider when you need to make a new file – and when it is OK to overwrite the file with a changed content. In most instances, computers don't keep track of changes unless specified (Word has the possibility of "track-changes", and OneDrive, as well as other Cloud-based drives, keep track of changes to files, at least for some time), so you would need to consider the need of re-constructing the changes made. For instance, writing a manuscript with others might work best if new versions of the files are saved between each round of changes – but on the other hand, it might not be necessary to keep track of all individual changes.

It might be relevant to invent some kind of naming of the files, making it possible to identify the order in which they were changed – for instance by including the date of the version in the file name.

12.5 Encryption

If having un-anonymized[7] data on your computer, this hard-disk should either be placed behind locked doors (like the archiving mentioned above) or – if you are using a laptop – the hard-disk must be encrypted, meaning that if you lose the computer, reading the hard-disk will be impossible. It is not enough having password-protection on the computer, clever people will be able to access the hard disk.

Your computer department may help you, otherwise, Windows professional versions have a built-in BitLocker encryption function, that allows you to encrypt and decrypt either the entire hard disk or (better) the logical drive with the data. You may also use other software that can encrypt logical drives (for instance freeware programs like TrueCrypt or DiskCryptor).

Whatever you chose, the idea is that you must "mount" the drive using a strong password, when mounted the drive is accessible without restriction, but when the computer is closed the program should automatically dismount the drive, making it impossible to read without the password.

This means that you would need to remember the password for having access to the encrypted drive – it is virtually impossible to get access without it.

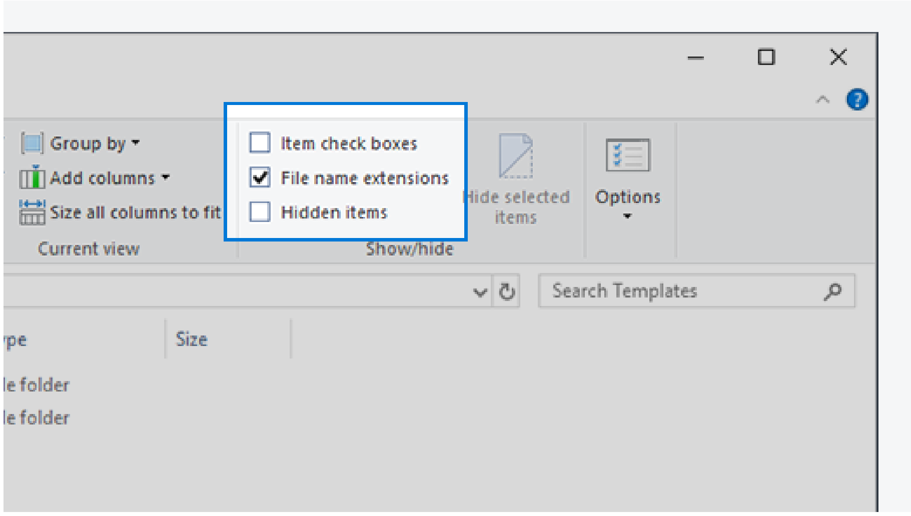
[7] Only data without any possibility of identifying the person are anonymized; if keeping a key-file or if people may recognize themselves, data is only "pseudo-mized" and are still covered by the GDPR even if the do not contain identifying information. Anonymized data are not covered by the GDPR

12.6 A note on working on Windows computers

12.6.1 File extensions

Most programs will produce files with an extension that identify to which program the file belongs. For instance, Word will produce .docx files, Excel will produce .xlsx files and so on.

By default Windows is not showing the actual extension, but rather show a graphical representation – but this might be misleading, as you might have tried to open a .xlsx file with Word, by that having taught Windows that you think the two belong. Instead, looking at the extension, you will be able to see that it is actually an Excel file. This can be changed either in File Explorer under "view", where you mark that you want Windows to show you the extensions (see below) or in the control panel once and for all by telling Windows to always show the extensions.



☐ Item check boxes
☑ File name extensions
☐ Hidden items

Figure 10 Select viewing the file name extensions in file explorer

12.6.2 Open files from within the program

Windows will show the program with which the file is associated, making it easy to double-click the file and by that opening the program and reading the file. While convenient, it will occasional result in programs not behaving like you would want it to.

Instead, I recommend *always* to open the program first and *then* from within the program to open the file. It secures that you always know what is going on – and it makes it possible to document the work you are doing, for instance the actual file being used (see section III).

12.6.3 Copying files

It is possible to copy files just as you would do with text. Mark the files and right-click them, choose copy and then right click in the new folder and choose paste. You might also mark the files and move them with the mouse while keeping the left mouse button pressed.

12.6.4 Using write-protection of files

Right click the file in the folder and choose “Properties”. Here, it is possible to mark the file as “write-protected” in order to prevent accidental overwriting of important files. When trying to overwrite a write-protected file, Windows will deny so and suggest “saving as” in order to allow that a new name is chosen.

This is especially relevant for your raw data files.

Part II

Data management and documentation in EpiData

13 EpiData in short

EpiData is a group of programs dedicated for DMD, grown from an old WHO project called EpiInfo, but with a focus on small and focused functions. The kinship with the old EpiInfo was especially seen in the previous program (EpiData Entry 3.0) with its multi-file structure, but this has been replaced in the new DMD-family, consisting of EpiData Manager (EDM), EpiData Entry Client (EDEC) and EpiData Analysis. They all use, define and fill out only one file with the suffix .epx, that is containing both the structure and the data. I will call the .epx-file the *E-file*.

The programs are public domain and can be used free of charge.

For the purpose of management and documentation we will use the first two programs, EDM is used to set up the data structure and implement control functions, while data keying is the sole purpose of EDEC. The idea is that while the project manager is setting up the entry picture and deciding on data types, labels and documentation in EDM, the persons doing the actual typing in EDEC should not be able to change easily these types of information.

The Analysis part of the EpiData family is for analysing data, but data can easily be exported from EDM for use in other statistical packages (SPSS, STATA and in raw .CSV format), and the use of EpiData Analysis is not covered in this text.

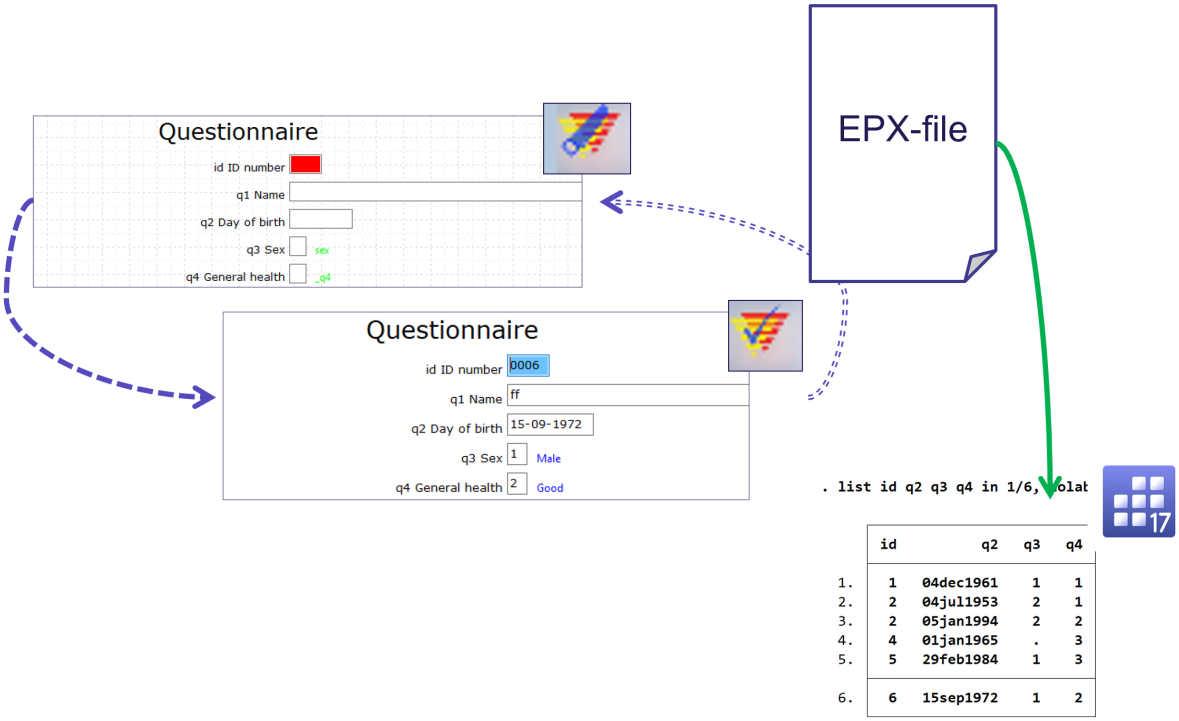
The most important thing to remember is that EpiData Manager (EDM) is used for setting up the database by the person in charge (I call this person the *"manager"*) – EpiData Entry Client (EDEC) is only used for entering data in the *same* E-file – and can be used by several persons (I call them *"keyers"*).

The manager will use both EDM and EDEC in the process of setting up the entry-file to secure that the E-file works as anticipated. The keyer will only use EDEC to fill in data in the E-file.

The programs are not well documented (no manual and only a small number of help-files – for now!).

This reflects that ED is being continuously developed and mostly by people in their spare time, but there is a large community (`http://lists.umanitoba.ca/pipermail/epidata-list/`) offering help, and this section is written in order to partly fill out the need.

ED is being developed to comply with the rules related to Good Clinical Practice (GCP). It means that possibilities for access-control to different parts of the work have recently been implemented. Later more advanced controls of data entry are anticipated, but this depends on funding from the users of ED.



Questionnaire

id ID number

q1 Name

q2 Day of birth

q3 Sex sex

q4 General health _q4

Questionnaire

id ID number 0006

q1 Name ff

q2 Day of birth 15-09-1972

q3 Sex 1 Male

q4 General health 2 Good

```
. list id q2 q3 q4 in 1/6, nolab

     +-----------------------------+
     | id          q2   q3   q4    |
     |-----------------------------|
  1. |  1   04dec1961    1    1    |
  2. |  2   04jul1953    2    1    |
  3. |  2   05jan1994    2    2    |
  4. |  4   01jan1965    .    3    |
  5. |  5   29feb1984    1    3    |
     |-----------------------------|
  6. |  6   15sep1972    1    2    |
     +-----------------------------+
```

Figure 11 The relation of EDM and EDEC to the mutual E-file (.epx file) - and the possibility to export to e.g. STATA

14 Preparing the Database in EpiData

The manager must know how the database should be structured (see chapter 5.5) *before* setting up the E-file. It will also be necessary to have a good idea of the variables and values in the dataset, although some of this can easily be developed during the process.

When the E-file is ready for entry of data, it will deliver a codebook for documentation of structure and checks (see chapter 10).

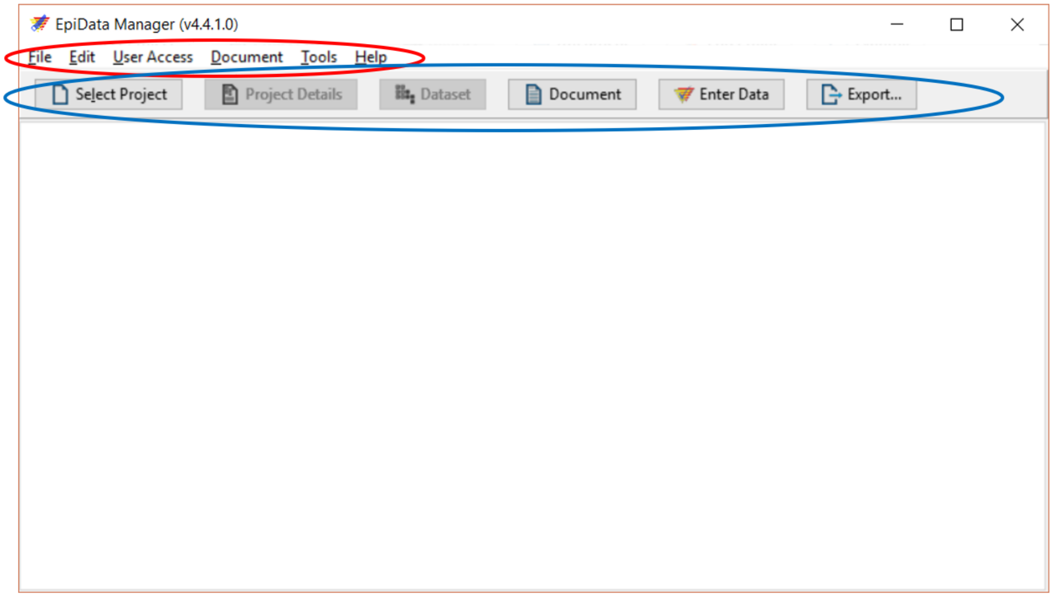
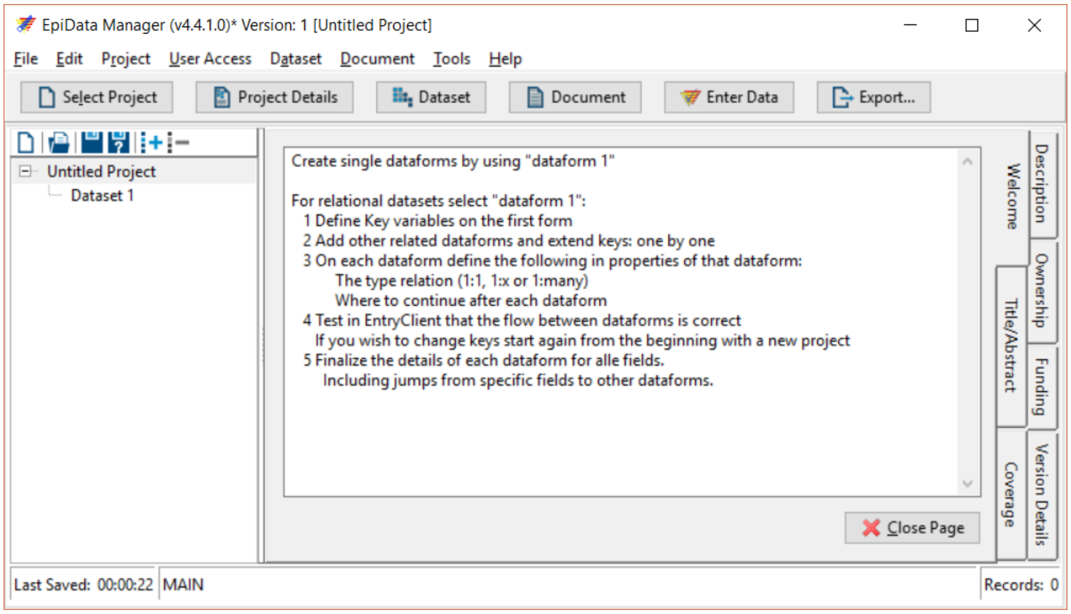

EpiData Manager (v4.4.1.0)
File Edit User Access Document Tools Help
Select Project
Project Details
Dataset
Document
Enter Data
Export...

Figure 12 EpiData Manager without a project. The program possibilities are found in roller blind curtains (red eclipse) and general functions have their own button (blue eclipse)

Opening EDM will produce an empty picture. The program has a lot of possibilities, accessed by using functions in curtains (marked in the figure by the red eclipse), but access to the most common procedures can also be accessed using 6 buttons (blue eclipse).

For setting up a new project use the curtain File>New project or use CTRL+N). An empty "Untitled project" is created (left pane) with a single entry-file ("Dataset 1").

More than one database (*dataset*) can be handled in the same *project*; this would be relevant if several questionnaires for the same person at different times was collected or if both questionnaire data and blood test were collected.



Create single dataforms by using "dataform 1"

For relational datasets select "dataform 1":
1 Define Key variables on the first form
2 Add other related dataforms and extend keys: one by one
3 On each dataform define the following in properties of that dataform:
The type relation (1:1, 1:x or 1:many)
Where to continue after each dataform
4 Test in EntryClient that the flow between dataforms is correct
If you wish to change keys start again from the beginning with a new project
5 Finalize the details of each dataform for alle fields.
Including jumps from specific fields to other dataforms.

Figure 13 New empty project.

The "Untitled project" contains metadata on the project ("data on data"), which should be described and will be included as documentation (see chapter 23). You can leave the metadata empty for now and return later using Project>Study information or the button Project Details to fill out the metadata.

The information “Dataset 1”, which describes the data form, can be renamed to something more informative. Right-click the text and choose “Dataset properties”. In the “basic” tab, the Dataform can be named and labelled, and when choosing OK, the information will be changed.

In the left pane, new Datasets can be created, opened and deleted (using the "+" and "-" buttons). If using the + and – buttons while the “Dataform” is in focus, related data forms are possible, this will, however, assume that a key variable is defined (see chapter 17.1).

You should now save the .epx file using File>Save project (CTRL+S). The program will suggest to save it in your document folder, which might not be the best place (see chapter 12.6). Instead, move it to the relevant folder and give the .epx file a relevant name or set up the EDM (see chapter 24.1).

If you forget to save the E-file, an automatic backup is performed every 10th minute, and the first time it will ask for a name.

At the bottom, a status bar is constantly showing when the E-file was last saved.

Unless you chose to do otherwise (and I would encourage to keep the standard), backup of the E-file will be conducted regularly with the possibility to return to a previous version (see chapter 12.2).

ED has been constructed to minimize the risk of data loss; If data could be lost because of your choice, a box will ask for confirmation – remember to read the alerts carefully!

Hint: Read the messages from ED carefully, they contain relevant information that preserves data!

This also means that if you attempt to close the program with data not saved – you will be warned.

When saved, the name will be shown in the heading part of the upper border of the program

15 Setting up the entry window

With the Dataset defined (see the previous chapter), the entry-form is ready for construction.

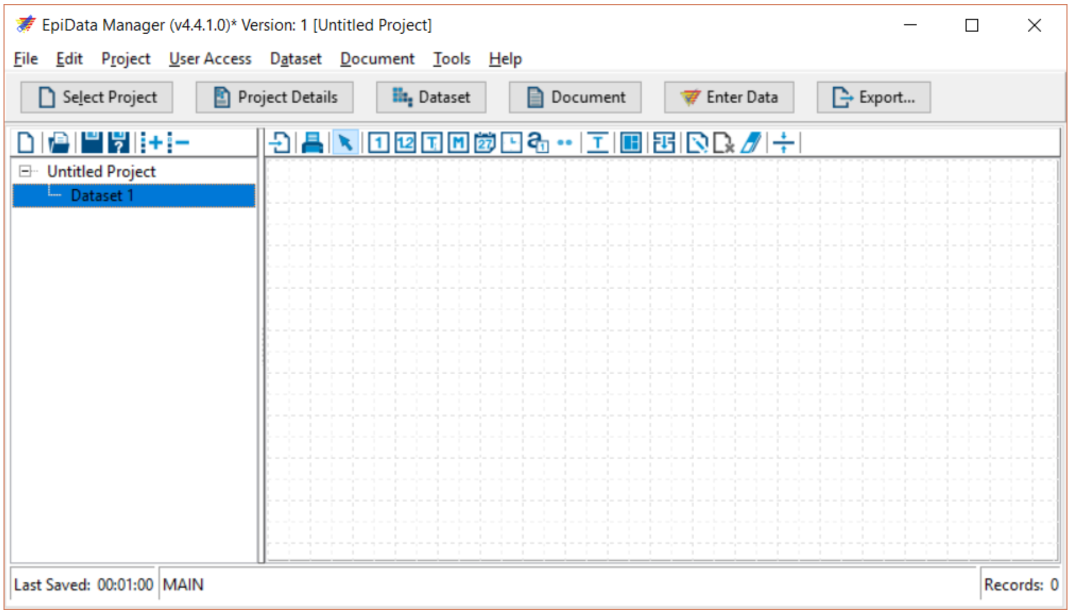


Figure 14 The "canvas" of EDM

Dataset 1" is the canvas (marked as a squared paper) for setting up the variable structure in this database. If the project contains more databases, each will have a canvas of its own.

Hint: "Select Project" – "open recent" holds a list of the last opened projects and gives quick access

Try to mimic the original data form (placement of variables, changes of pages etc.) when setting up the entry canvas to help the keyer in keeping track of the actual variable.

The definition starts with marking the placement of the variable on the canvas and by that defining where the information is to be keyed into the E-file when switching to EDEC.

It is possible to cut-and-paste text directly from the questionnaire in e.g. Word – but it is not advisable, as a lot of defining information is pasted along with the text. Instead, copy the text in Word, paste it into a program like Notepad to get rid of all the extra information – and then copy it again from Notepad and paste it on the canvas. Or better still, just mark the variable and set up the information in the properties box (see later)

15.1 Defining variables

You will need to know whether the information to be held by the variable is numbers, text or dates, that is you will have to "translate" the type of information into the right type of variable (see chapter 5).

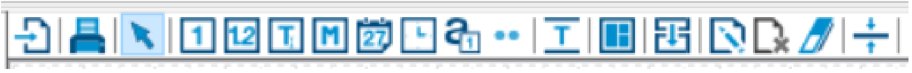


Figure 15 Icons for defining variables - click the desired type of variable and place it on the canvas

In the row of icons above the canvas you will find buttons for setting up the variables, starting right of the marked arrow. They include:

- Integer (0, 1, 2, 3...), marked with "1"
- Float (0.998; 1.43; 2.0...), marked with "1.2"
- String (text; a; A; 1; t2) marked with T, length corresponding to the number of characters.
- Memo (also string, gives a box of several lines to fill in text) marked with M
- Dates, for instance in the form of DMY (day, month, year) marked with a calendar sheet.

  Undocumented: the date variable suppose you mean this month and this year, so entering (in EDEC) "12" in a date variable will read as 12/09/19 if keyed in September 2019, while 12-08 enter 12/08/19

- Time (hours, minutes, seconds) marked with a watch
- Auto-increment variable (Integer-variable, that always increases one for each new record) marked with an "a" and a number. This can be used to insert an identifying number for each variable.
- A lot of others possibilities (in the .. button) – different types of dates (American), autodate (today), autotime (now), a yes-no variable (Boolean; Y/N, y/n, 1/0 will all record as Y or N) and an Uppercase string (that will convert keyed text into CAPITAL LETTERS)

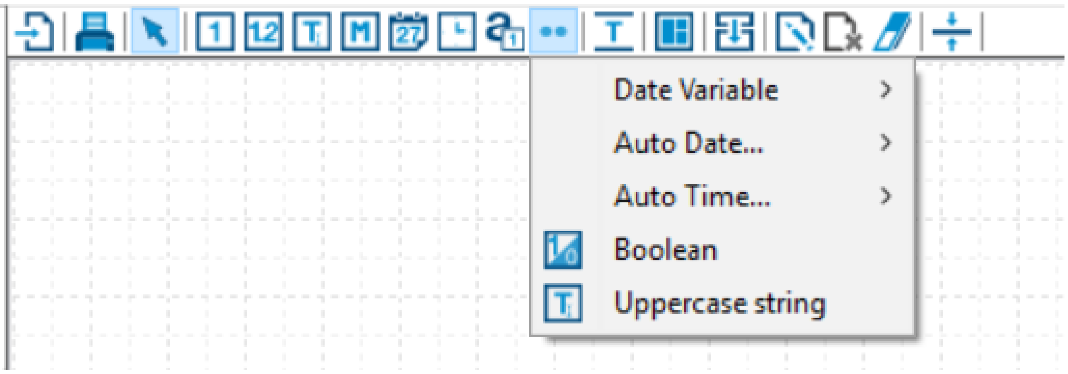


Figure 16 Further variable types for creating variables

The row of icons also includes (to the left of the variable types) a possibility to import data (see chapter 21) and a printer icon to print the contents of the canvas.

To the right, there are icons for aligning and putting more text to the canvas (see chapter 17).

Click the icon representing the type of information and then click the canvas where the variable is supposed to be. It can moved on the canvas later.

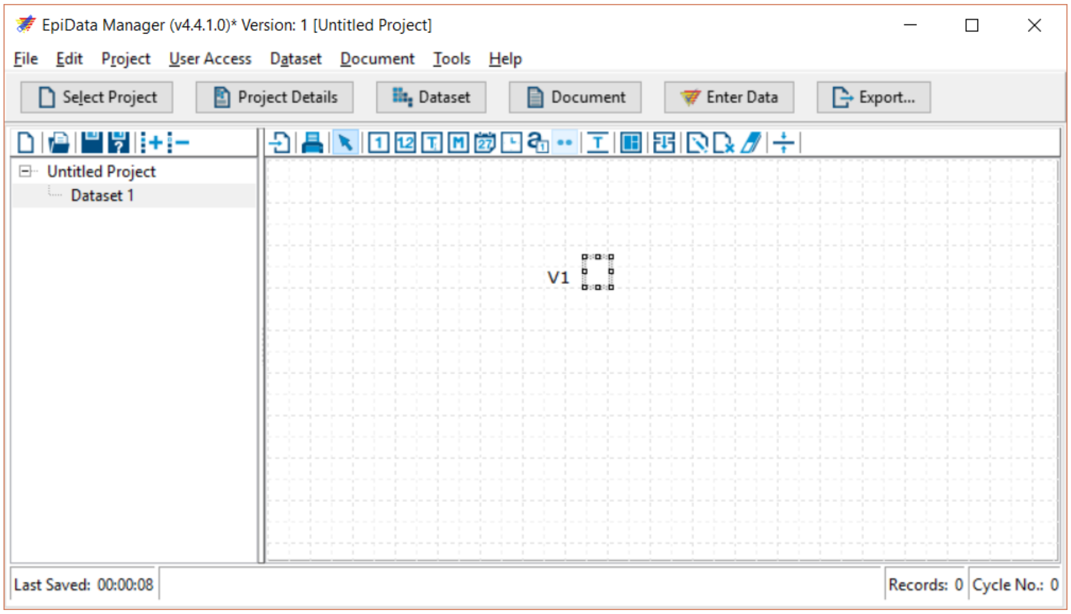


Figure 17 The marking for a new, empty variable with the name v1 (can be changed)

The mark on the canvas (a rectangle with 8 dots) shows the location and the length of the variable, and at the same time the Variable Properties box shows up for including information (properties) to be kept with that variable. The variable is automatically named.

If needed, the variable can be re-located by selecting the variable mark on the canvas with the mouse and moving it.

15.2 The properties box

The box related to each variable, have up to 5 panes depending on the variable type used (the type is stated in the upper right corner and can't be changed without deleting the variable-marker, by that losing the keyed information).

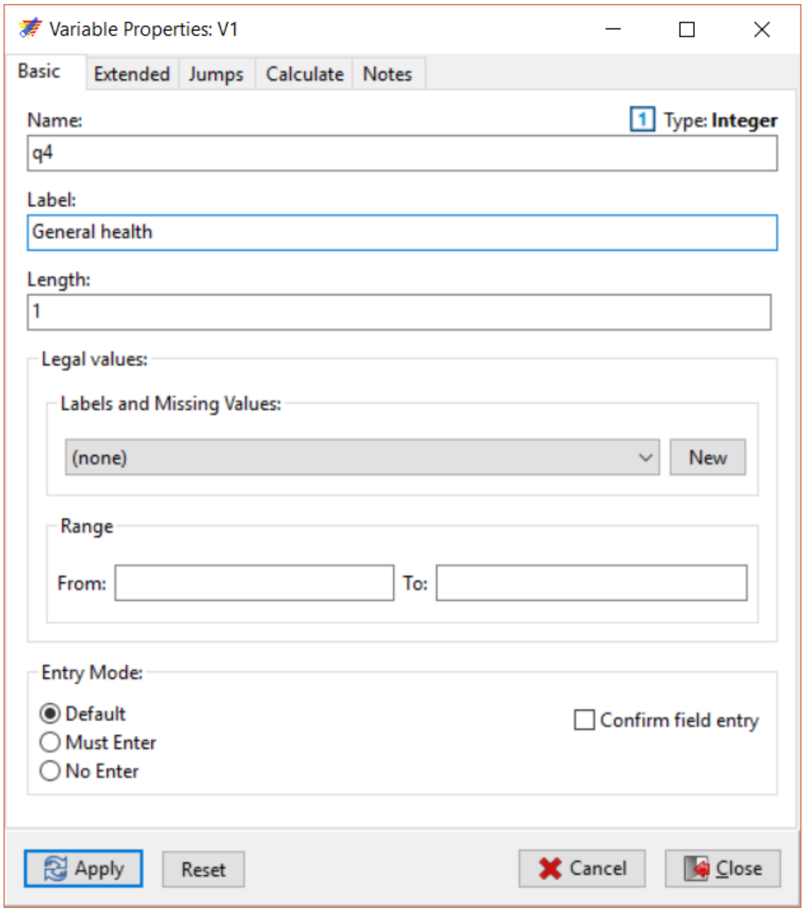
> Hint: think carefully on the type of variable, that is to hold the information in the E-file as it can't be changed afterwards

The name of the variable is predefined in accordance with the choices made in relation to the setup of the program (see chapter 24) but can be changed in the box.

15.2.1 Basic information on the variable

In all types of variables, the first pane in the box includes basic information on the variable.

The name of the variable ("**Name**") must start with a letter. As previous suggested (see chapter 5.2), use a short name that also represents the information structure (for instance q1 for the first variable on the questionnaire, q2 for the second, j1 for the first variable from a patient record etc.) and then assign a label that includes more information on the question to help identify it later on.



Name: q4

Type: Integer

Label: General health

Length: 1

Labels and Missing Values: (none)

Range

From:

To:

Entry Mode:

● Default

○ Must Enter

○ No Enter

□ Confirm field entry

Figure 18 Variable Properties box for a Integer variable. Note that the box is labelled as properties for V1, which is the name given by EDM - this will change to the new "Name:" q4, when hitting Apply or Close.

The "**Label**" holds the variable label, it can be long, but as it might be used in tables, figures etc., it should be kept reasonably short, mimicking the information that it holds (chapter 5.2). I seldom exceed 25 characters and by that often at the same time condense the information. As you will have the original information in your documentation, it need not to be accurate, and it will not give long tables when using the information later on.

The ”**Length**” of the variable should correspond to the amount of information necessary, a variable length in a numeric variable of length "1" can handle values 0-9; if one needs to key in “16” in the variable, the length should be "2" (as this will allow representation of 00-99) and so on.

In strings (text variables), the length should correspond to the number of characters allowed in the questionnaire.

The variables will, according to the setting up of the Variable Properties (see below), identify at the canvas as the variable name (id, q1, q2, q3…) and the variable label (ID number, Name, Day of birth and sex) for the first 4 variables of the questionnaire (Figure 20).

Using more of the possibilities in the box will show more information on the canvas.

In the "Legal values:" box, two different choices will define what values can be written in the variable.

The "**Labels and Missing Values**" curtain allows the possibility to reuse labels for values (see below)

"**Range**" define the possible legal values, so that numeric values from a certain value through all higher values until the number stated in "to" are accepted at entry, while values outside are not accepted.

"Label and Missing Values" can be used together with "Range" in order to define, for instance, that 1 to 5 stated in "Range" and defining 9 as a value label in the curtain, will give the possibility to record 1, 2, 3, 4, 5 and 9, but not 0, 6, 7 and 8.

"**Entry mode**" is used to define whether a variable must be filled in ("**Must enter**"), which is not default. A "must enter" might be considered if it is an ID-variable. Although we naturally want all information supplied, it might not always be present and it will halt work, as the keyer is not able to proceed unless giving a valid information. So it should be carefully considered whether the information should be demanded.

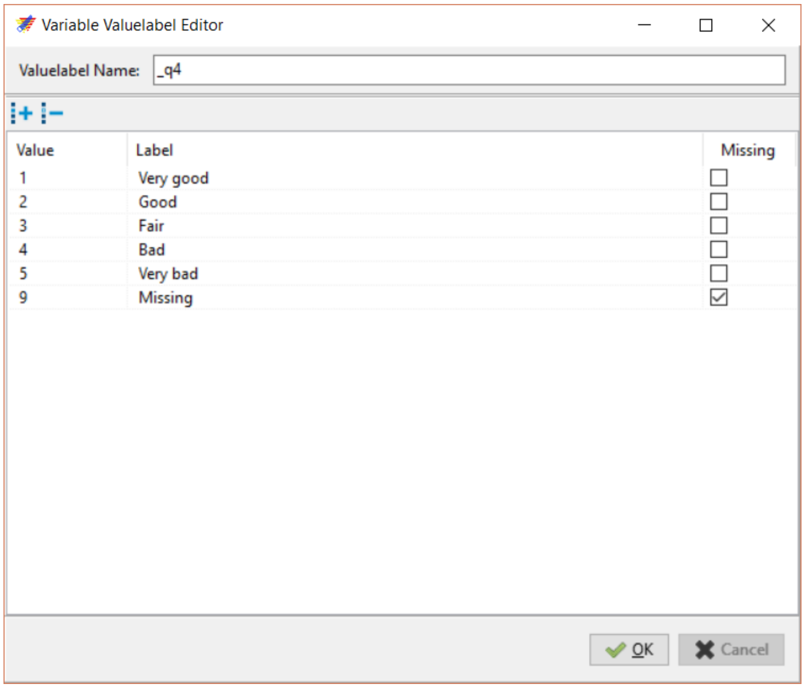
In the EDM and EDEC variables in the E-file marked as "must enter" are shown in red.

A "**No Enter**" variable can be used to present calculations during entry that the keyer will not be able to access.

"**Confirm field entry**" is used to signal that the keying person must hit <Enter> to proceed. It will slow down work, so this setting should also be used with caution.

15.2.2 Defining value labels

To set up a new set of value labels, click the "new" button to the right of "Labels and Missing Values" in the Variable Properties box. The corresponding new value label is by default named in a way that correspondent to the variable (here _q4 as the variable is named q4), but the name can be changed as the label set does not need to be named as the corresponding variable, and could, for instance, be defined for use with many different variables sharing the same set of answers. If a value set of “yes”, “no” and “don’t know” was used several times in the questionnaire, they could be defined once and reused for other variables by choosing the value label name for all those variables. In that case, a more relevant name for the value label set would be “yesno” – to be able to easily find it in a long list of defined value label set.



Valuelabel Name: _q4

Value	Label	Missing
1	Very good	□
2	Good	□
3	Fair	□
4	Bad	□
5	Very bad	□
9	Missing	☑

Figure 19 Variable valuelabel editor

For each value in a nominal or ordinal variable, a label should be defined and labeled.

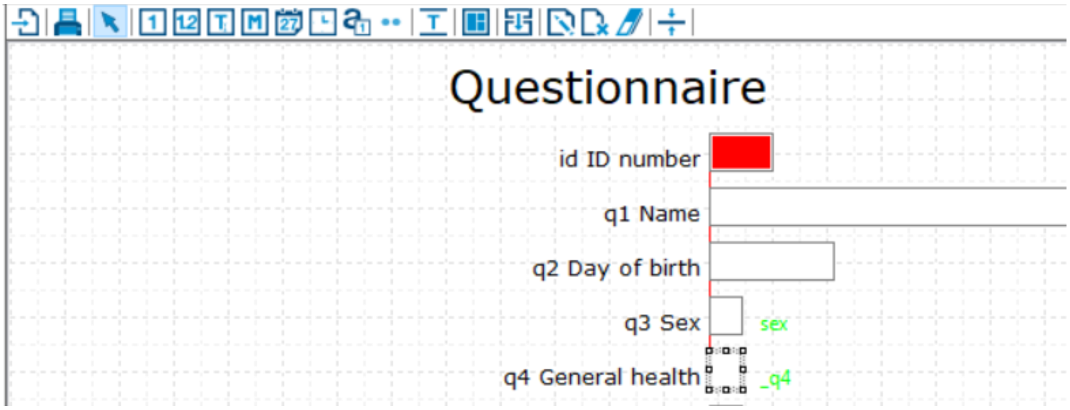
The value and corresponding label are added by clicking the blue "+"- mark and after checking the value (numbering starts with 1) the corresponding label is written. A <return> should be hit, by that EDM understands that the label is defined and the “+” can be clicked again to enter the next label. It is not possible to avoid using the + (or -) signs.

The suggested number under “Value” can be changed if necessary, for instance when defining value 9 after defining value 5.

Define also at least one missing value and mark it with a check-mark in the “Missing” column.

When the value label set has been finished with OK, its name can be found in the "Labels and Missing Values" curtain and by that reused for other variables.

EDM can import longer external label list for using with the keyed values (consider a list of codes and corresponding text for ICD-10-coding).



Questionnaire

id ID number

q1 Name

q2 Day of birth

q3 Sex sex

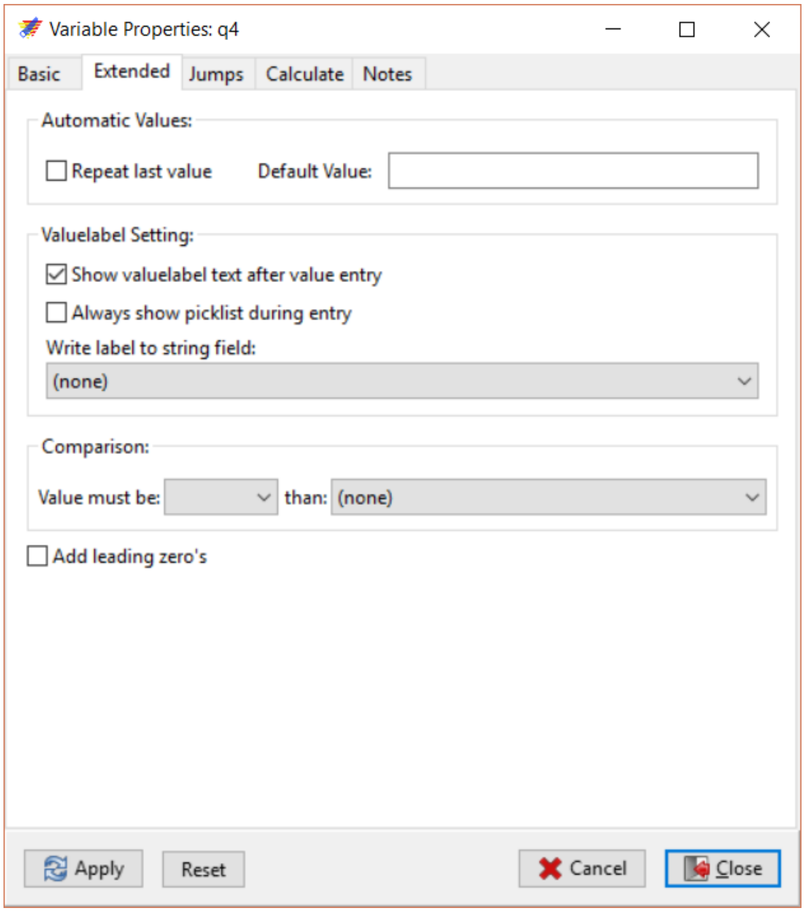
q4 General health _q4

Figure 20 The defined variables for the first 5 variables

Having defined value labels and assigning them to the variable, the name of the value label shows up in green to the right of the variable mark on the canvas. ID is now marked in red, as it has been defined as a "must enter" variable.

15.2.3 Extended pane

In the next pane of the Variable Properties box, more sophisticated possibilities are collected.



Automatic Values:

☐ Repeat last value Default Value:

Valuelabel Setting:

☑ Show valuelabel text after value entry

☐ Always show picklist during entry

Write label to string field: (none)

Comparison:

Value must be: than: (none)

☐ Add leading zero's

Figure 21 Possible choices in the "Extended" pane

If the same value is anticipated in most records it is possible to mark that either the value is **repeated** from the previous record – or to put a **default value** in the variable, that can be changed by the keyer if necessary.

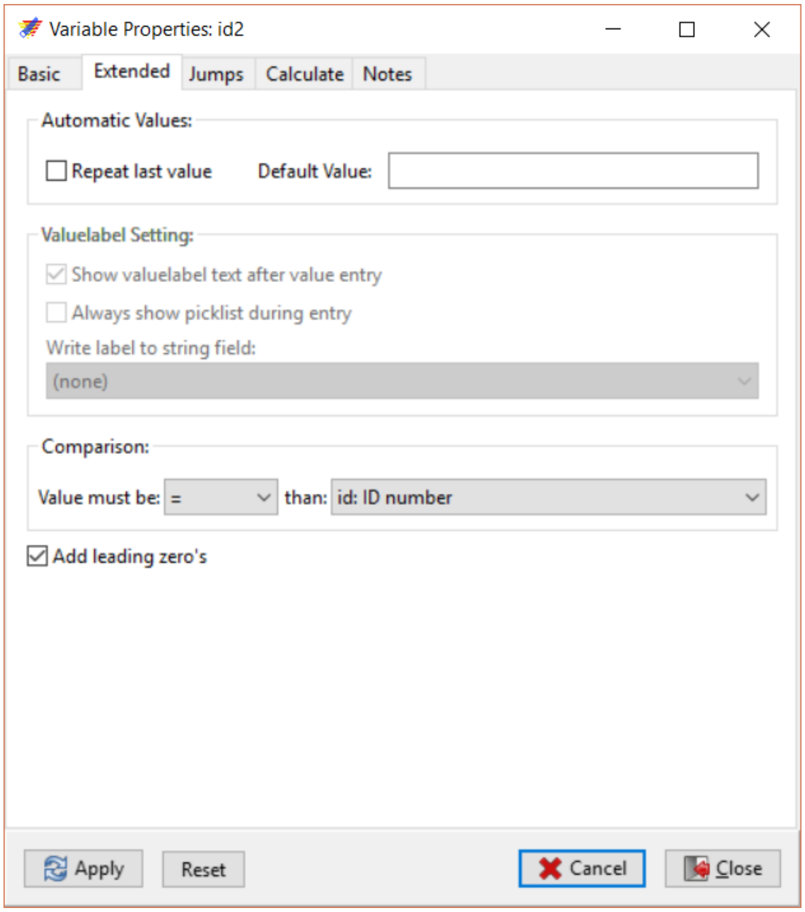
It would come handy if for instance having a “Date of filling out” in the questionnaire, and most of the records were filled out the same date. Instead of keying the information for each of the records, it could be repeated and thus show the same value. The keyer would then just have to hit <Enter> to verify this value.

The section related to "Valuelabel setting" is described in the subchapter Setting up help for the keyer (15.2.4) below.

In the section “**Comparison:**” the content of this variable can be compared with another numeric variable in the dataset, that has already been defined and thus is present on the curtain list in the right column.

The comparison is made when leaving the variable in focus, so normally it would have to compare with a variable already filled – and thus the comparison should only be made in the last of the variables being compared. Comparisons may both evaluate whether variables are equal or different from each other.

It could be used to ensure that two different keying of an ID variable (first and last in each record for instance) are the same, and it won't allow the keyer to continue until the comparisons are fulfilled.



Automatic Values:

□ Repeat last value Default Value:

Valuelabel Setting:

☑ Show valuelabel text after value entry

□ Always show picklist during entry

Write label to string field: (none)

Comparison:

Value must be: = than: id: ID number

☑ Add leading zero's

Figure 22 Comparisons of values for variable "ID2" with that of a previous existing variable "ID"

This should be considered as a record with a wrong ID is nearly impossible to find again, so by forcing the keyer to be sure that the ID-number is correct will prevent this type of fault.

In the bottom of the pane, it is possible to mark a field to be filled with **leading zeros** even if it is a numeric variable (for instance to give an ID-number the value of 0001)

15.2.4 Setting up help for the keyer

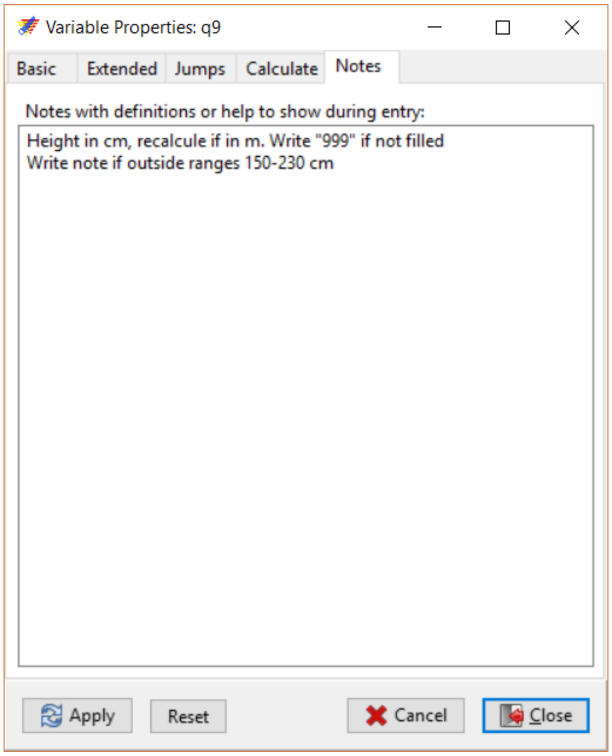
In the middle of the pane, with the heading "**Valuelabel Setting:**", it is possible to give feed back to the keyer in response to the choices made.

These possibilities will work only when value labels have been defined under the Basic pane. If selecting "**Show valuelabel...**", the corresponding label is shown just right to the variable during keying so that keying "4" will make EDEC present the corresponding value label "Bad" alongside the keyed number.

It is possible to present the value label list defined in the "Basic" pane to the keyer automatically by marking "**Always show picklist...**" and let the keyer either hit the number or use the mouse to choose the value for the variable. It might slow down the keyers work, though, so it should normally not be used for all variables.

Finally, it is possible to put the value *label* into a variable of its own (a string variable must be defined before this is possible, it will then be present in the curtain). It could be relevant if for instance having a set of postal codes and corresponding names of towns as a value label list, and letting the keyer entering just the postal code and making EDEC registering the town name automatically in a no-enter text variable. This would both speed up the keying process (not having to key in the city name), securing correspondence between postal code and name of the city and letting the keyer assess whether the two correspond.

Further information can be presented to the keyer by writing this in the last "Notes" pane in the Variable Properties box. The information written here will always be shown when the variable is in focus, stating for instance how to record missing, that the keyer is to remember that the height is to be recorded as nearest cm or that the keyer should make a note if something unusual is anticipated to happen.

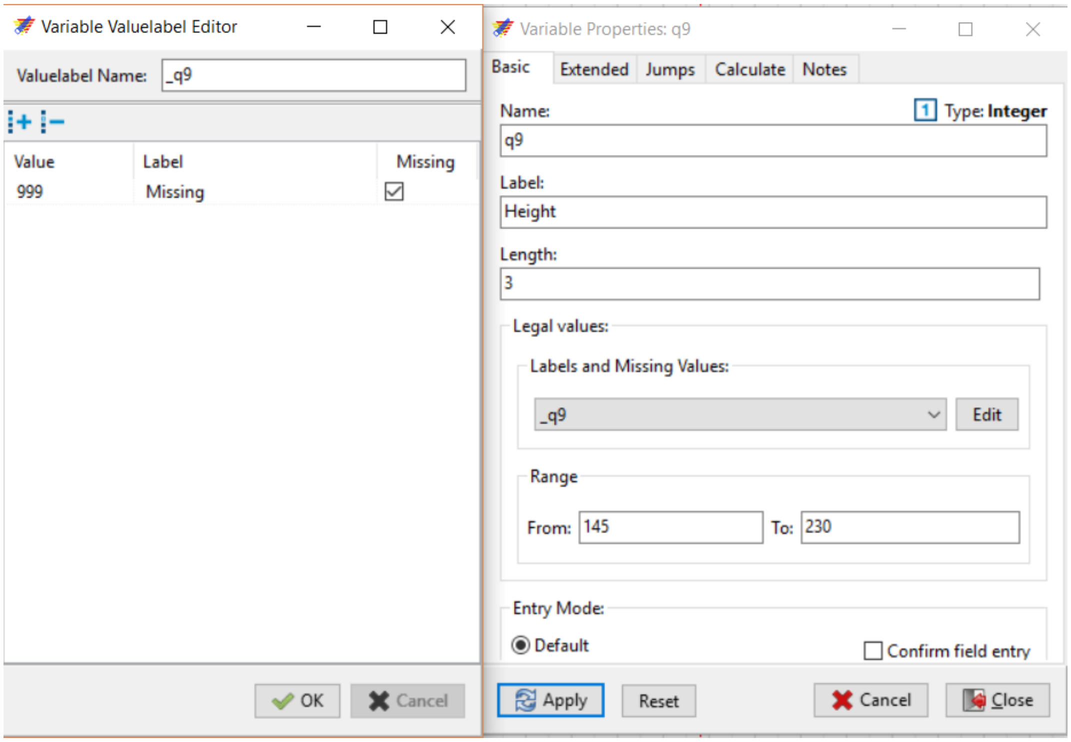


Notes with definitions or help to show during entry:

Height in cm, recalcule if in m. Write "999" if not filled
Write note if outside ranges 150-230 cm

Figure 23 Information in the "notes" pane will be presented to the keyer when the variable comes in focus

In the figure below, an example of the use of both definition of range and a value label for missing values will allow values from 145-230 and 999 to be coded. If further writing a note as in the figure above, trying to type a wrong number in the variable will result in the response given in figure 25. I purposeful wrote 150-230 in the note to show where the note information goes; this should of cause correspond to the range given.



Valuelabel Name: _q9

Value	Label	Missing
999	Missing	☑

Name: q9

Type: Integer

Label: Height

Length: 3

Legal values:

Labels and Missing Values: _q9

Range

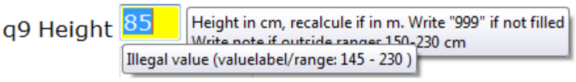
From: 145 To: 230

Entry Mode:

● Default

□ Confirm field entry

Figure 24 Setting up range and valuel abel for a variable



q9 Height 85

Height in cm, recalcule if in m. Write "999" if not filled
Write note if outside range: 150-230 cm

Illegal value (valuelabel/range: 145 - 230)

Figure 25 Response to the keyer if trying to key a wrong number. The information from the "note" is shown to the right (remark 150-230), the values in "range" (see above) is shown hoovering above the field (145-230)

The field turns yellow to indicate that the keyer will not be able to move away from the variable until not violating the information any more.

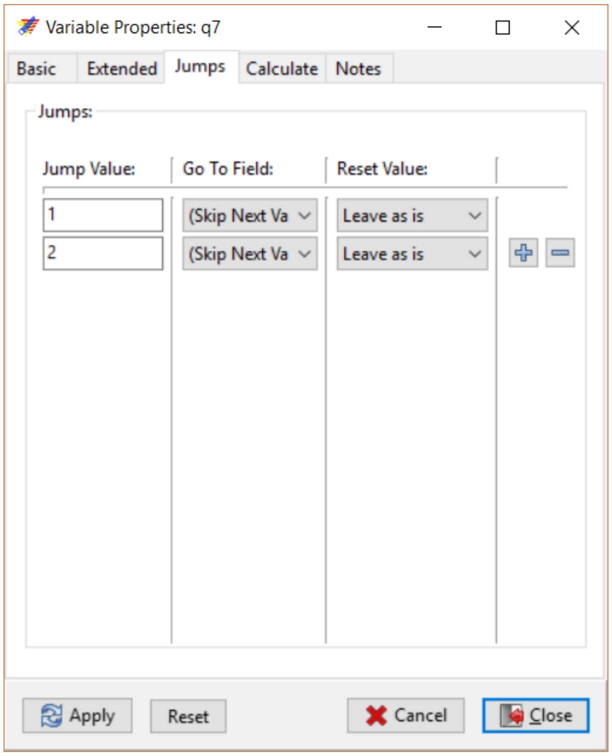
If the respondent was actually 235 cm high, the keyer will not be able to proceed, instead the keyer should be instructed to write 999 for missing (or another accepted code) and at the same time making a note, so that the manager at a later time will be able to change this record (either changing the range setting of the variable to allow for the value or preferable to change it during later management in the statistical program and at the same time documenting the change with the inclusion of information on the situation).

15.2.5 Structuring the keying using jumps

Often the information in an entry file will not be completely linear but may branch depending on the values keyed in. It might be defined in the questionnaire (if you mark being "male" then skip the following questions on pregnancy and move on to question 22) or result from the data structure (if for instance, something is missing).

EDM can define how keying different values move to a field not situated immediate after the actual field, be directed to a different sections, or to enforce actions in response to the value entered.

It is first possible to define jumps when the variables have been defined and so this will normally be defined after setting up the entire canvas. When the variables have been defined the third pane of the Variable Properties gives access to define jumps to all variables (Fields). Jumping definitions are established by clicking the "+", giving the value to be evaluated ("Jump Value:") and what to be jumped to if the variable contains this specific value ("Go to field" - for instance skipping a field or pointing to a specific field). Possibilities also include leaving the section (pages) or record (in that instance saving the record and open a new empty one).



Jump Value:	Go To Field:	Reset Value:
1	(Skip Next Va	Leave as is
2	(Skip Next Va	Leave as is

Figure 26 Jump properties for the variable

More jumps are possible for the same variable, in principle one per value defined, but it is normally just a few values, that should be defined, as those values that are not defined will just proceed to the next variable.

With the "Reset value" possibility, the variable in focus can be set to missing values.

15.2.6 Calculate information and other derived information

In the fourth pane the possibility to calculate the value of another variable ("Result") that is already defined is given. This is often used in combination with a "No entry" value in the result variable (marking in the basic pane, so it won't be changed by the keyer). Calculation possibilities include time or date differences, or the possibility to fill string variables with a combination of other variables.

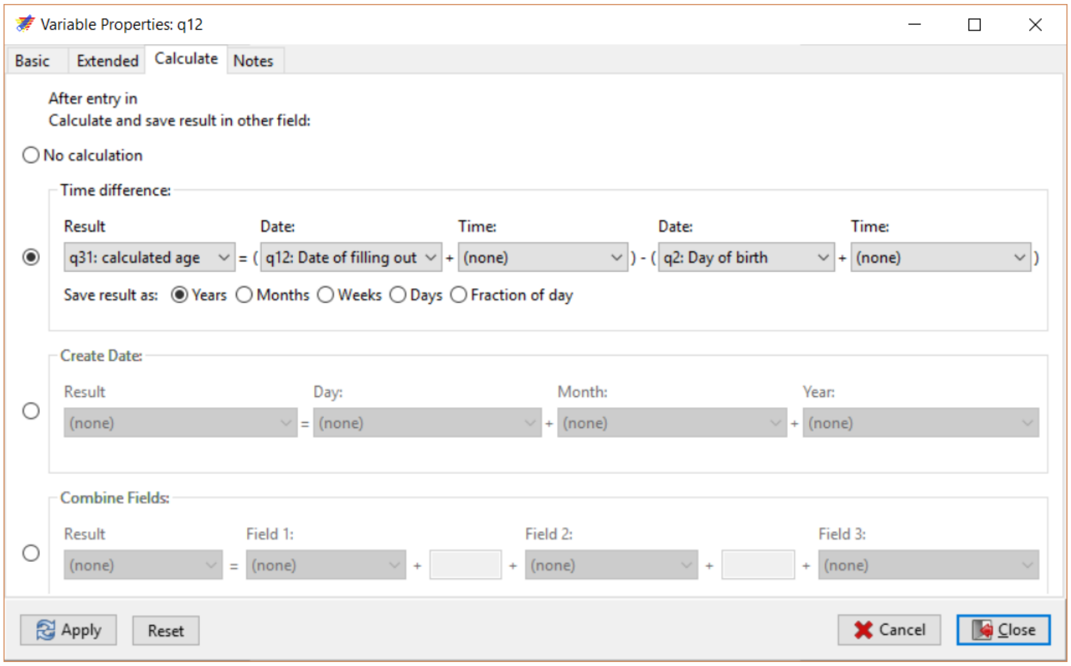
Time difference could be used to calculate age based on birth date and today's date or to calculate the time between different visits at the outpatient clinic. While this is often done later in the statistical program, it could be relevant to inform the keyer to make sure that the information provided in the two variables are correct.

Create date could be used if keying day and month in two variables, a third variable (marked as "no enter") might include the year and this should then be put into a date variable (also marked "no enter"). The defined variables are part of the dataset and will be exported to the statistical program.

Calculations should be made in the last variable entered and all variables should be defined before the calculation are being set up in the variable properties.

In this example, calculations are defined in the "date of filling out the questionnaire", and in a defined variable (q31, that is a non-enter field) the result of a subtraction of date of birth (q2) from date of filling out (q12) is placed and defined as being in years. The calculation is made when the q12 field is left.

Anticipate more calculation possibilities in future versions of ED.



After entry in
Calculate and save result in other field:

○ No calculation

● Time difference:

Result		Date:		Time:		Date:		Time:	
q31: calculated age	= (	q12: Date of filling out	+	(none)	) - (	q2: Day of birth	+	(none)	)

Save result as: ● Years ○ Months ○ Weeks ○ Days ○ Fraction of day

○ Create Date:

Result		Day:		Month:		Year:
(none)	=	(none)	+	(none)	+	(none)

○ Combine Fields:

Result		Field 1:				Field 2:				Field 3:
(none)	=	(none)	+		+	(none)	+		+	(none)

Figure 27 Calculation pane with example of calculation

The last pane ("Notes") is used to present information to the keyer using EDEC (see the previous section). The note-pane can be used to inform the keyer how to handle for instance missing values or stressing a certain way of coding information.

15.3 Information on the canvas in EDM

When the Variable Properties of the variable have been filled out and Apply have been hit, the canvas will now include information relating to the variable that all comes from the Properties box.

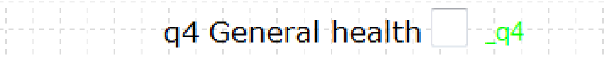


Figure 28 Definitions from the Variable Properties shown at the canvas. q4 is only shown if changing the set up of EDM.

Text at the left of the variable marker is the variable label (from the "Basic" pane), if a Value label has been assigned, the name of the Value label is shown in green to the right of the variable, which makes it rather easy to secure that all relevant variables have been assigned both variable-

and value labels. This is, however, not default but should be defined in the setting up of EDM (chapter 24.)

The information relating to the variable can be changed by double clicking the 8-point Variable marker – or the variable marker can be deleted by selecting the 8-point mark and hitting <Delete>.

The variable marker can be copied, by that also copying all information from the Variable Properties (but with a new variable name). By that, after having used time for setting up a variable with all relevant information, it can easily be re-used for a similar variable, for instance, if several questions in the questionnaire have the same structure with the same possibilities for answers and the same value labels.

In this example, I've used <CTRL-C><CTRL-V> to copy the first variable after it was finished (with the relevant _filled value label) and then just change the variable name and variable label.

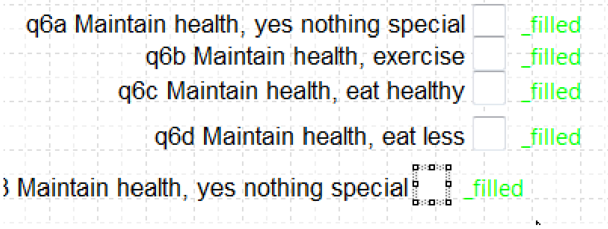

q6a Maintain health, yes nothing special
_filled
q6b Maintain health, exercise
_filled
q6c Maintain health, eat healthy
_filled
q6d Maintain health, eat less
_filled
3 Maintain health, yes nothing special
_filled

Figure 29 Copying variables on the canvas

16 Checking and changing the data structure

When the entry fields have been sketched on the canvas it is normally necessary to control that the E-file will work as planned.

This is done by closing (and saving) the E-file in EDM and opening it again with EDEC. In the Windows version, this can be done directly by clicking the "Enter Data..." button. This will close EDM and open EDEC while securing that the data have been saved properly.

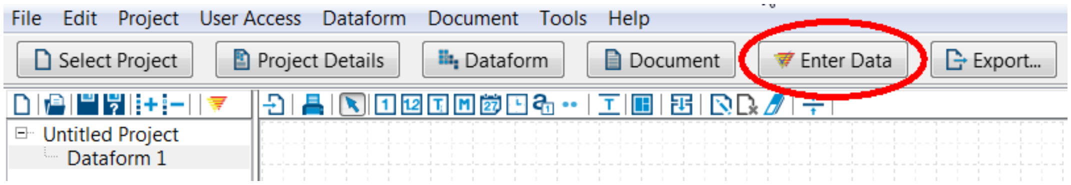


Figure 30 Using the "Enter Data..." button to close EDM and open EDEC

In the Mac-version, this is not working at the moment, so you have to save the E-file, close EDM, open EDEC and read the E-file in this program. The E-file cannot be open in both programs simultaneously.

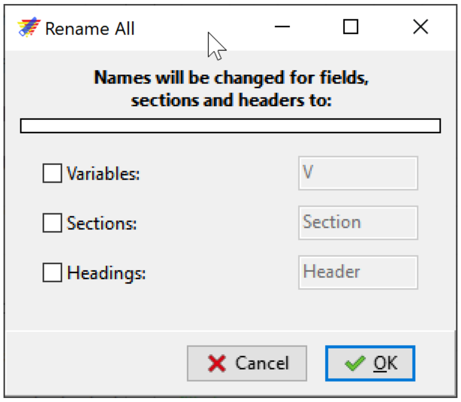
When testing in EDEC is over (see chapter 18 for a description of working in EDEC), the E-file should be closed again (eventually without saving data while testing) and EDM opened again after which the E-file should be loaded. There is no button for this, but use the Recent Project to open the E-file from EDM.

If problems have been identified, most changes can easily be made when back in EDM.

When data have been keyed into EDEC and saved (both during testing and after going live), changing the variables in EDM will prompt ED to try to save as much data as possible, and warn the user if changes will result in loss of data.

Nevertheless, although making changes to the structure are possible, they should be kept to an absolute minimum after going live and beginning to key in real data. Better test the E-file thorough on a small set of test data (the first 10 records for instance) and when the testing is finished, delete all records and start all over to ensure that data are keyed properly.

Smaller changes might be accomplished using the Variable Properties. More thorough changes in the data structure with renaming variables can be accomplished using the Dataform -> Rename Variables option.



Names will be changed for fields, sections and headers to:

☐ Variables: V

☐ Sections: Section

☐ Headings: Header

Figure 31 Dataform - Rename variables, headings and sections

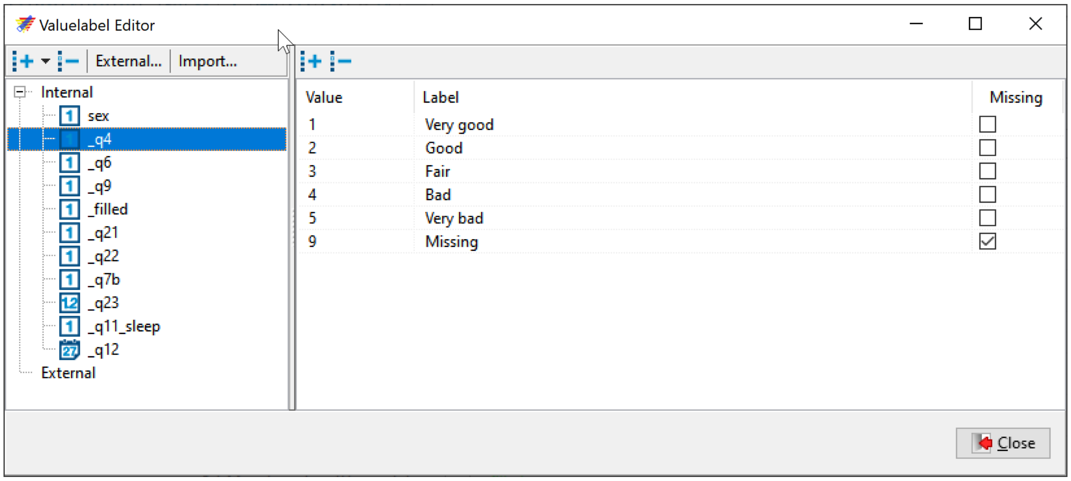
This gives the possibility to give all variables a new ”prefix” (e.g. ”q” instead of ”V”) and it will have an impact on all variables, which will also be named with consecutive numbers. Nearly all definitions (including jumps) will be preserved (unless variables are deleted).

Likewise, it is possible to give new names for sections and headings (see next chapter).

It is not possible to re-do the changes, and the possibility should be used with caution, in my example, for instance, I would loose the naming of the ID-variable, that would be V1 or q1 and the relationship with the numbering in the questionnaire would be lose.

16.1 Changing or editing value labels

Value labels will most often be changed using the Variable Properties pane, but it is also possible to make changes in the Valuelabel editor either by right clicking project or use the Project curtain and, in both instances, chose Valuelabel editor. The editor shows the existing value labels with their name in the left side and the content in the right side window (when clicking the name on the left)



Value	Label	Missing
1	Very good	☐
2	Good	☐
3	Fair	☐
4	Bad	☐
5	Very bad	☐
9	Missing	☑

Figure 32 Figuren skal rettes til

It is possible to remove value labels (by using the blue minus mark (-) at the left side) or to set up new value label sets by using the (+) sign. It is necessary to determine whether the variables for which the value is to be used are integer or string variable – but it is not necessary that the variable in itself is defined, as the value labels are not related directly to one variable, but can be used on several variables. The name of the value label can be changed by right-clicking the new name and in the right-side window values and labels can be defined.

Existing value labels can be selected and edited (right window).

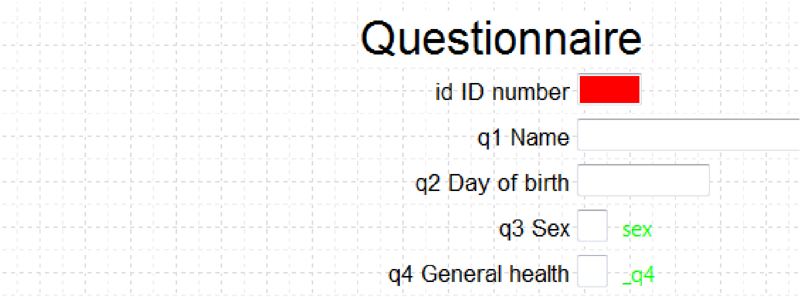
Using the External possibility, value labels from other E-files can be found and re-used. It is also possible to import value labels from external sources, for instances postal numbers and town names or ICD-10 codes with the corresponding disease names.

The file with the number and names can often be found on the Internet and should be in the form of an epx-file, a STATA dta-file, a comma-separated file (.csv) or a simple non-formatted txt-file.

17 Finishing the entry form

When the structure of the entry file is determined and is working as planned, the canvas can be tidied up and more information included. This is just for helping the keyer, the computer doesn't care.

17.1 Sections and headings



Questionnaire

id ID number

q1 Name

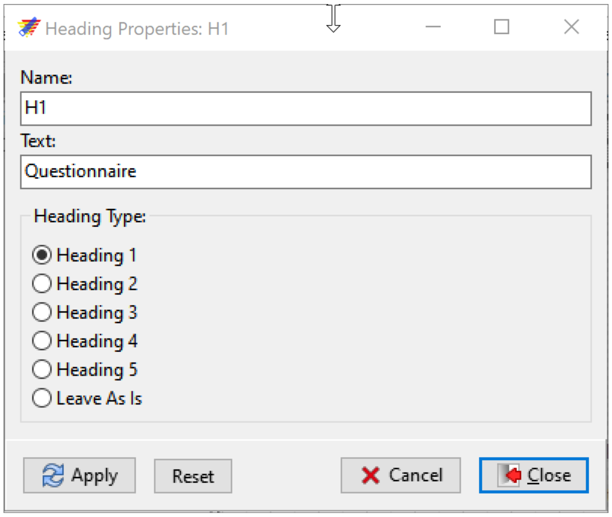
q2 Day of birth

q3 Sex sex

q4 General health _q4

Figure 33 The text "Questionnaire" is defined as a "heading"

It is possible to include text, that is not related to variables – these are defined using the "Headings" ("T" button). This could be used to mimic some of the helping text from the questionnaire to help the keyer navigate or to include helping text on the canvas.



Heading Properties: H1

Name: H1

Text: Questionnaire

Heading Type:

● Heading 1
○ Heading 2
○ Heading 3
○ Heading 4
○ Heading 5
○ Leave As Is

Figure 34 Definition of headings in the questionnaire

In the "Heading Properties", the heading is given a name (H1), in the “Text:” section the information to be put on the canvas is written, and the "Heading Type" is solely defining the size of the text.

The size of the headings chosen by "type" can be defined in the preferences (see chapter 24)

It is likewise possible to define " Sections" during the building of the form – this might correspond to different pages in the questionnaire, they are inserted using the button just right of "T" that is used to mark an area of the canvas. They should be defined *before* setting up the variables - or the variables can be cut and pasted into the area afterwards.

Besides for mimicing the lay out of a questionnaire, sections can be used to help navigating, as "Jumping" can "leave section"

Having set up the variable 8-point markers often means that they are not situated exactly below each other. Using the buttons marked with an "I" makes it possible to align variables and text.

Using the “Sides:” buttons make the variables align under each other, and afterwards their distance can also be aligned. Alignments can be invoked at any time and can be used also at only some of the variables if they are marked (using <CTRL>-click with the mouse or holding the right mouse-button while selecting). To mark all variables, use <CTRL><A>.

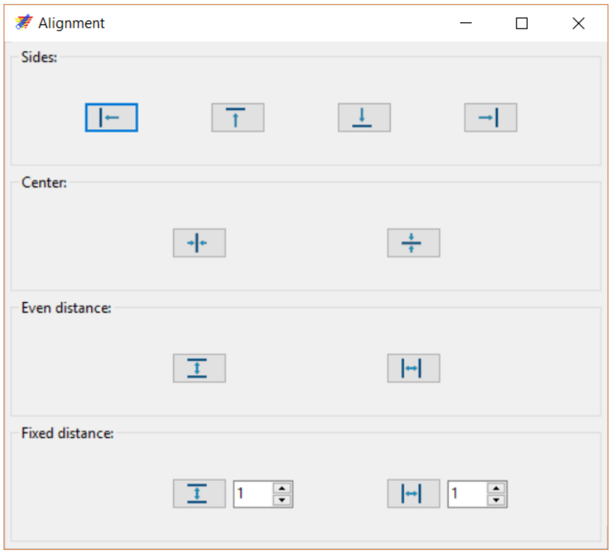
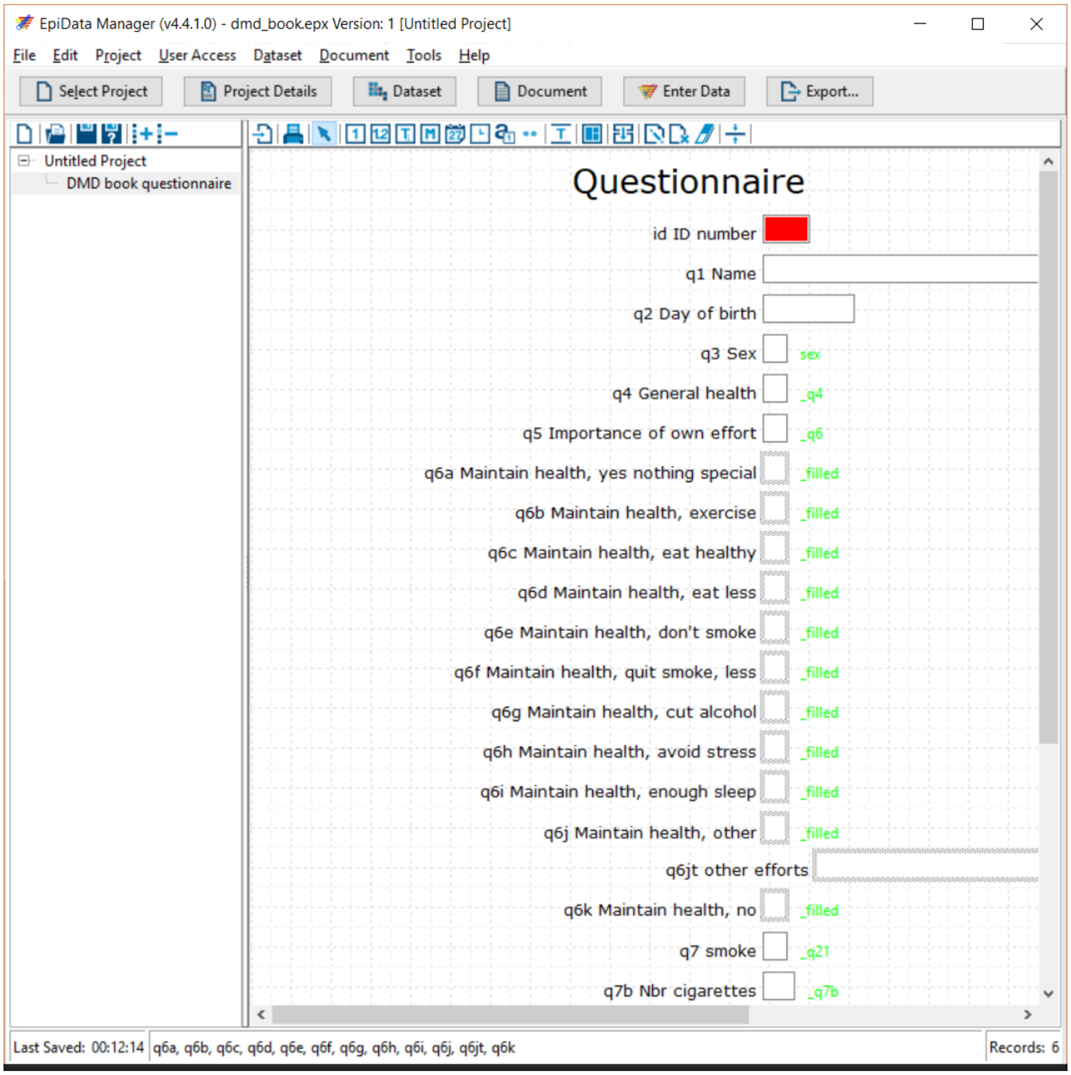


Figure 35 Possibilities for aligning variables on the canvas

At the canvas selected variable markers for adjusting are shown slightly darker. In the bottom part of the window, EDM shows the actual variables marked (if several have been marked they are shown in sequence).

The size of the canvas is one page - it can be prolonged by using the button Extend dataform (situated third to the right of the dots) or <CTRL> + <ENTER>, which will both append one page below the last page. Mark and move variables as needed.

Besides that, the bottom shows the number of actual records in the E-file at the right (entering will have been conducted in EDEC). It is also showing the time elapsed since last saving of the E-file.



Questionnaire

id ID number

q1 Name

q2 Day of birth

q3 Sex _sex

q4 General health _q4

q5 Importance of own effort _q6

q6a Maintain health, yes nothing special _filled

q6b Maintain health, exercise _filled

q6c Maintain health, eat healthy _filled

q6d Maintain health, eat less _filled

q6e Maintain health, don't smoke _filled

q6f Maintain health, quit smoke, less _filled

q6g Maintain health, cut alcohol _filled

q6h Maintain health, avoid stress _filled

q6i Maintain health, enough sleep _filled

q6j Maintain health, other _filled

q6jt other efforts

q6k Maintain health, no _filled

q7 smoke _q21

q7b Nbr cigarettes _q7b

Last Saved: 00:12:14 | q6a, q6b, q6c, q6d, q6e, q6f, q6g, q6h, q6i, q6j, q6jt, q6k | Records: 6

Figure 36 Having marked several variables on the canvas, EDM shows their names

Once again, the manager will have to control the setup by opening the E-file in EDEC, both to control the layout, but especially to try how the form will be working when trying to input data (see chapter 18).

The entry form can't be changed in EDEC.

17.2 Variable, value and data labels

At this point make sure that every variable and set of values are properly labelled by looking at the canvas. You should also consider setting up a dataset label by choosing the project pane and filling out the information possibilities marked at the right side of the page. This information is transferred to both SPSS and STATA, but will also be helpful if later working on the E-files.

17.3 Deciding whether to use the security settings

If necessary, you can implement security settings for the project, but it is not necessary. You should start by defining groups with different roles in relation to access and rights. ED has an administrator group with all possible rights, and it is possible to define groups having for instance rights to change part of the questionnaire or rights to enter data only.

In order to implement the use, a user must be given administrator rights (User Access > Extended Access > Add User/Group Administration), then the project must be saved and re-opened in order to the security setting to take effect.

Then groups should be defined (Define Groups) and given the appropriate rights.

Each administrator and keyer will be defined as a user with login name and password and is added to a relevant group.

Finally, the data form will be related to the groups (Define Entry Rights), allowing the possibility to control access to read, update, create and delete entries.

It is also possible to handle passwords, for instance by defining a date for expiration.

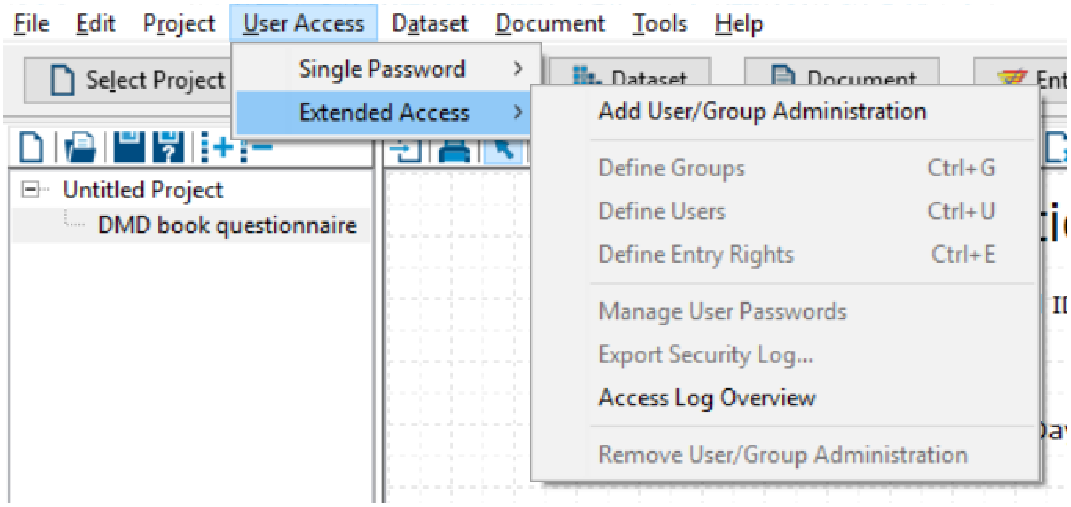


Figure 37 Setting security settings

The access control can be removed again for the project (by the Administrator).

For more information on the security setting, see “EpiData User and Group Administration” in the help-menu (Help -> Tutorials (Local) -> ExtendedAccess)

18 Entering data in EpiData

Entering data takes place in the other part of the EpiData DMD-family; EpiData Entry Client (EDEC), which is a stand-alone program that can be installed without EDM. If having others to key in data, they should normally not have access to EDM.

The start window shows the simplicity of the program, its only purpose is to enter data into the E-file, and several people, eventually not taking part in the project management may use the same entry form for entering data with only a short introduction to the program. The administrator will also use EDEC in order to control that the E-file works as anticipated, and can afterward key in missing information.

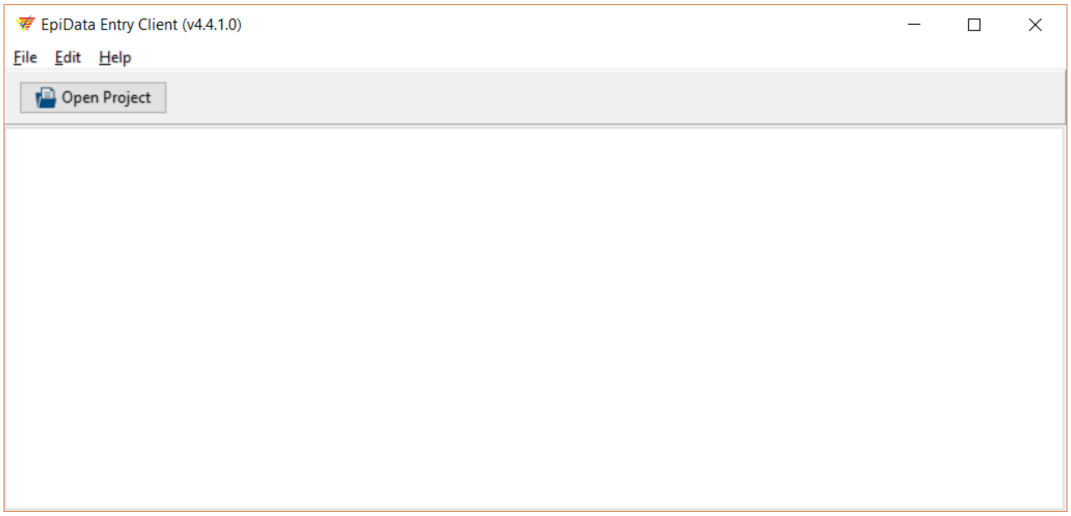
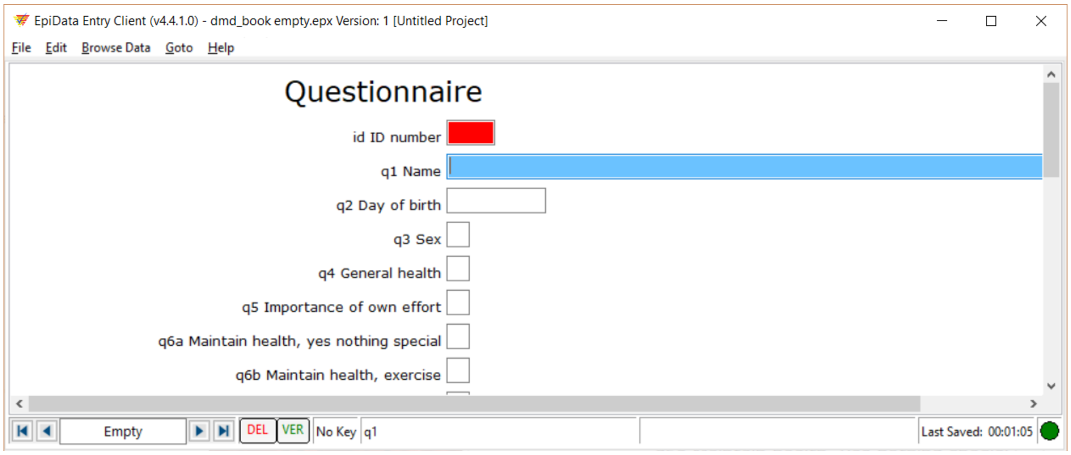


Figure 38 EpiData Entry Client opening look

For entering data, the keyer must open the relevant project (E-file), and later, this will be the most recent projects. They may also be opened by using <shift>+<ctrl>+num

The keyer will be guided by the information stated during the preparation of the data form.

EDEC uses color coding to guide the keyer (red for a ”must enter” field, blue for the actual field, yellow is a field, where the actual value will not meet the definitions being set up) and the information stated in the Properties Box will eventually present information or even the value label box to the keyer.


EpiData Entry Client (v4.4.1.0) - dmd_book empty.epx Version: 1 [Untitled Project]

Questionnaire

id ID number
q1 Name
q2 Day of birth
q3 Sex
q4 General health
q5 Importance of own effort
q6a Maintain health, yes nothing special
q6b Maintain health, exercise

Empty

No Key

q1

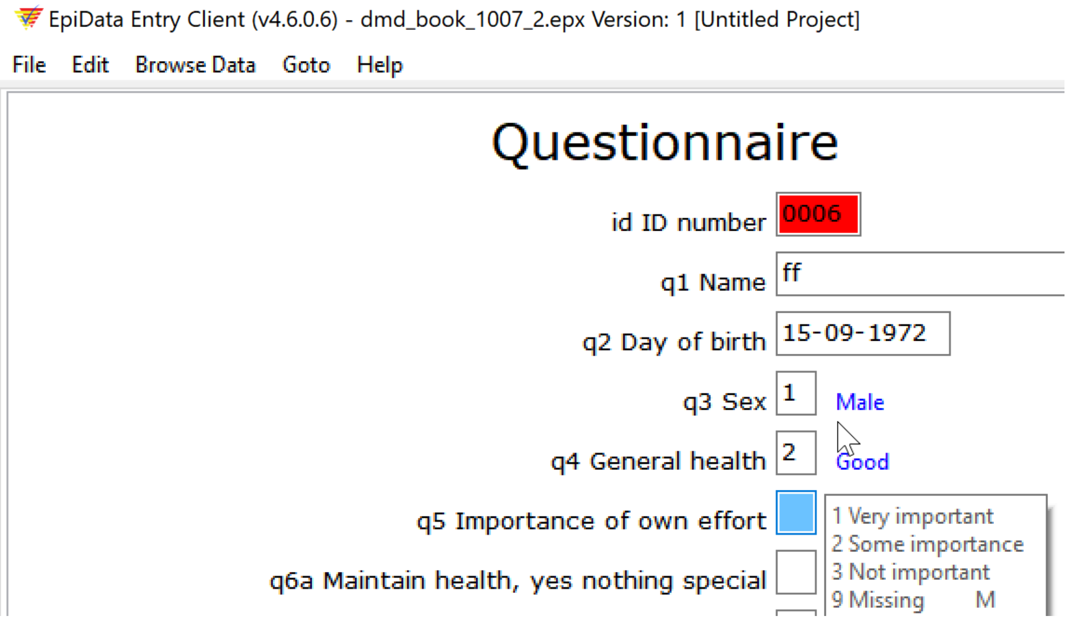
Last Saved: 00:01:05

Figure 39 EpiData Entry Client (EDEC) with an empty data set

The keyer can identify the record and variable in the bottom part of the view.

The actual record number is shown, and for the marked variable name is shown. A red "DEL" and green "VER" is used to show whether a record is marked for deletion or verification. If a key is defined (see chapter 17.1), this is shown, and the time since last saving is given.

If the information on value labels is not showing, it is available (press F9 or "+" to open).


EpiData Entry Client (v4.6.0.6) - dmd_book_1007_2.epx Version: 1 [Untitled Project]

Questionnaire

id ID number 0006

q1 Name ff

q2 Day of birth 15-09-1972

q3 Sex 1 Male

q4 General health 2 Good

q5 Importance of own effort

1 Very important
2 Some importance
3 Not important
9 Missing M

q6a Maintain health, yes nothing special

Figure 40 EDEC with the 6th record during entry. The marker is at the q5 variable (blue) and

the value label information is open. The id-variable is red, as it is a must-enter field

During entry blue value labels are visible just right of the variable if it was defined in the ”Extended pane”, helping the keyer to secure that the keyed information was according to the questionnaire.

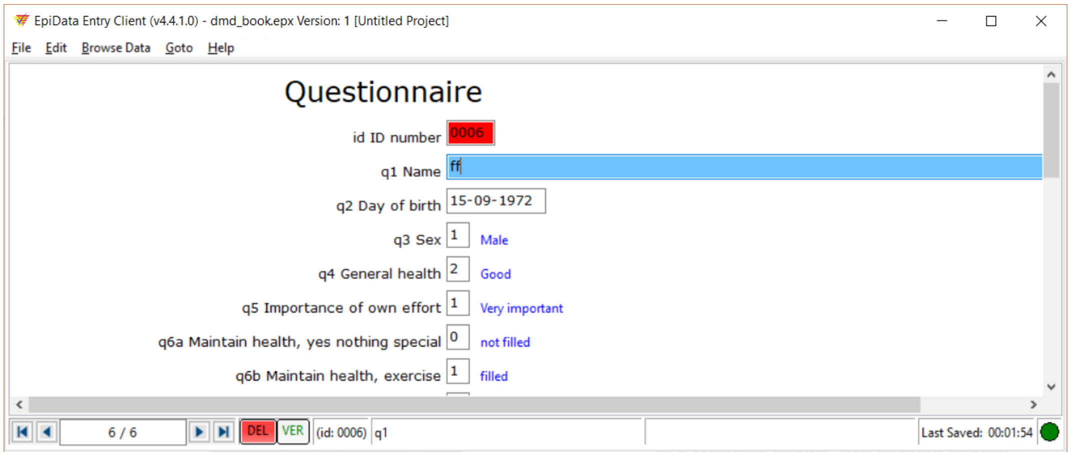
In the bottom left part of the window, records can be sifted through by using the arrows and it is shown how many records exist in the dataset (here 4).

If a new record is needed, it is inserted by pressing <CTRL><N> (for new), that will produce a new empty record at the end of those already keyed. Normally, however, you just pass the last variable in the last record and by this automatically creates a new record.

If having to delete the actual record (it might turn out that it has been entered twice), clicking on the red “DEL” will mark the record as deleted, shown by the DEL text in the information line turning into a red field, but the record is still present, and clicking the red DEL again will remove the marking, making it non-deleted.

While this might feel awkward, it will secure that data is not lost until the manager chose to do so.

Records marked as deleted will thus reside until the dataset is ”packed” (in EDM, textwtools > pack dataset) or exported. This way, the keyer is not able to delete records.


EpiData Entry Client (v4.4.1.0) - dmd_book.epx Version: 1 [Untitled Project]
File
Edit
Browse Data
Goto
Help
DEL
VER

Questionnaire

id ID number 0006

q1 Name ff

q2 Day of birth 15-09-1972

q3 Sex 1 Male

q4 General health 2 Good

q5 Importance of own effort 1 Very important

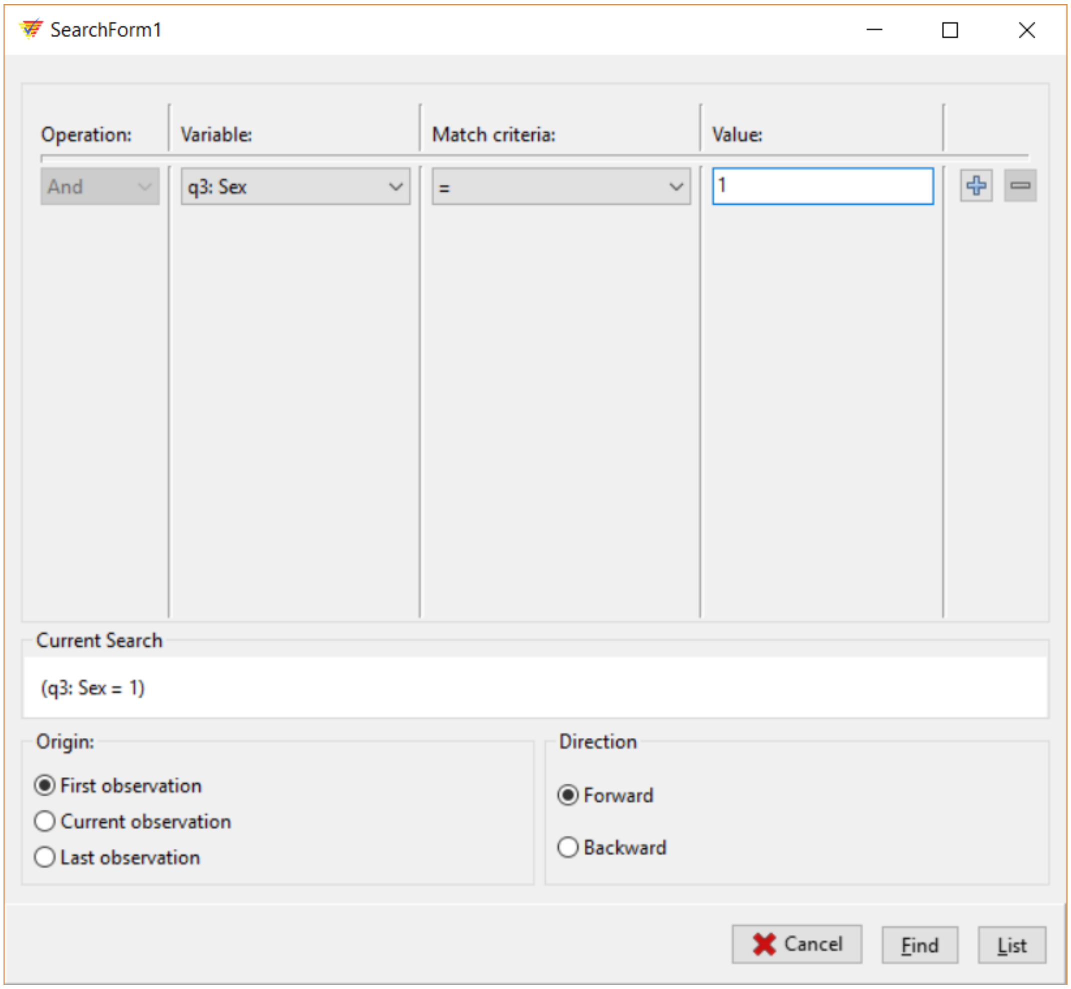
q6a Maintain health, yes nothing special 0 not filled

q6b Maintain health, exercise 1 filled

6 / 6 (id: 0006) q1 Last Saved: 00:01:54

Figure 41 The record has been marked as "deleted" by pressing the red "DEL" at the bottom

It is possible to search entered data using the “Browse Data” curtain and setting in values for variables, eventually combining them. In the example below I’m searching for records that have the value 1 (males) in variable q3. Using the "+" sign more variables can be evaluated, eventually also for being larger, smaller or unequal to the value.



Operation:	Variable:	Match criteria:	Value:
And	q3: Sex	=	1

Current Search

(q3: Sex = 1)

Origin:

- ● First observation
- ○ Current observation
- ○ Last observation

Direction

- ● Forward
- ○ Backward

Figure 42 Search form for finding records

It is also possible to browse and either choosing a particular record or typing information in the variables, that should be found.

Reaching the last field (variable) and leaving it, will make EDEC prompt for saving data if changes have been made, and if answered yes, this will both save the record and open a new empty record.

18.1 Using note generation during entry (F5)

When something that was not foreseen during preparation happens, the keyer will have to make decisions and should document these for the administrator to evaluate and eventually change.

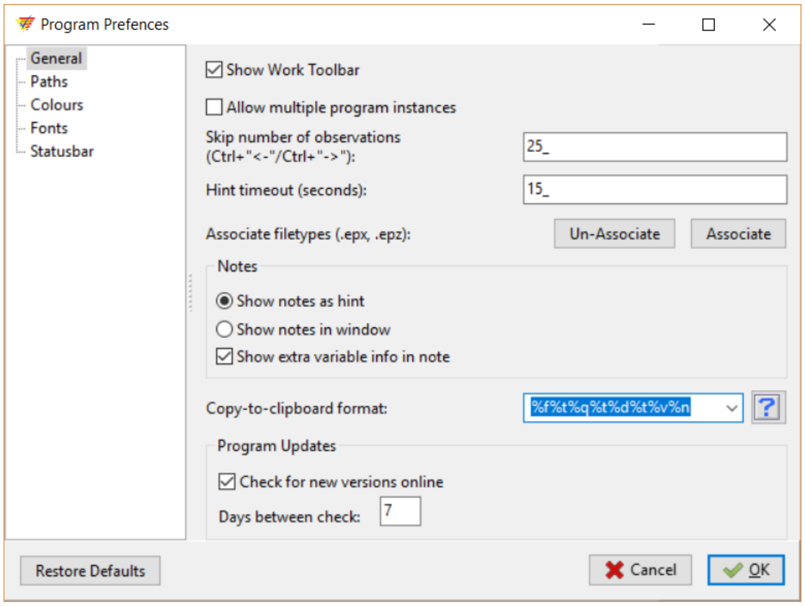
Maybe the informant has put two crosses in a field meant to include only one, and the keyer will need to decide on what to do if this was not anticipated and communicated prior to the keying.

The administrator will need information on: date, record, variable, keying person, problem and decision

The keyer should be instructed to make a note in a file for the collection of this sort of information (for instance named xxx.not.docx or xxx.not.txt). The administrator must instruct the keyers to make such a file and to keep documentation of the problems encountered and how it was handled.

The keyer can use <shift>+<ctrl>+<c> or F5 in EDEC for placing a copy of either the entire record or the actual variable in the clip board and then pasting it for a text-file (works best in a text-editor like notepad)

In Edit>Preferences, Copy-to-clipboard a format line can be inserted to steer what information should be copied if using F5.



☑ Show Work Toolbar

□ Allow multiple program instances

Skip number of observations (Ctrl+"<-"/Ctrl+"->"): 25_

Hint timeout (seconds): 15_

Associate filetypes (.epx, .epz):

Notes

● Show notes as hint

○ Show notes in window

☑ Show extra variable info in note

Copy-to-clipboard format: %f%t%q%t%d%t%v%n

Program Updates

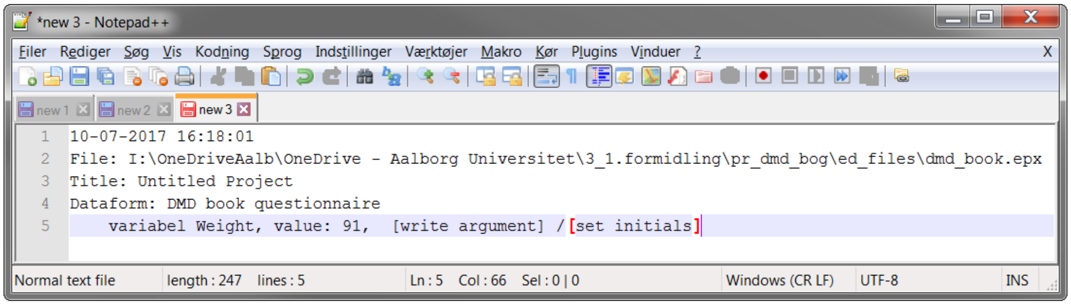
☑ Check for new versions online

Days between check: 7

Figure 43 Preferences for EDM, "Copy-to-clipboard format:" defines what is written to the clipboard when hitting F5

If, for instance, writing/copying the following text into the window (copy-to-clipboard format) %gd %gn %gf %gp variabel %q, value: %d, %v [write argument] / [set initials] will make a copy of actual date, time, file, data form, variable and value (all defined by the "%" part) + put some short text information in the square brackets into the clipboard, and when pasted to the ...not.txt file, this will help to secure that the keyer remembers to describe the problem and solution and to put initials in the file.

The resulting file would read as shown in the figure below.



```
10-07-2017 16:18:01
File: I:\OneDriveAalb\OneDrive - Aalborg Universitet\3_1.formidling\pr_dmd_bog\ed_files\dmd_book.epx
Title: Untitled Project
Dataform: DMD book questionnaire
    variabel Weight, value: 91,  [write argument] /[set initials]
```

Figure 44 Using the above clipboard format and hitting the F5 button, then paste in Notepad will give this information

The keyer could then fill out the last line, writing for instance in the square brackets: variable Weight, value: 91, in the questionnaire it is stated "90-92" - I've chosen the middle value / HEB. By that the Manager will be able to evaluate whether this is the right decision and eventually change it accordingly.

Another possibility is to include a variable in the dataset for this. It would be relevant to set up a memo field as the last one and instruct the keyer to write any deviations during entry of this field. It will then be included in the dataset and it could be excluded when exporting the dataset for analyses. For the administrator, going through the memo-variable and deciding what to do is fast. Again, in order to document the work being done, changes should be written (eventually in the same memo-field) and saved under a new name.

18.2 The LOCK-file problem

ED mark that the E-file is in use by placing a special LOCK-file when opening the E-file it in either EDM or EDEC. Normally, the file will be deleted automatically when EDM/EDEC is closed and the E-file is again free, but sometimes the deletion of the LOCK-file is compromised, and the user get information that the E-file is in use by another user. In this case – after having ensured that the file is NOT open in another session of either EDM or EDEC – the LOCK-file should be found and manually deleted. It will be placed in the same folder as the e-file itself.

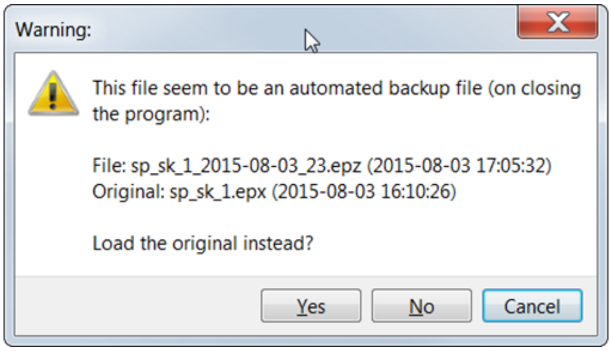
18.3 Automatic back-up

ED has an automatic back-up procedure as a standard (but that can be changed, see chapter X), which will produce a set of regular backups, the default is once every 10 minute.

The default is that back-ups are placed in a folder called "backup", that is automatically created in the folder containing the EPX file. These back-up datasets are zipped to preserve space and are called .epz-filer (z for zipped) and named: xxx_yyyy-mm-dd_lbnr.epz, where xxx is the original name, yyyy–mm-dd is the year, month and day of the back-up, and lbnr is a number increasing during the day.

This makes it easy to find the newest back-up file if anything goes wrong, and if needed an older working version of the database can be found

The .epz file can be opened directly in EDM/EDEC but will provide a warning, that an original file exists. As always, it is highly advisable to read the warnings in ED!


Warning:

This file seem to be an automated backup file (on closing the program):

File: sp_sk_1_2015-08-03_23.epz (2015-08-03 17:05:32)
Original: sp_sk_1.epx (2015-08-03 16:10:26)

Load the original instead?

Figure 45 Setting up the key variable, several variables might be included

If choosing to work on a back-up file it should be renamed using save-as and it is suggested to keep information on the back-up number used.

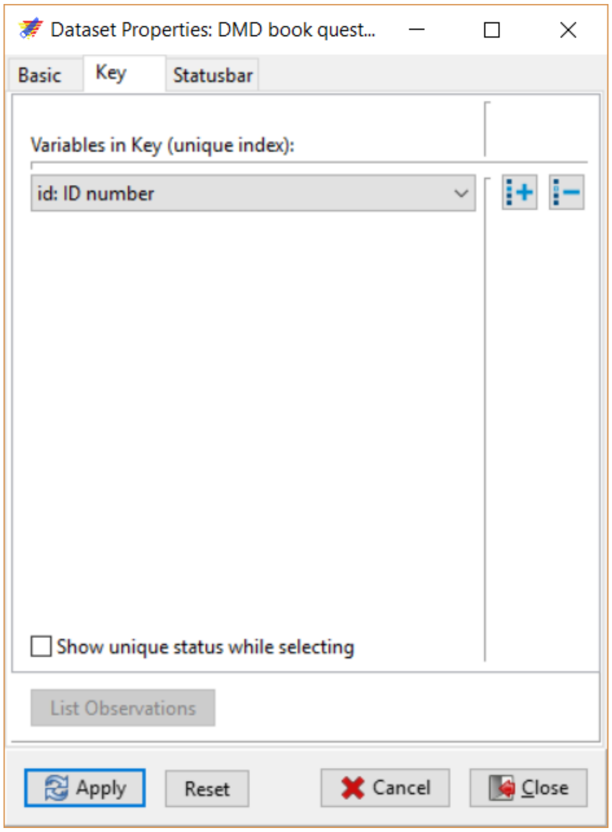
The new file should be saved using the standard .epx type.

19 Unique identification of record

I have already suggested that each record should be identified by an identification variable, but this will not protect against the same number being used in several rows.

Instead, we should construct and define a unique identification of each record, that is only used once in the dataset.

This key variable consists of one or a combination of variables, that will only be found once, it could be an ID-number alone if the dataset is one-dimensional, or it could be a combination of for instance ID and date of visit if the same person could have more than one entry in the dataset.



Variables in Key (unique index):

id: ID number

□ Show unique status while selecting

Figure 46 Defining a unique key

In EDM and EDEC the key variables become red (as they are automatically set as “must enter” variables).

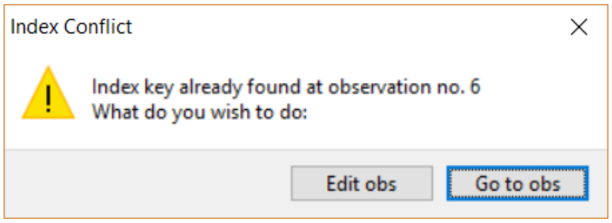
When defining the key variable(s) this is part of the database, not the variables itself, so it is set for the entire “canvas”. Right-click project or use the curtain Dataset and choose Dataset properties and choose the Key pane.

When the unique key is chosen after keying has started, EDM will evaluate whether the existing data are in fact unique and otherwise post a warning.

If the key is not unique you could either chose one more variable until unique status is found or list observations that are not unique in order to see whether errors did occur until now

In EDEC the presence of a key will be shown in the status bar with the current key (for instance: (id: 0006) for the record with ID numbered 6).

Then, if trying to use a key that is already in use during entry, EDEC will ask whether:



Index Conflict

Index key already found at observation no. 6
What do you wish to do:

Figure 47 Warning if trying to use a non-unique value during entry when a key has been defined

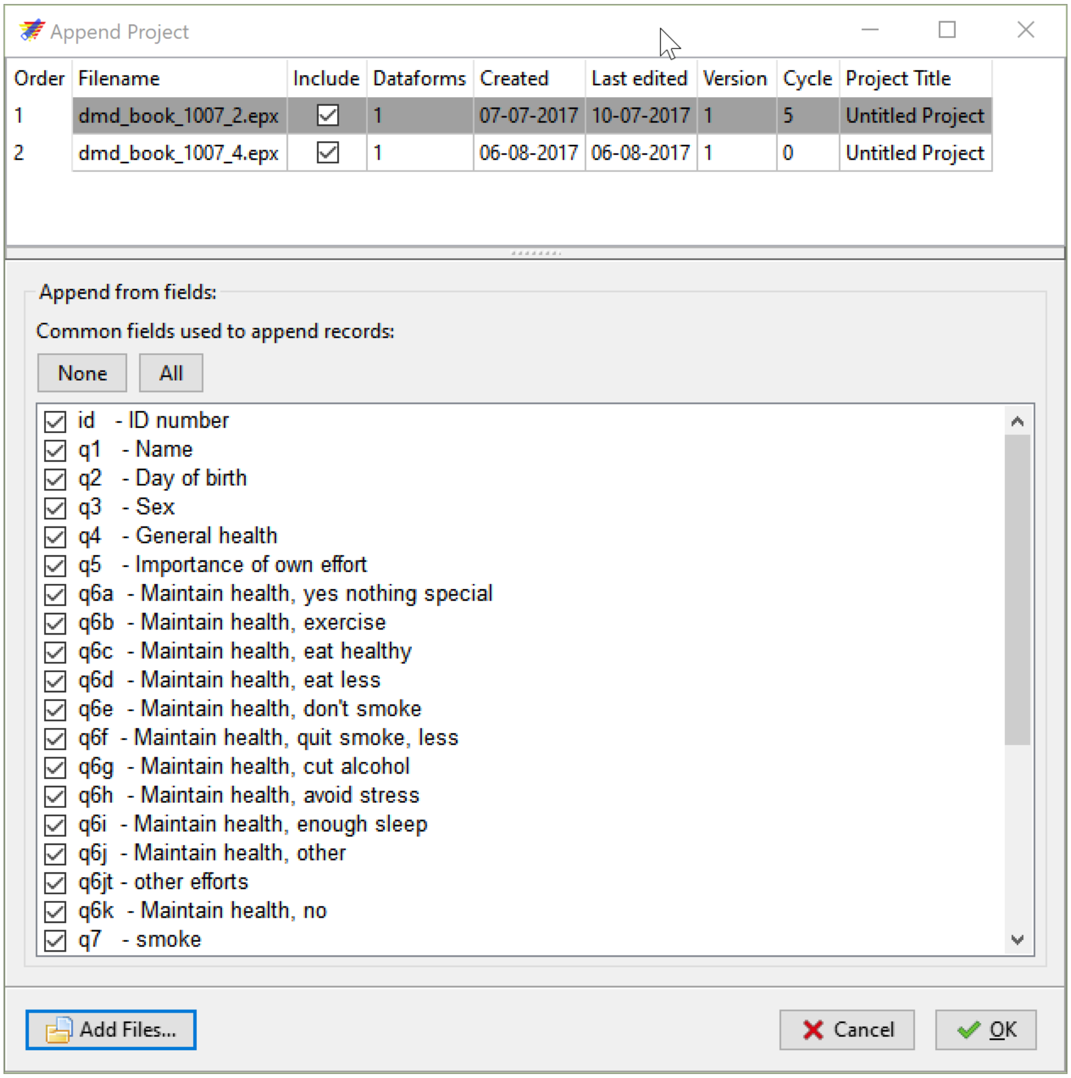
- You want to change the wrongly keyed information (Edit obs) or
- Bring the existing record in focus Go to obs for the user to choose whether it should be changed.

This last possibility is an easy way of finding a record that should be changed or have amended information.

20 Combining EDEC dataset in EDM

Having different keyers to prepare different parts of the questionnaire will end up having several E-files, the individual E-files should eventually be collected and combined, as each file is assumed to hold *different* records.

This is made by the administrator in EDM.



Order	Filename	Include	Dataforms	Created	Last edited	Version	Cycle	Project Title
1	dmd_book_1007_2.epx	☑	1	07-07-2017	10-07-2017	1	5	Untitled Project
2	dmd_book_1007_4.epx	☑	1	06-08-2017	06-08-2017	1	0	Untitled Project

Append from fields:

Common fields used to append records:

- ☑ id - ID number
- ☑ q1 - Name
- ☑ q2 - Day of birth
- ☑ q3 - Sex
- ☑ q4 - General health
- ☑ q5 - Importance of own effort
- ☑ q6a - Maintain health, yes nothing special
- ☑ q6b - Maintain health, exercise
- ☑ q6c - Maintain health, eat healthy
- ☑ q6d - Maintain health, eat less
- ☑ q6e - Maintain health, don't smoke
- ☑ q6f - Maintain health, quit smoke, less
- ☑ q6g - Maintain health, cut alcohol
- ☑ q6h - Maintain health, avoid stress
- ☑ q6i - Maintain health, enough sleep
- ☑ q6j - Maintain health, other
- ☑ q6jt - other efforts
- ☑ q6k - Maintain health, no
- ☑ q7 - smoke

Figure 48 Combining several E-files into one

With the first of the files open, using Tools > Append will produce a box with the possibility to set up a list of all the different files to be appended (use the Add Files... button at the bottom). Many E-files can be lined up and their order might be changed. Afterwards, EDM produce a list of the variables that are common (hopefully all) and gives the possibility to exclude variables, that should not be included in the final E-file.

The resulting file is to be saved in a new file, and the merging should also be documented by saving the report file produced by EDM.

21 Importing data into EDM

EDM can import data from external sources, for instance, if you receive data in another database format.

By importing data into EDM it can be evaluated using the same tools as described above, in many instances, but this can be cumbersome work unless an E-file is already existing.

The tools that can be used for a newly imported data file include Document > Count records by Id, which controls whether duplicated data are present (e.g. that some records have been entered twice) – control can be made on all variables or on those with certain information.

Data Content Validation can control whether checks defined will be violated by the already entered data. This will be helpful if a E-file is already defined, for instance if someone have been keying in data in a structure resembling the E-file.

22 Controlling data

Even when setting up a thorough entry file with checks and jumps, errors are obliged to be made during entry and as has previously been argued, these errors should be found and corrected before using the data.

Two possible solutions are at hand, either proof reading data or use the double entry possibility given by EDM.

22.1 Proof reading data

The data can be proof read, for instance by two persons working together. One should then read the keyed data, the other controlling the source (questionnaire). In EDM the data can be shown by using Dataset > Browse Data or it can be written in tables using EDEC, Browse Data > All Data , making it easier to read.

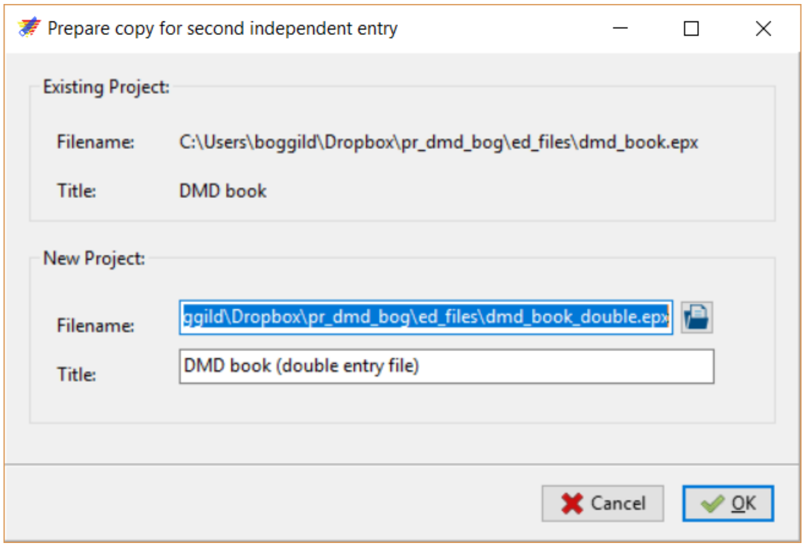
The data are shown as their corresponding value labels, but this can be changed by toggling the first button.

Errors should then be marked on the source and afterwards changed to document the changes made.

Proof reading data is hard work, takes time and it won't work with large datasets, as people lose their concentration.

22.2 Double entry

Instead, it is suggested to re-enter data and compare the two independent files, as it is highly unlikely that the same error should occur the same place in the datasets. If the two files are the same variable by variable, then it is supposed to be accurate.



Existing Project:

Filename: C:\Users\boggild\Dropbox\pr_dmd_bog\ed_files\dmd_book.epx

Title: DMD book

New Project:

Filename: ggild\Dropbox\pr_dmd_bog\ed_files\dmd_book_double.epx

Title: DMD book (double entry file)

Figure 49 Preparing a file for double entry.

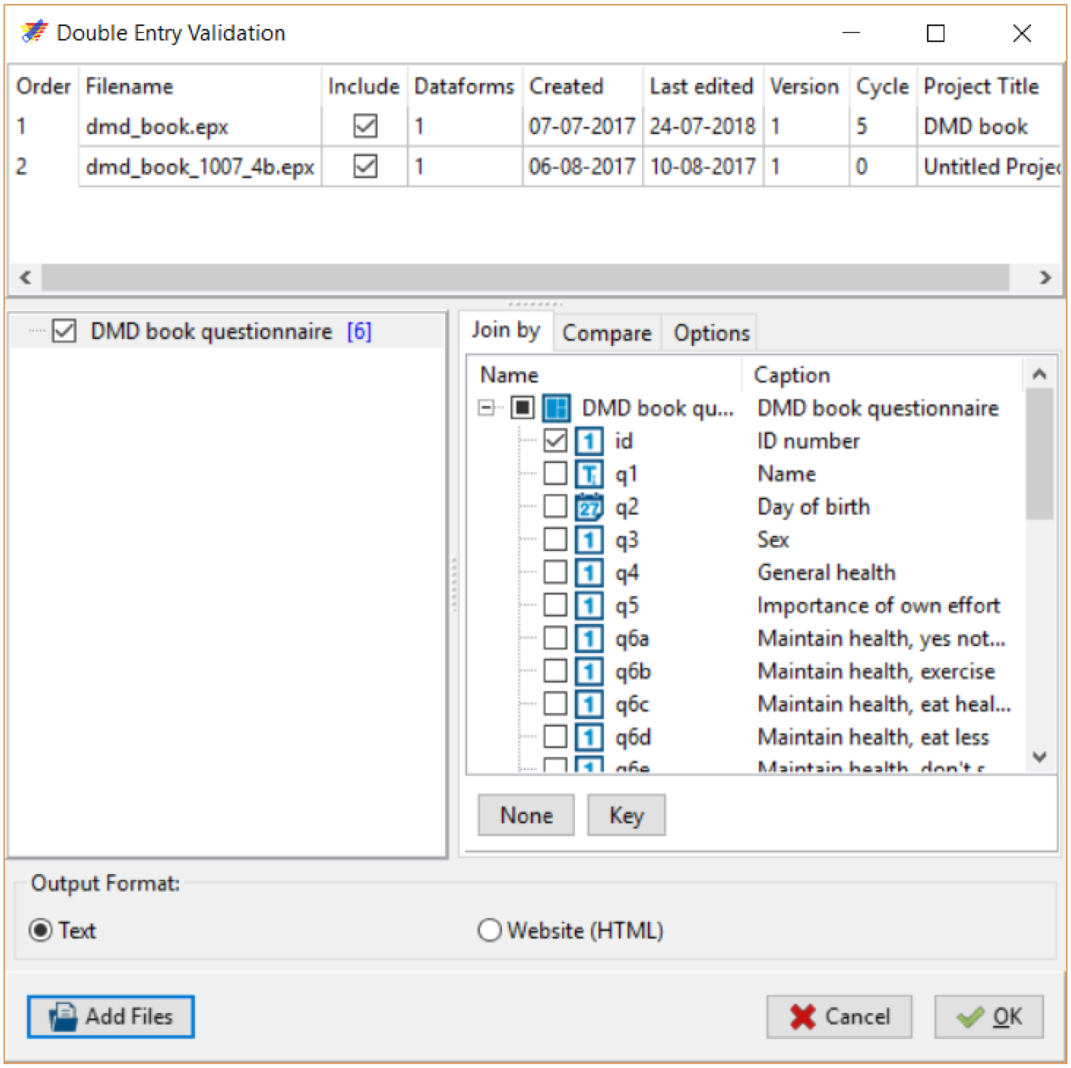
In EDM a copy of the E-file structure is prepared without data by using Tools > Prepare Double Entry. EDM suggests a filename with information that the file is a result of double entered data, but this is not mandatory. The file structure is identical to the original structure but will not have any records in it.

It should be keyed in the same way as the original file, although the records don't need to be entered in the same order, in that case EDM just needs to know how records are identified, for instance by defining a KEY

The new project is NOT opened automatically after preparation of the copy.

When the second file has been keyed in and is ready, the two files are compared using Document > Compare Duplicate Files and the actual file in EDM will be chosen automatically while the file to be compared with should be marked using the "Add files" button below

EDM suggest a possible key for identification of the corresponding records in the two files at the Join by pane, if having defined key unique (see chapter 17.1) then this is used.



Order	Filename	Include	Dataforms	Created	Last edited	Version	Cycle	Project Title
1	dmd_book.epx	☑	1	07-07-2017	24-07-2018	1	5	DMD book
2	dmd_book_1007_4b.epx	☑	1	06-08-2017	10-08-2017	1	0	Untitled Projec

☑ DMD book questionnaire [6]

Name	Caption
■ DMD book qu...	DMD book questionnaire
☑ id	ID number
□ q1	Name
□ q2	Day of birth
□ q3	Sex
□ q4	General health
□ q5	Importance of own effort
□ q6a	Maintain health, yes not...
□ q6b	Maintain health, exercise
□ q6c	Maintain health, eat heal...
□ q6d	Maintain health, eat less
□ q6e	Maintain health, don't s

Output Format:

● Text ○ Website (HTML)

Figure 50 Comparing two possible identical E-files after double entry

Under the compare pane it is possible to exclude text variables, that would normally not be compared.

In the third Options-pane it is possible to further exclude comparisons of deleted records and not to compare case status in text variables.

It is also possible to allow missing records in the second file. This is not advisable if we compare the complete data, but if the database is very large, it might be unrealistic to double-enter all data. Instead, an estimation of the proportion of errors could be given by re-entering some of the records a second time and use this to calculate the error percentage for this part of the sample. It would at the same time give an indication of the need to find resources to re-enter the rest of the data. If using double entry for this purpose, you should allow missing records in the second (re-entered smaller) file. Furthermore, you should not attempt to correct the errors as this will bias the data.

It is possible to add a new variable, that mark whether the two records are the same, making it easy to find the non-matching ones. This is advisable.

A report of the comparison results, showing the files being compared, the key used and an overview of the number of differences found is produced. The differences are shown with the key and the differing information in the two files. Both records and variables that are not the same are identified (see figure below).

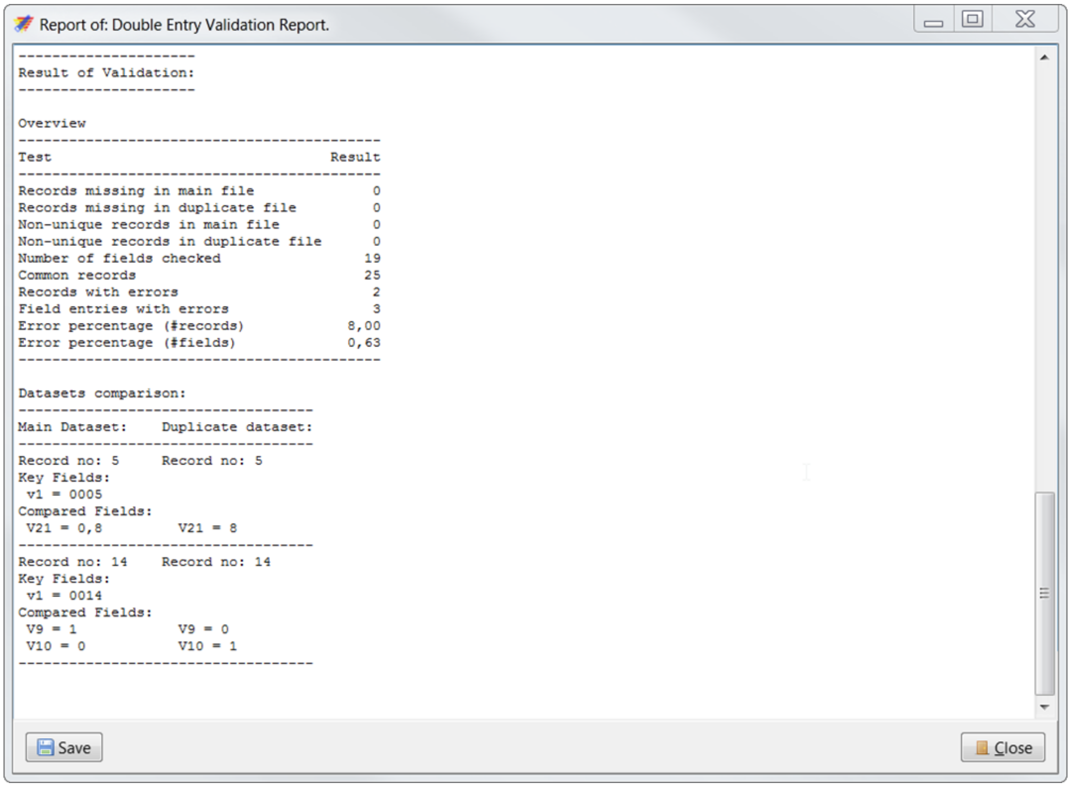
This report is part of the documentation and should be saved.

Next, if using the double entry procedure for all data, the differences identified should be solved, possibly by going back to the original data (questionnaire) and finding out which information is correct.

In order to document the work, we should not change either of the two existing files, but rather copy one of them and make the changes in this file.

This copied file should be given a new name, that shows that the content is the final version (*.final.eps).

The errors shown in the report should then be changed in this final E-file and afterward documented, for instance using document > codebook.



```
---------------------
Result of Validation:
---------------------

Overview
-------------------------------------------
Test                                 Result
-------------------------------------------
Records missing in main file              0
Records missing in duplicate file         0
Non-unique records in main file           0
Non-unique records in duplicate file      0
Number of fields checked                 19
Common records                           25
Records with errors                       2
Field entries with errors                 3
Error percentage (#records)            8,00
Error percentage (#fields)             0,63
-------------------------------------------

Datasets comparison:
-----------------------------------
Main Dataset:    Duplicate dataset:
-----------------------------------
Record no: 5     Record no: 5
Key Fields:
 v1 = 0005
Compared Fields:
 V21 = 0,8         V21 = 8
-----------------------------------
Record no: 14    Record no: 14
Key Fields:
 v1 = 0014
Compared Fields:
 V9 = 1            V9 = 0
 V10 = 0           V10 = 1
-----------------------------------
```

Figure 51 Result of double entry verification. Two differences identified, in record 5 the VAS scale is either 0.8 or 8 and in record 14 it seems as V9 and V10 are switched around

23 Documenting data structure

EDM can be used to write documentation of the entry form (database).

In the curtain Document > Report Structure access is given to text output of a range of possibilities.

"**Project Overview**" will provide a description of the project "label", corresponding to the information provided to the base project (see chapter 14).

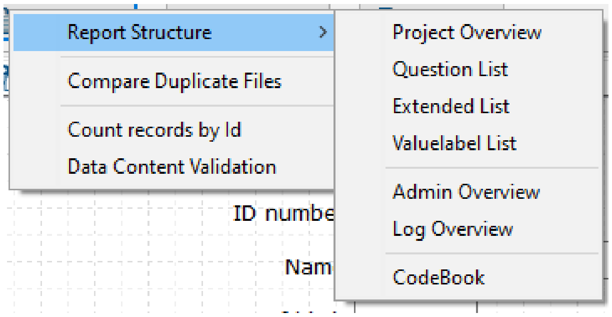


Figure 52 Possibilities for producing documentation in EDM.

The "**Question List**" describes all defined variables with their variable labels

"**Extended List**" describes which checks were set up in the "Properties Variable" panes.

The "**Valuelabel List**" provides a list of the value labels defined in the "Properties Variable"

Finally, the "**CodeBook**" option describes variables with most of the above and can be used as the base of a total documentation from these

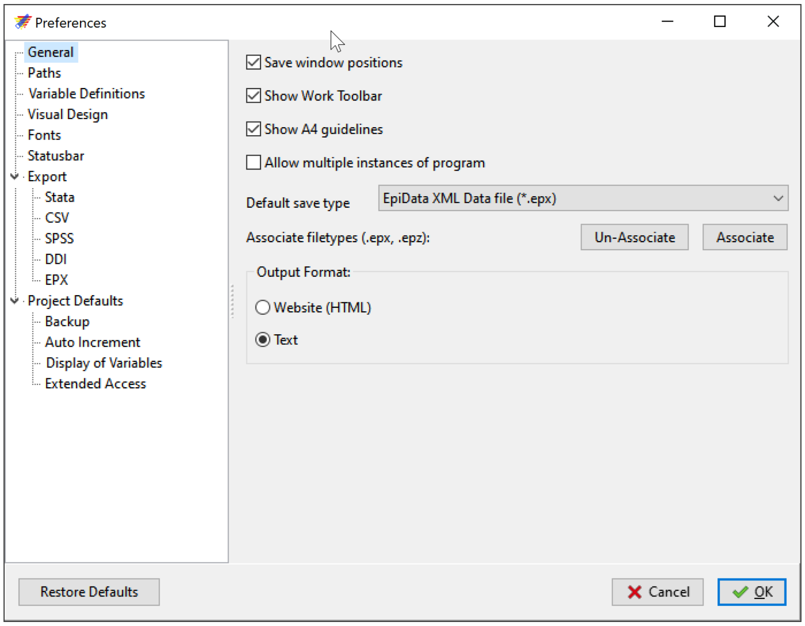
A fifth entry, **Admin Overview** is only working when the "User Access possibilities have been set up, allowing different persons to have different tasks and using password protection for the database. The documentation would include list of the defined users and their roles, including information on their last access to the data.

24 Setting up EpiData

It is possible to change the behaviour of EDM and EDEC. In both programs choosing Edit > Preferences will display a list of settings that can be changed. Possibilities differ between the programs.

24.1 EDM preferences

In EDM the box looks like this:



☑ Save window positions

☑ Show Work Toolbar

☑ Show A4 guidelines

□ Allow multiple instances of program

Default save type: EpiData XML Data file (*.epx)

Associate filetypes (.epx, .epz):

Output Format:

○ Website (HTML)

● Text

Figure 53 Changing preferences in EDM

Of special interest is the “Save window position”, especially when using multiple screens, as ED will “disappear” if unplugging the screen at which ED was used. Using multiple screens, I would not save the windows position. It is possible to put ED back on its default position using Shift+Ctrl+0.

It is suggested not to allow multiple instances of the program, as this might end up in locked files (see chapter 18.2).

At the next group (Paths) the default folders for the E-files can be defined, and at the “Variable Definitions”, the length and default dates can be defined. The “1_ _”, “5_ _” and “20_” can be changed for longer integer, float and string variables. Also default date type is defined here. The "Variable Naming" can be changed (e.g. Q will name variables from Q1 and upwards).

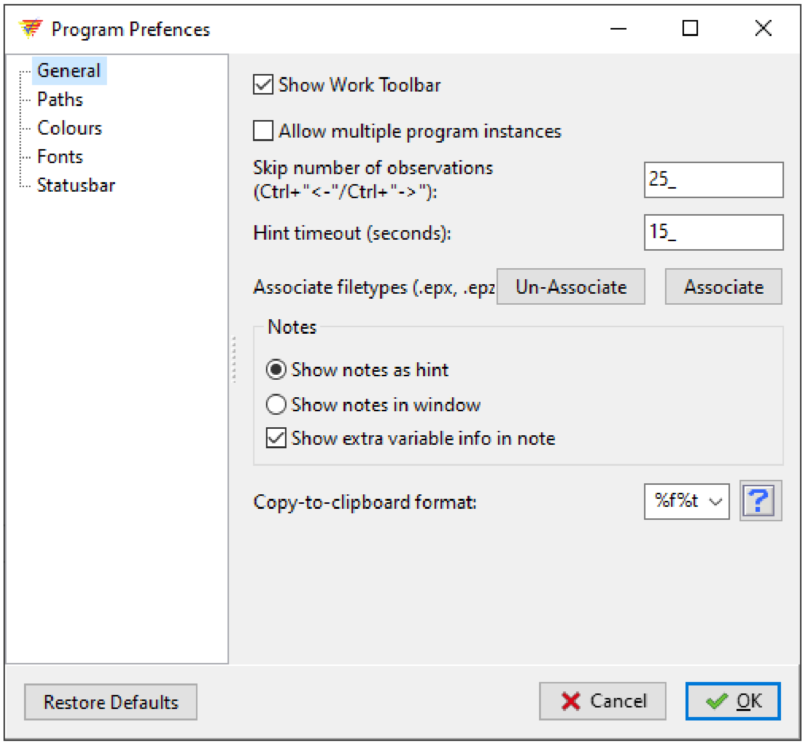
"Visual Design" and "Fonts" define how the variables and characters look on the canvas.

In the "Statusbar" information to be described in the lower part of the window is shown. Each element can be moved by using the arrows at the right of the element.

In the Export possibilities, it is possible to define how data should be exported to the 5 different file types, although this can also be changed in the exporting tool itself. It is possible to export deleted observations, otherwise, they will be left with the E-file, and value labels can be exported for STATA and SPSS (but not CSV and DDI).

Finally, in the "Project Defaults" backup can be defined, the 10_ in "Timed Recovery Backup Interval:" means 10 minutes, and can be changed. It is also possible to have the backup mailed to a mail recipient. It is possible to change the start of auto incrementing data, Finally, as is used in this text, it is possible to change the appearance of variable names, so they appear besides labels, it is also possible to change the 8-dot variable border, and the appearance of the value label.

I like to have the variable name shown on the canvas (as is used in the examples shown here), this can be achieved in the "Display of Variables" section by choosing "show variable names (besides variable label)".



☑ Show Work Toolbar

□ Allow multiple program instances

Skip number of observations (Ctrl+"<-"/Ctrl+"->"): 25_

Hint timeout (seconds): 15_

Associate filetypes (.epx, .epz

Notes

● Show notes as hint
○ Show notes in window
☑ Show extra variable info in note

Copy-to-clipboard format: %f%t

Figure 54 Figuren skal rettes til

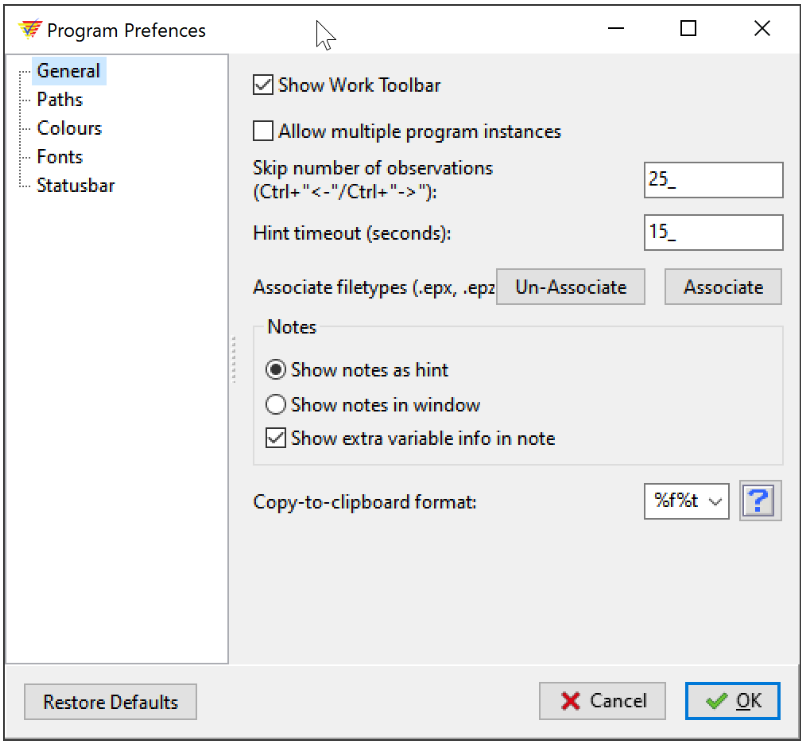
If not satisfied with changes, the “Restore Defaults” will have ED working as it did when installed.

24.2 EDEC preferences

In EDEC possibilities are fewer, some are essentially the same (“Paths”, “Fonts” and “Statusbar”) as in EDM, in the General part it is further possible to define the amount of time that hints are shown, and notes might be shown in regular windows.

The “Copy-to-clipboard” format can be used to define which information from the actual records that should be included in the information going to the clipboard, and from that included in a report (see chapter 18.1). More information is given in a wiki document, that can be called by clicking the question mark.

In the “Colours” part, the use of different colours for EDEC can be defined.



☑ Show Work Toolbar

□ Allow multiple program instances

Skip number of observations (Ctrl+"<-"/Ctrl+"->"): 25_

Hint timeout (seconds): 15_

Associate filetypes (.epx, .epz

Notes

● Show notes as hint
○ Show notes in window
☑ Show extra variable info in note

Copy-to-clipboard format: %f%t

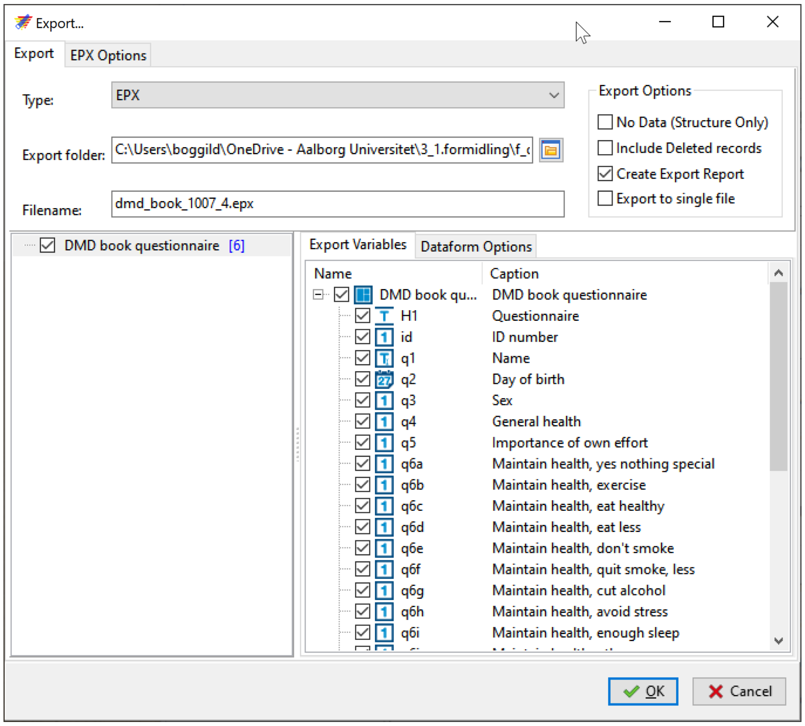
Figure 55 Preferences in EDEC

25 Exporting data to statistical software

25.1 Exporting to ED

While it might seem odd that ED can export dataset in its own type (.epx), it is highly relevant, as you can use it to copy the structure (for instance after testing (see chapter 16) without data, to pack the records marked for deletion (the red marking in ED) making the marked records disappearing, to deselect some of the variables from the dataset (for instance for making un-identifiable dataset (see chapter 6.7)) or to export only some of the records (Dataform Options).

It is also possible to exclude exporting of value labels (the upper pane EPX options).



Type: EPX

Export folder: C:\Users\boggild\OneDrive - Aalborg Universitet\3_1.formidling\f_(

Filename: dmd_book_1007_4.epx

Export Options
- □ No Data (Structure Only)
- □ Include Deleted records
- ☑ Create Export Report
- □ Export to single file

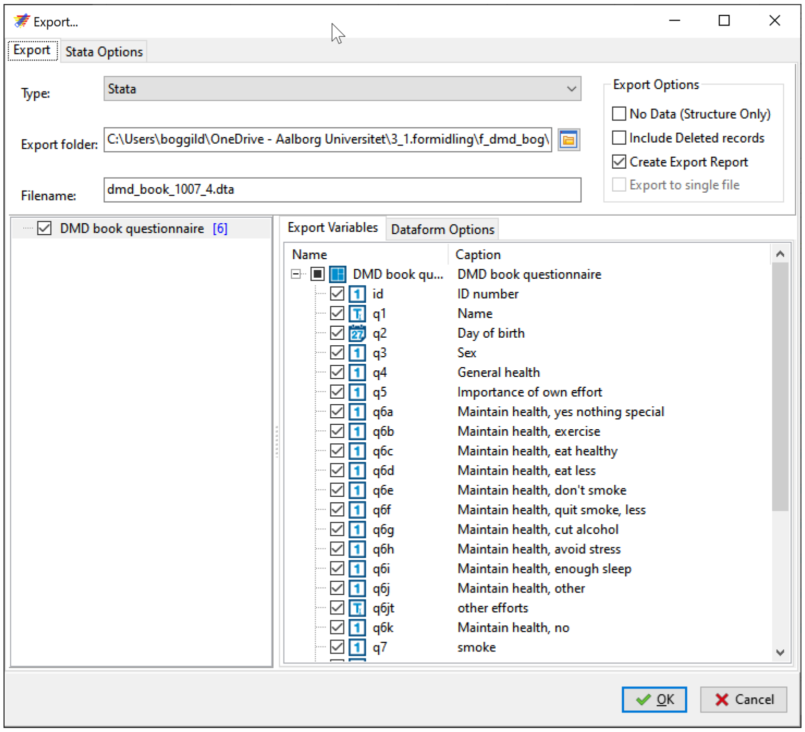
☑ DMD book questionnaire [6]

Name	Caption
☑ DMD book qu...	DMD book questionnaire
☑ H1	Questionnaire
☑ id	ID number
☑ q1	Name
☑ q2	Day of birth
☑ q3	Sex
☑ q4	General health
☑ q5	Importance of own effort
☑ q6a	Maintain health, yes nothing special
☑ q6b	Maintain health, exercise
☑ q6c	Maintain health, eat healthy
☑ q6d	Maintain health, eat less
☑ q6e	Maintain health, don't smoke
☑ q6f	Maintain health, quit smoke, less
☑ q6g	Maintain health, cut alcohol
☑ q6h	Maintain health, avoid stress
☑ q6i	Maintain health, enough sleep

Figure 56 Exporting to ED format

25.2 Exporting data to STATA and SPSS

EDM includes the possibility to export data directly to STATA and SPSS. ED knows different versions of the two programs and this should be marked in the “Options” pane, either in the setup to use the same version in all new instances (see chapter 24.1) or in the actual session.



Type: Stata

Export folder: C:\Users\boggild\OneDrive - Aalborg Universitet\3_1.formidling\f_dmd_bog\

Filename: dmd_book_1007_4.dta

Export Options

☐ No Data (Structure Only)

☐ Include Deleted records

☑ Create Export Report

☐ Export to single file

☑ DMD book questionnaire [6]

Name	Caption
DMD book qu...	DMD book questionnaire
☑ id	ID number
☑ q1	Name
☑ q2	Day of birth
☑ q3	Sex
☑ q4	General health
☑ q5	Importance of own effort
☑ q6a	Maintain health, yes nothing special
☑ q6b	Maintain health, exercise
☑ q6c	Maintain health, eat healthy
☑ q6d	Maintain health, eat less
☑ q6e	Maintain health, don't smoke
☑ q6f	Maintain health, quit smoke, less
☑ q6g	Maintain health, cut alcohol
☑ q6h	Maintain health, avoid stress
☑ q6i	Maintain health, enough sleep
☑ q6j	Maintain health, other
☑ q6jt	other efforts
☑ q6k	Maintain health, no
☑ q7	smoke

Figure 57 Figuren skal rettes til

In the pane labelled Stata options different versions of STATA can be chosen, this is relevant if reading STATA-files with e.g. R or SAS (see below). It is also possible to rename variables for uppercase, lowercase or as it is present in ED (as STATA is reading q1 and Q1 as two different variables). Consider saving as lowercase, which is easier to use in STATA

In the SPSS options, only the possibility to exclude labels can be selected.

25.3 Exporting data to SAS and R

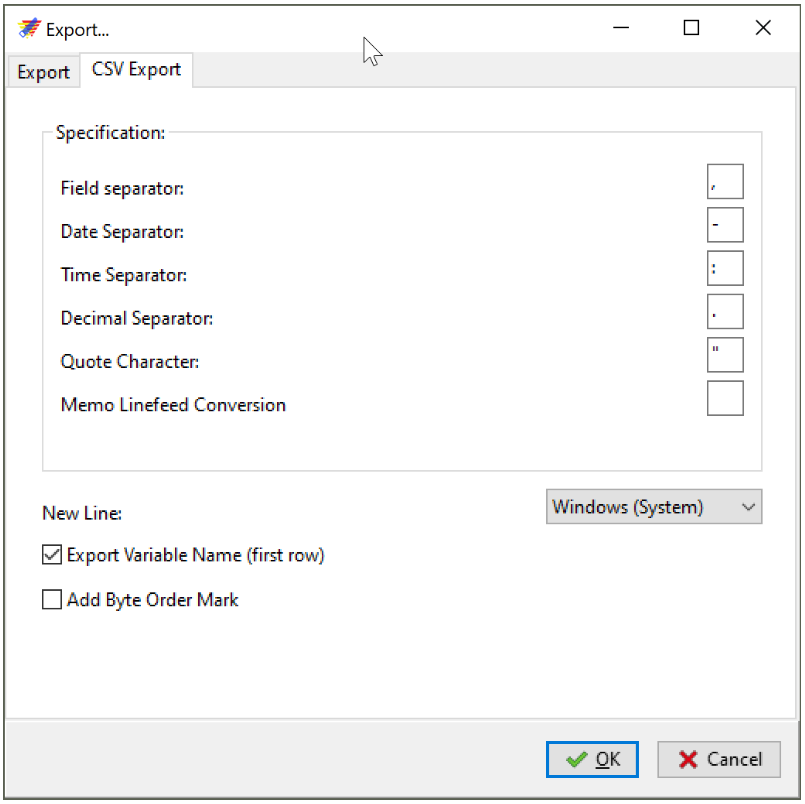
Although no direct export possibility exist, SAS can read STATA and SPSS files directly (using proc import)

SAS 9.4 can read STATA .dta files from version 12, so in the options pane, this version should be chosen. For SPSS this is not an issue.

R can import STATA and SPSS files directly as well, eventually using packages (the package ‘foreign’ is working with both file types (and many others), but not with STATA after version 12. Either make sure to export in version 12 or install the R-package readstata13 as a possibility; although named 13, it will read all versions (up to version 17 at the moment), which will include all labels).

25.4 Exporting to other statistical programs, using CSV

For nearly all statistical programs, data can normally be imported from a comma-separated file (CSV). In the CSV-options in EDM, it is possible to change separators and to secure export of variable names. The CSV options do not allow export of variable- and value labels. In this instance, it is advisable to document the dataset and to include labelling in the other statistical software (see chapter 39).



Field separator: ,

Date Separator: -

Time Separator: :

Decimal Separator: .

Quote Character: "

Memo Linefeed Conversion

New Line: Windows (System)

☑ Export Variable Name (first row)

☐ Add Byte Order Mark

Figure 58 Setting up separators for exporting in CSV format

25.5 Exporting for archiving

The .DDI format is an international standard (Data Documentation Initiative) meant to ease data archiving, and exporting in this format from EDM is used for archiving data in the Danish State Archives.

It is a free standard, and might also be used to move data including labelling from one platform to another. The exporting possibilities have a variety of settings.

Part III

Data management and documentation in REDCap

26 REDCap in short

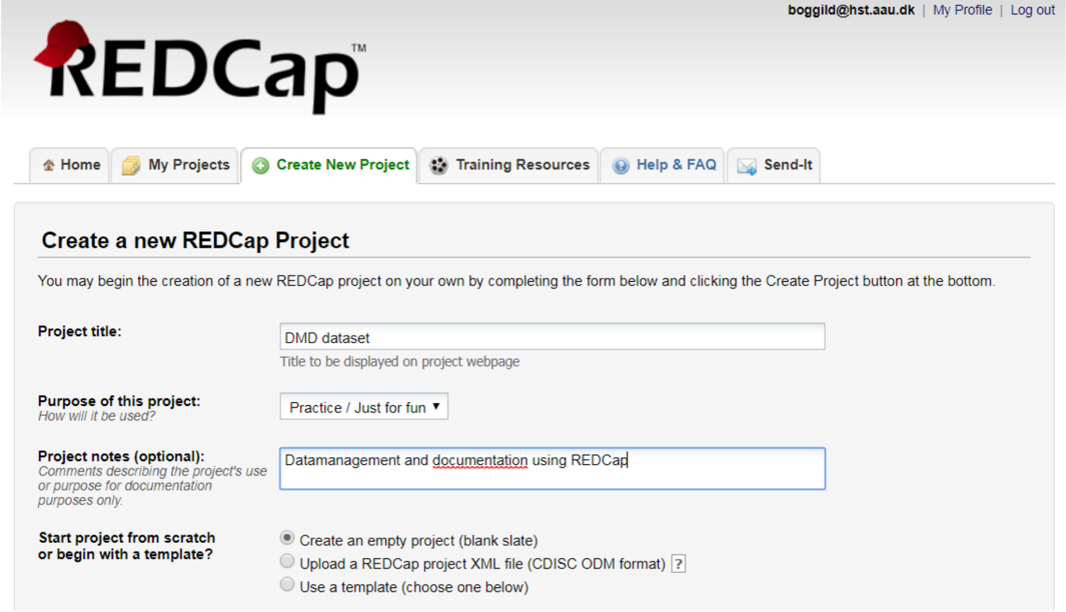
REDCap (an Research Electronic Data Capture system) is specifically designed to comply with demands for secure handling of personal data. It can be used also to implement DMD principles in the setting up of data in relation to both own manual registration of data and the collection of survey data over the Internet.

In REDCap data are collected in "instruments", that can be regarded as single or multiple variables in questionnaires. All instruments are parts of projects, but can be moved across projects if needed. The REDCap consortium hosts a large collection of instruments in the Shared Library, that can be re-used for your own project, for instance if you collect data with a validated scale.

You will need to be granted access to a REDCap installation, probably by an administrator. It is not possible to run REDCap on your own computer (see chapter 2.3).

27 Setting up a project

When having received access you will have a home section with access to all of your projects, but also with access to other functions. This text will only focus on the use of REDCap for setting up instruments in accordance with the principles in this book, for the other functions you are referred to other texts and the many video tutorials on the REDCap homepage.



Create a new REDCap Project

You may begin the creation of a new REDCap project on your own by completing the form below and clicking the Create Project button at the bottom.

Project title: DMD dataset
Title to be displayed on project webpage

Purpose of this project: *How will it be used?* Practice / Just for fun ▾

Project notes (optional): *Comments describing the project's use or purpose for documentation purposes only.* Datamanagement and documentation using REDCap

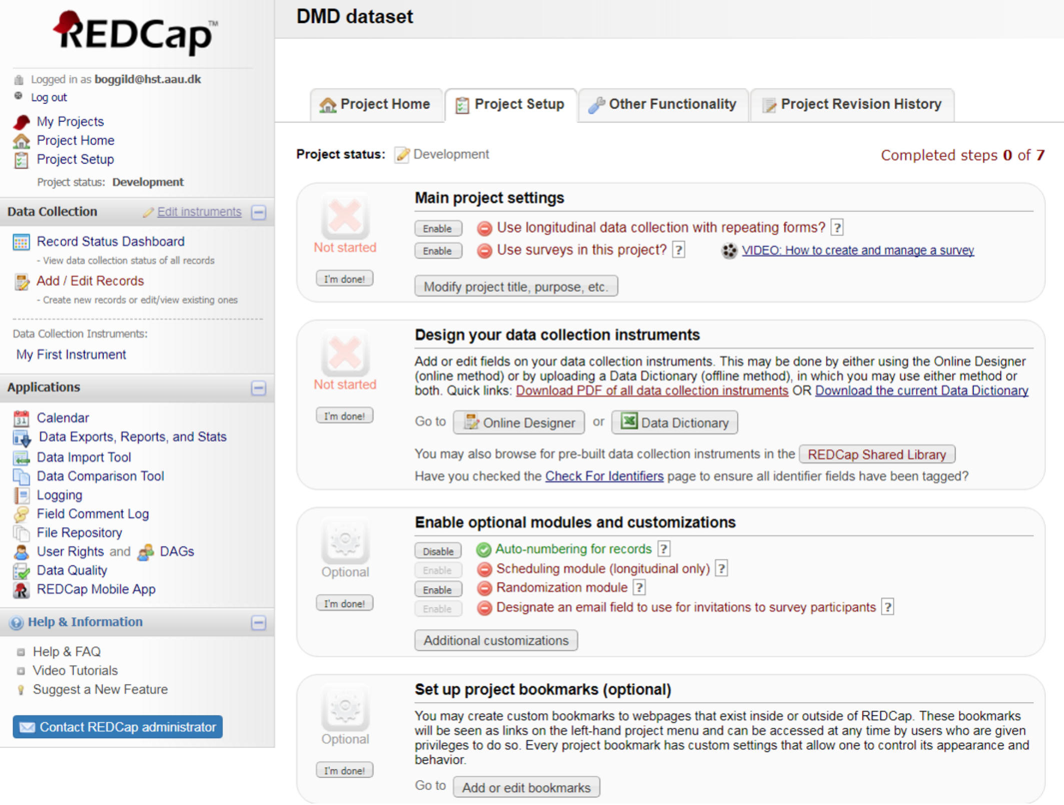
Start project from scratch or begin with a template?
● Create an empty project (blank slate)
○ Upload a REDCap project XML file (CDISC ODM format) ?
○ Use a template (choose one below)

Figure 59 Creating new REDCap project

In the "Create New Project" section pane you create a new project, it will have to be named and you may define it in relation to whether it is for testing or for actual use (this will be changed when going into production). Finally, the pane has room for describing the project in a note, corresponding to the data labels, previous mentioned (see chapter 10).

REDCap will then allow you to chose different possibilities for the project (defining whether the project is using longitudinal data (allowing re-use of the same instruments, for instance, for several visits), will be used to send out questionnaires (survey) and allowing respondents to fill in the data themselves), including different types of modules, setting up bookmarks and to define Users for the project and granting them rights and permissions). In this text only the use of instruments relating to the use of survey as used in preceding chapters will be considered, but the program has many other possibilities.

The project will both include meta-data about the project, and you will then set up the questionnaire within the project (see below). During project setup it is possible to mark each element as "done" when it has been finished, but it is still possible to change the contents of the elements, so it is only for your own focus. When the project has been set up, it should be tested before being moved into production. You can change meta-data as well as the instruments during set up and testing. Once in production, the project should not be changed.



DMD dataset

Project status: Development

Completed steps **0** of **7**

Main project settings

Use longitudinal data collection with repeating forms? ?

Use surveys in this project? ? VIDEO: How to create and manage a survey

Design your data collection instruments

Add or edit fields on your data collection instruments. This may be done by either using the Online Designer (online method) or by uploading a Data Dictionary (offline method), in which you may use either method or both. Quick links: Download PDF of all data collection instruments OR Download the current Data Dictionary

Go to Online Designer or Data Dictionary

You may also browse for pre-built data collection instruments in the REDCap Shared Library

Have you checked the Check For Identifiers page to ensure all identifier fields have been tagged?

Enable optional modules and customizations

Auto-numbering for records ?

Scheduling module (longitudinal only) ?

Randomization module ?

Designate an email field to use for invitations to survey participants ?

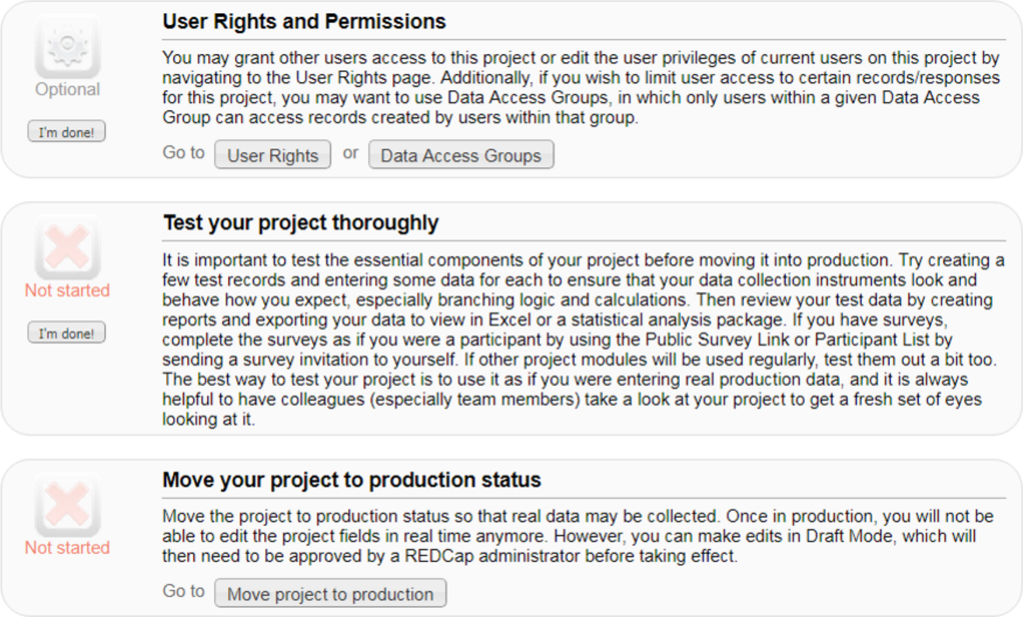
Set up project bookmarks (optional)

You may create custom bookmarks to webpages that exist inside or outside of REDCap. These bookmarks will be seen as links on the left-hand project menu and can be accessed at any time by users who are given privileges to do so. Every project bookmark has custom settings that allow one to control its appearance and behavior.

Go to Add or edit bookmarks

Figure 60 Project setup

Like in EpiData it is possible to grant different permissions for different members of the project. This allows you to have more people keying in information, but keeping possibilities for change of instruments, change of data etc for yourself or a smaller number of administrators. This is defined in the "User Rights and Permissions" further down on the list.



User Rights and Permissions

You may grant other users access to this project or edit the user privileges of current users on this project by navigating to the User Rights page. Additionally, if you wish to limit user access to certain records/responses for this project, you may want to use Data Access Groups, in which only users within a given Data Access Group can access records created by users within that group.

Test your project thoroughly

It is important to test the essential components of your project before moving it into production. Try creating a few test records and entering some data for each to ensure that your data collection instruments look and behave how you expect, especially branching logic and calculations. Then review your test data by creating reports and exporting your data to view in Excel or a statistical analysis package. If you have surveys, complete the surveys as if you were a participant by using the Public Survey Link or Participant List by sending a survey invitation to yourself. If other project modules will be used regularly, test them out a bit too. The best way to test your project is to use it as if you were entering real production data, and it is always helpful to have colleagues (especially team members) take a look at your project to get a fresh set of eyes looking at it.

Move your project to production status

Move the project to production status so that real data may be collected. Once in production, you will not be able to edit the project fields in real time anymore. However, you can make edits in Draft Mode, which will then need to be approved by a REDCap administrator before taking effect.

Figure 61 Project setup (continued)

The left side of the home section also allows you to navigate to other parts of REDCap, for instance for exporting data to statistical programs (see below), to import a data structure, to compare data (see below), define reports, put file (for instance with personal identification) into a repository etc.

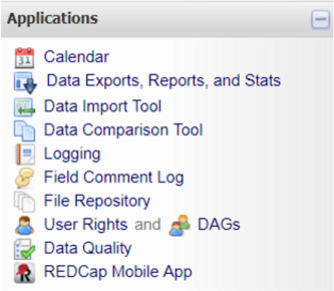


Figure 62 Navigating to other applications in REDCap in the left side

28 Setting up instruments for the project

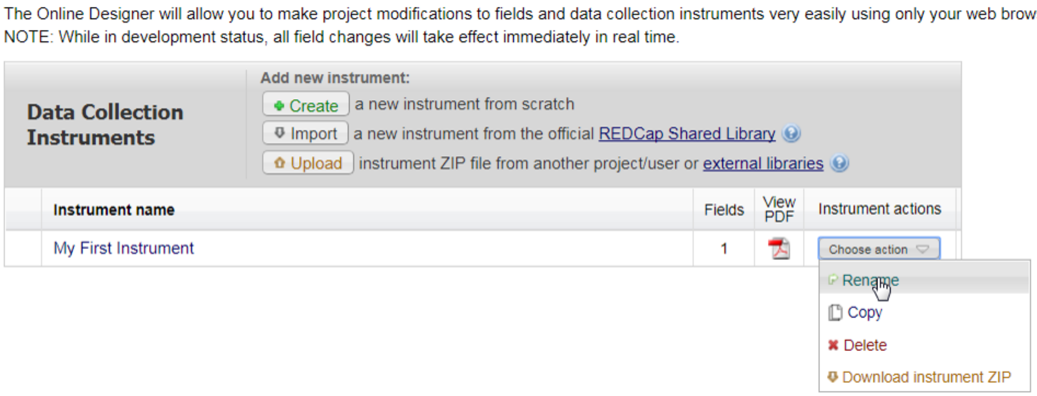
Instruments are single or multiple sets of variables as described in section I (see chapter 5). It is possible to have all variables in one instrument, or variables can be divided into several instruments. This makes it easy re-using existing Shared Library instruments or to re-use questionnaires you have previously developed for other purposes. If using REDCap for survey purposes, this also define how each part of the questionnaire is presented for the respondent.

Every project is born with an instrument that includes only a record ID. This content can't be changed, but the instrument might be renamed for something more descriptive. The record ID will identify the record regardless of whether other identification variables are present or not, and correspond to the identification of the record, previously argued (see chapter 5.4).

It is advisable to collect all information on identification of the participant in its own instrument with just that content. It allows for a very easy way of making data pseudo-anonymous afterwards, as you can export data from each instrument.

Setting up instruments is done in the "online designer", which is described below.

For each instrument defined, it can be renamed, copied for reuse in the project, moved within the project or moved to another project by downloading it in a zipped file.



The Online Designer will allow you to make project modifications to fields and data collection instruments very easily using only your web browser
NOTE: While in development status, all field changes will take effect immediately in real time.

Data Collection Instruments

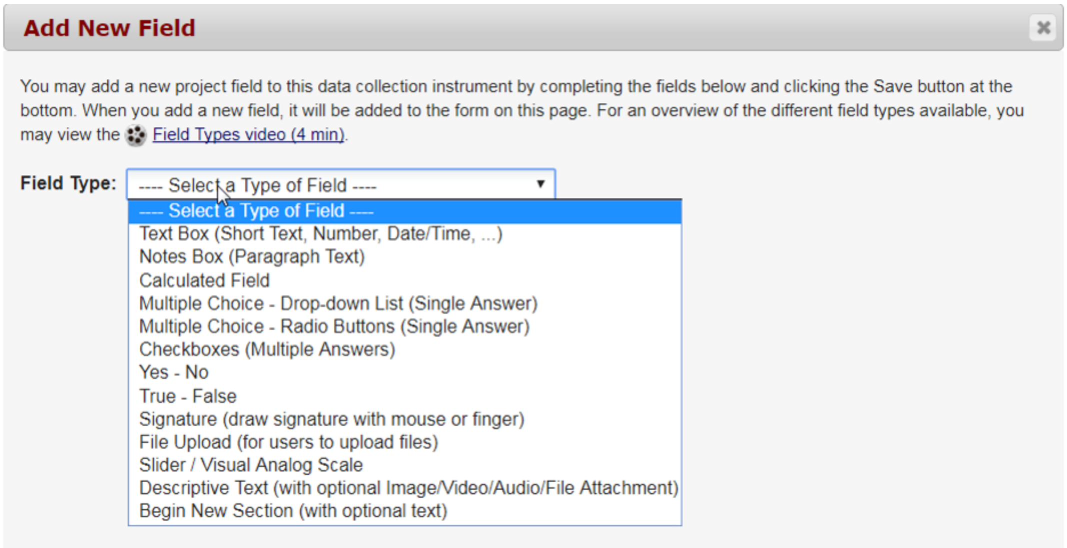
Add new instrument:
Create a new instrument from scratch
Import a new instrument from the official REDCap Shared Library
Upload instrument ZIP file from another project/user or external libraries

Instrument name	Fields	View PDF	Instrument actions
My First Instrument	1		Choose action

Figure 63 The online designer includes description of the instruments in your project

Click on "Create" and define a name for the instrument. Then you will have access to define variables relating to new fields. Fields are of different types and you start by choosing which type you need (corresponding to type of content). Although the first option is called a text box, it is also used for numbers, both ratio-interval and dates. This type of field also allows for setting up check-boxes (Multiple Choice), either as radio buttons (see below)) or drop-down lists. It can also handle multiple answers (checkboxes), and boolean yes-no/true-false buttons. Finally, a VAS (Visual Analogue Scale) ruler is available, when used it record the distance at which the marker was placed.

I'll show how to use only a couple of the possibilities in this text, but the principles work across other types of fields.



Add New Field

You may add a new project field to this data collection instrument by completing the fields below and clicking the Save button at the bottom. When you add a new field, it will be added to the form on this page. For an overview of the different field types available, you may view the Field Types video (4 min).

Field Type: ---- Select a Type of Field ----

---- Select a Type of Field ----
Text Box (Short Text, Number, Date/Time, ...)
Notes Box (Paragraph Text)
Calculated Field
Multiple Choice - Drop-down List (Single Answer)
Multiple Choice - Radio Buttons (Single Answer)
Checkboxes (Multiple Answers)
Yes - No
True - False
Signature (draw signature with mouse or finger)
File Upload (for users to upload files)
Slider / Visual Analog Scale
Descriptive Text (with optional Image/Video/Audio/File Attachment)
Begin New Section (with optional text)

Figure 64 Choice of different types of fields

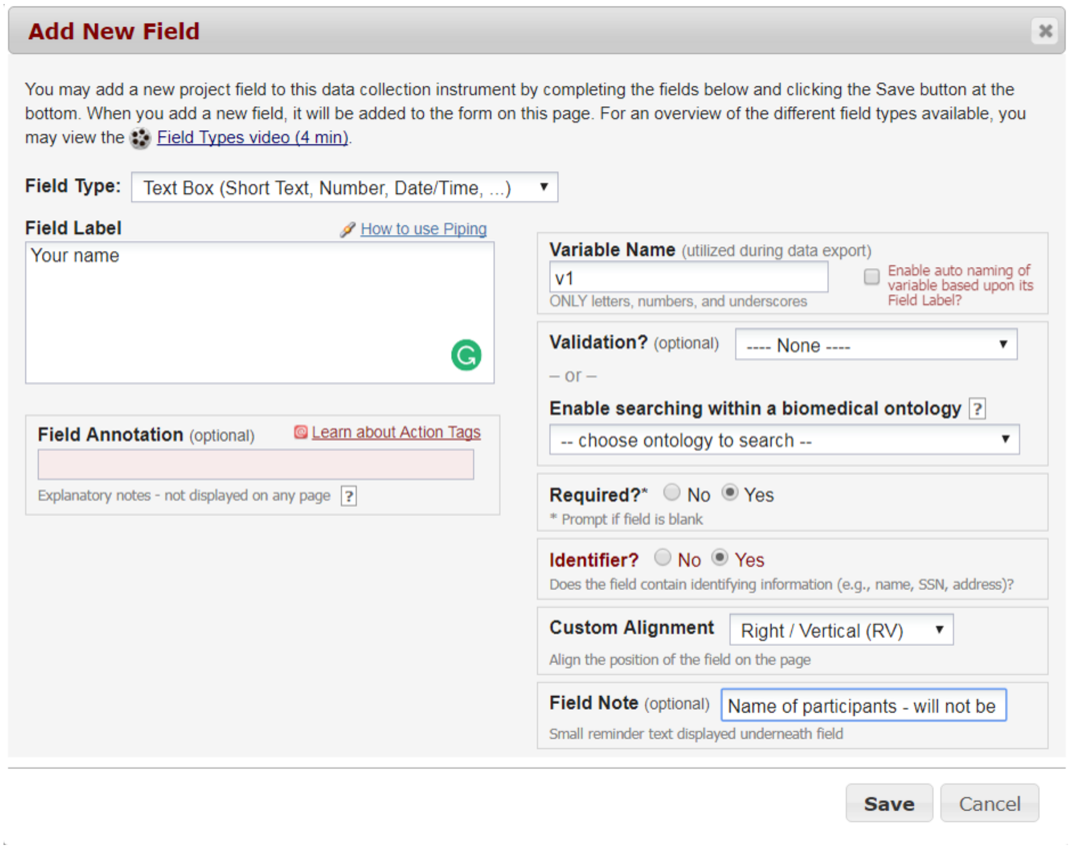
28.1 Setting up a text box (including numbers)

This type of field (variable) is used for both text variables, integer and other text and numbers as dates. When choosing a type, different information can be related to the field, which is saved together with the instrument and possibly reused when exporting the data. The variable is named at the right and should only include letters and numbers. REDCap will suggest a name, but it can be changed. "Field Label" is a longer description of the content of this variable and will serve both as the text given in the questionnaire and the variable label (see chapter 5.2).

It is also possible to set up extra helping text in the questionnaire by using "field notes". This information will be presented as a helping text for the keyer.

A field can be marked as "required" so it has to be filled out (a "must enter" field, that is not possible to proceed without filling it) and/or it can be defined as an identifier, making it easy to identify these identifying fields later on when exporting information, so that they are not exported but left in the project in REDCap.

Custom alignment defines how the field is presented at the screen and how the answers are placed in relation to the text. For VAS scales also defines whether the ruler is placed vertical or horizontal.



Add New Field

You may add a new project field to this data collection instrument by completing the fields below and clicking the Save button at the bottom. When you add a new field, it will be added to the form on this page. For an overview of the different field types available, you may view the Field Types video (4 min).

Field Type: Text Box (Short Text, Number, Date/Time, ...)

Field Label: Your name

Field Annotation (optional)
Explanatory notes - not displayed on any page

Variable Name (utilized during data export): v1
ONLY letters, numbers, and underscores
□ Enable auto naming of variable based upon its Field Label?

Validation? (optional): ---- None ----
– or –
Enable searching within a biomedical ontology: -- choose ontology to search --

Required?* ○ No ● Yes
* Prompt if field is blank

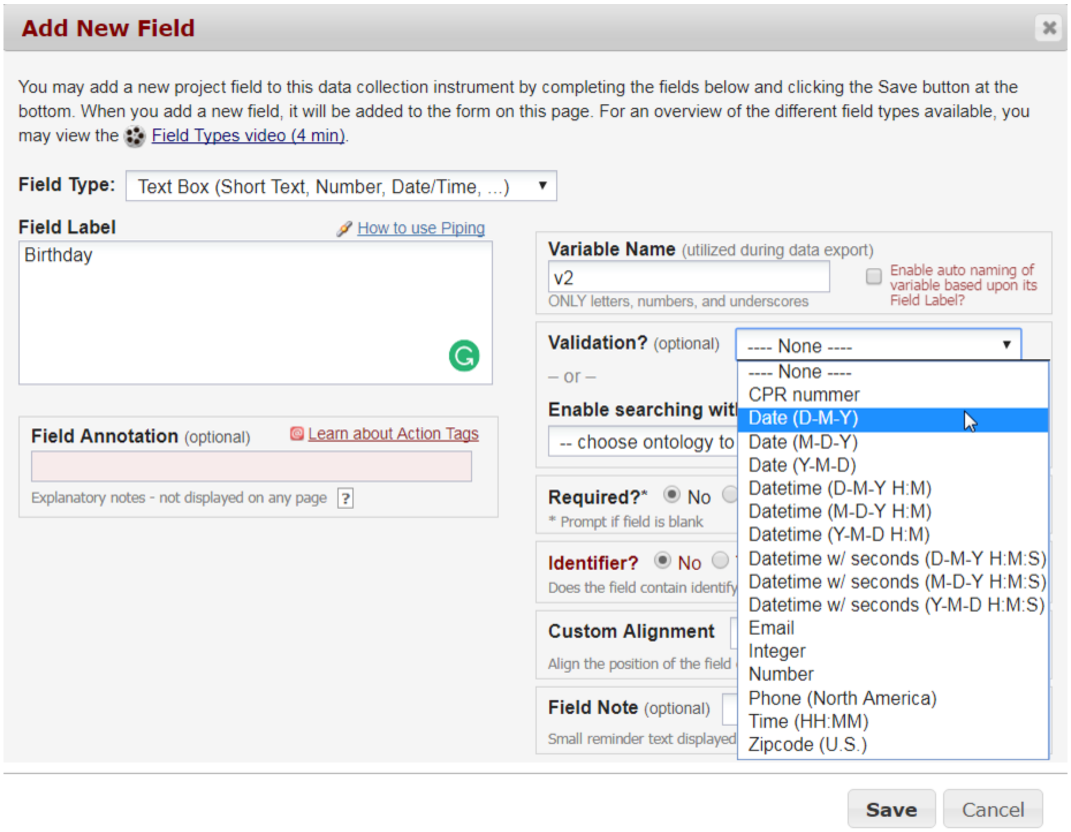
Identifier? ○ No ● Yes
Does the field contain identifying information (e.g., name, SSN, address)?

Custom Alignment: Right / Vertical (RV)
Align the position of the field on the page

Field Note (optional): Name of participants - will not be
Small reminder text displayed underneath field

Figure 65 Choosing the text box type of field in an instrument

In the validation box at the right, the content of the variable can be checked against a number of possibilities (dates, hours, integers and numbers are most used, but others exists). The CPR-number shown in the drop-down menu in this figure has been defined locally to validate the Danish CPR-number (a valid date and a modulus check), it is not a part of the core REDCap, but it display the possibility to change REDCap functionality according to local needs.



Add New Field

You may add a new project field to this data collection instrument by completing the fields below and clicking the Save button at the bottom. When you add a new field, it will be added to the form on this page. For an overview of the different field types available, you may view the Field Types video (4 min).

Field Type: Text Box (Short Text, Number, Date/Time, ...)

Field Label — How to use Piping

Birthday

Variable Name (utilized during data export)

v2

ONLY letters, numbers, and underscores

□ Enable auto naming of variable based upon its Field Label?

Validation? (optional) ---- None ----

- ---- None ----
- CPR nummer
- Date (D-M-Y)
- Date (M-D-Y)
- Date (Y-M-D)
- Datetime (D-M-Y H:M)
- Datetime (M-D-Y H:M)
- Datetime (Y-M-D H:M)
- Datetime w/ seconds (D-M-Y H:M:S)
- Datetime w/ seconds (M-D-Y H:M:S)
- Datetime w/ seconds (Y-M-D H:M:S)
- Email
- Integer
- Number
- Phone (North America)
- Time (HH:MM)
- Zipcode (U.S.)

– or –

Enable searching wit

-- choose ontology to

Field Annotation (optional) — Learn about Action Tags

Explanatory notes - not displayed on any page ?

Required?* ● No ○

* Prompt if field is blank

Identifier? ● No ○

Does the field contain identify

Custom Alignment

Align the position of the field

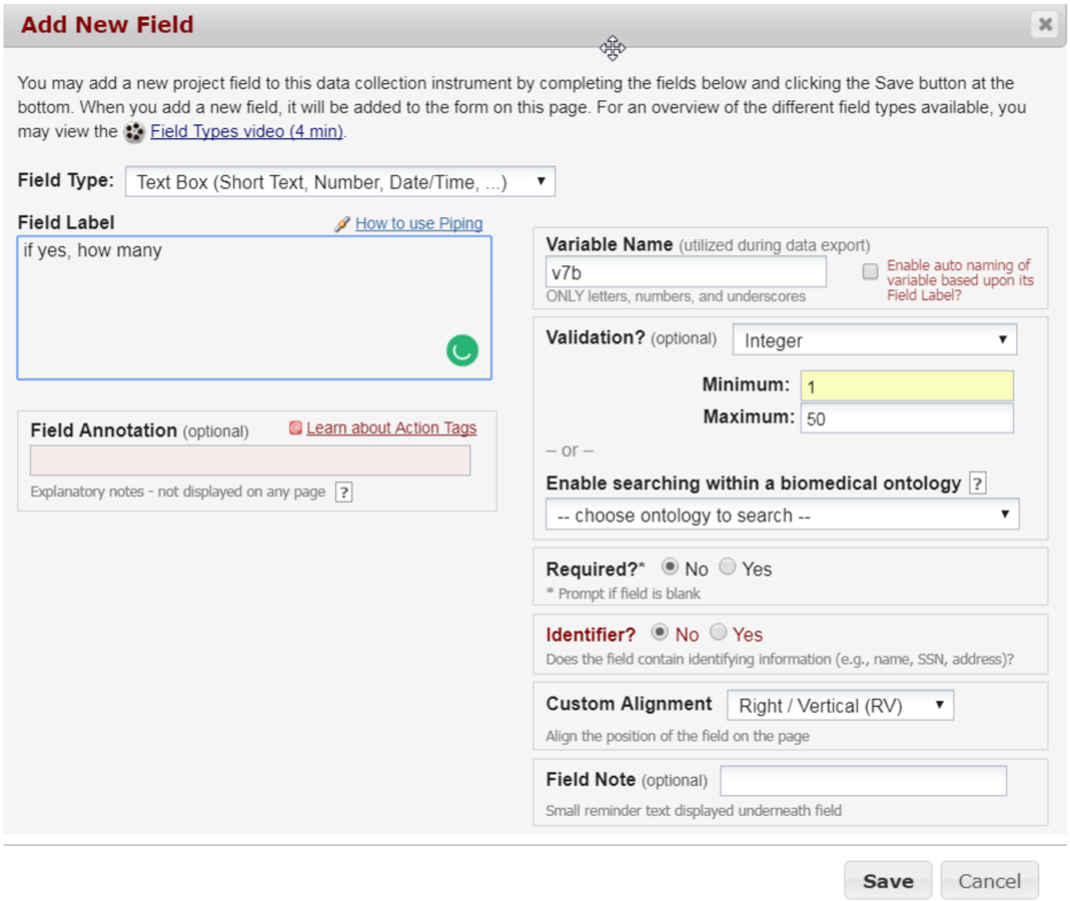
Field Note (optional)

Small reminder text displayed

Figure 66 Types of validation

Validation can also be on integer values, including information on ranges for the variable (for instance setting up possible values for age, height etc), validating also dates, time, that an e-mail includes a @ etc., but unlike in EpiData it is not possible to combine several validation rules, for instance when defining height between 149 and 220 cm and at the same time using 999 as unknown (as shown in chapter 15.2.4).

REDCap has a lot of other possibilities for handling the keying of data, e.g. to suggest a value as the default (using @default in "Action Tags" and defining the default value), using "piping" to extract the value from the answer in one field and putting it into the instruction for another field, using "Field Annotation" to include metadata for a given field etc. This makes REDCap a very powerful tool for use in relation to DMD-purposes.



Add New Field

You may add a new project field to this data collection instrument by completing the fields below and clicking the Save button at the bottom. When you add a new field, it will be added to the form on this page. For an overview of the different field types available, you may view the Field Types video (4 min).

Field Type: Text Box (Short Text, Number, Date/Time, ...)

Field Label How to use Piping

if yes, how many

Field Annotation (optional) Learn about Action Tags

Explanatory notes - not displayed on any page ?

Variable Name (utilized during data export)

v7b

ONLY letters, numbers, and underscores

□ Enable auto naming of variable based upon its Field Label?

Validation? (optional) Integer

Minimum: 1

Maximum: 50

– or –

Enable searching within a biomedical ontology ?

-- choose ontology to search --

Required?* ● No ○ Yes

* Prompt if field is blank

Identifier? ● No ○ Yes

Does the field contain identifying information (e.g., name, SSN, address)?

Custom Alignment Right / Vertical (RV)

Align the position of the field on the page

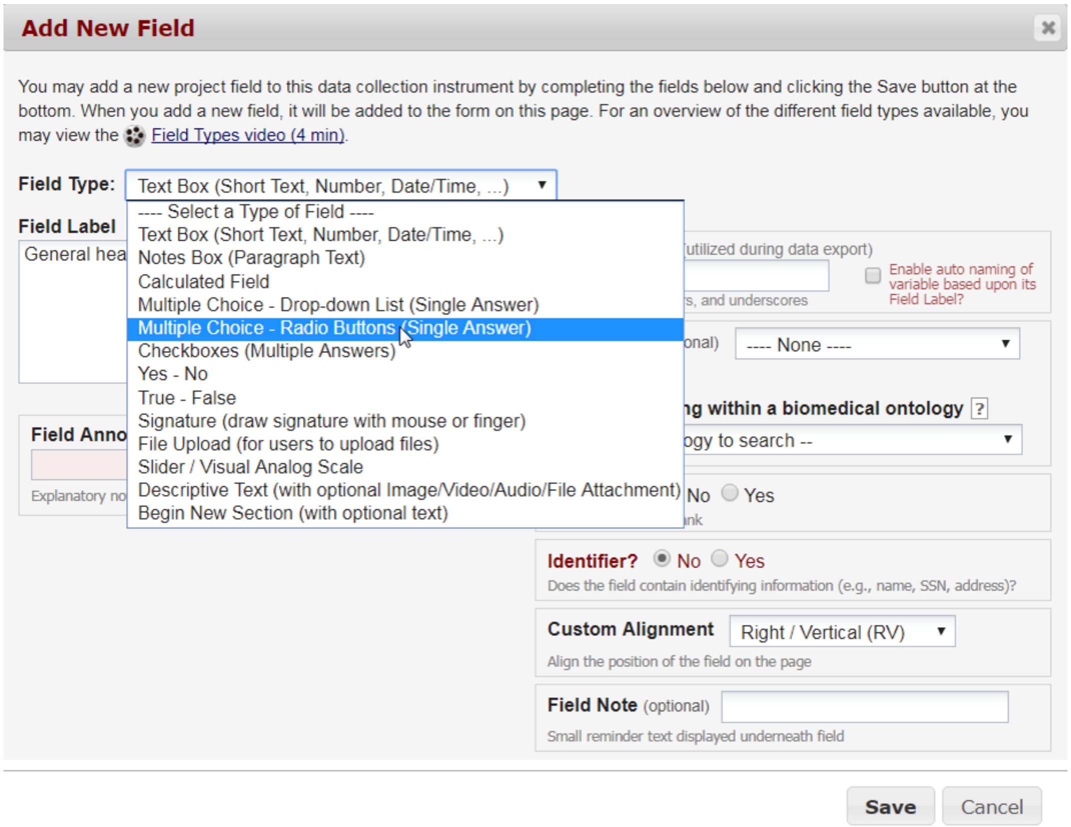
Field Note (optional)

Small reminder text displayed underneath field

Figure 67 Setting up validation in relation to minimum and maximum number allowed

28.2 Setting up a multiple choice list (ordinal and nominal variable)

The Field Type can be selected as a multiple choice lists with radio buttons or a drop-down list for each answer category. The difference is only how the field is shown. Each answer for the question is coded, in principle it is only necessary to have the number, but it is advisable to set up also the corresponding labels at the same time.



Add New Field

You may add a new project field to this data collection instrument by completing the fields below and clicking the Save button at the bottom. When you add a new field, it will be added to the form on this page. For an overview of the different field types available, you may view the Field Types video (4 min).

Field Type: Text Box (Short Text, Number, Date/Time, ...)

Field Label: General hea

Identifier? ● No ○ Yes
Does the field contain identifying information (e.g., name, SSN, address)?

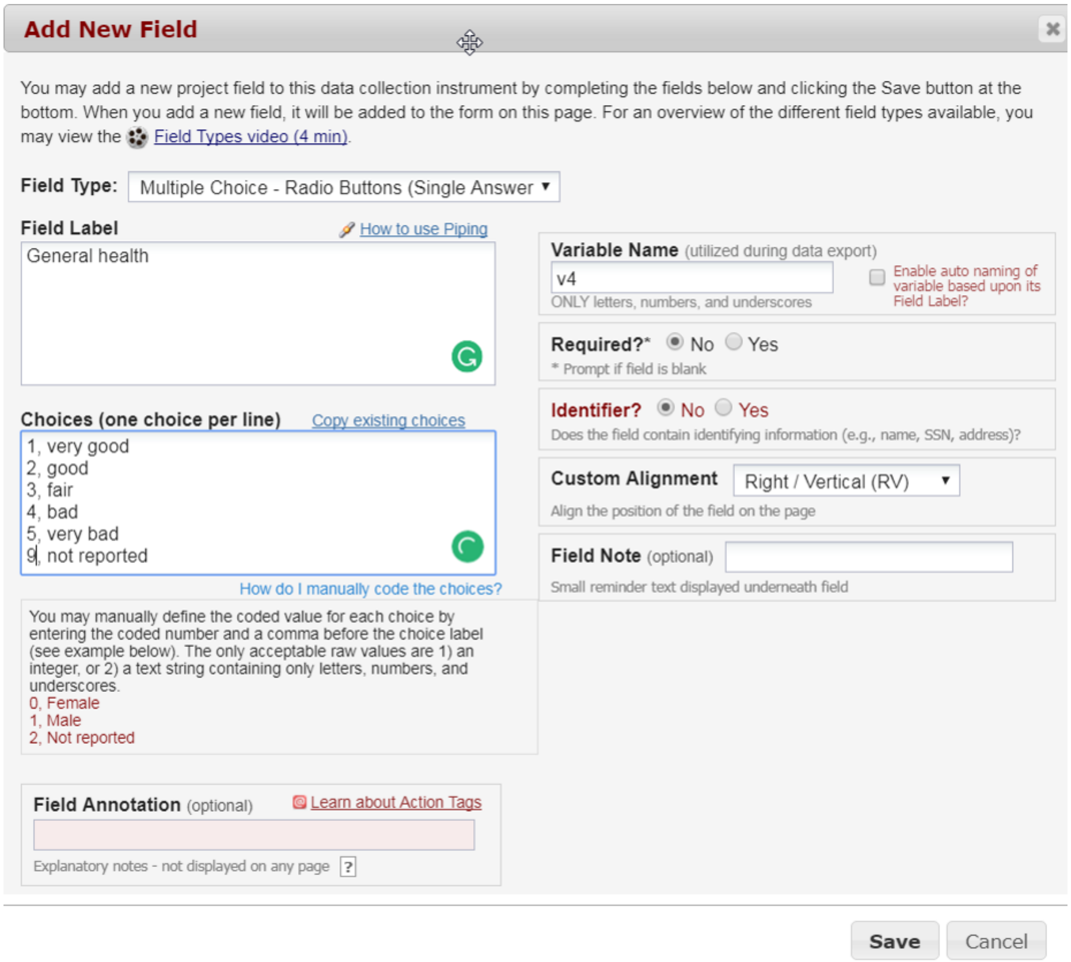
Custom Alignment: Right / Vertical (RV)
Align the position of the field on the page

Field Note (optional)
Small reminder text displayed underneath field

---- Select a Type of Field ----
Text Box (Short Text, Number, Date/Time, ...)
Notes Box (Paragraph Text)
Calculated Field
Multiple Choice - Drop-down List (Single Answer)
Multiple Choice - Radio Buttons (Single Answer)
Checkboxes (Multiple Answers)
Yes - No
True - False
Signature (draw signature with mouse or finger)
File Upload (for users to upload files)
Slider / Visual Analog Scale
Descriptive Text (with optional Image/Video/Audio/File Attachment)
Begin New Section (with optional text)
utilized during data export)
s, and underscores
Enable auto naming of variable based upon its Field Label?
onal)
---- None ----
ng within a biomedical ontology ?
ogy to search --
Field Anno
Explanatory no
No ○ Yes
nk
Save
Cancel

Figure 68 Choosing a radio button field

The answer possibilities are defined with a number and a corresponding text for the value label in the form 1, very good <new line> 2, good <new line> etc. Each number is to be on its own line and the comma is dividing it from the label. It is advisable also to include codes for "9, no answer". The number will be used in the resulting data set with the text used as the value labels.



Add New Field

You may add a new project field to this data collection instrument by completing the fields below and clicking the Save button at the bottom. When you add a new field, it will be added to the form on this page. For an overview of the different field types available, you may view the Field Types video (4 min).

Field Type: Multiple Choice - Radio Buttons (Single Answer ▾

Field Label — How to use Piping

General health

Choices (one choice per line) — Copy existing choices

1, very good
2, good
3, fair
4, bad
5, very bad
9, not reported

How do I manually code the choices?

You may manually define the coded value for each choice by entering the coded number and a comma before the choice label (see example below). The only acceptable raw values are 1) an integer, or 2) a text string containing only letters, numbers, and underscores.
0, Female
1, Male
2, Not reported

Field Annotation (optional) — Learn about Action Tags

Explanatory notes - not displayed on any page ?

Variable Name (utilized during data export): v4
ONLY letters, numbers, and underscores
□ Enable auto naming of variable based upon its Field Label?

Required?* ● No ○ Yes
* Prompt if field is blank

Identifier? ● No ○ Yes
Does the field contain identifying information (e.g., name, SSN, address)?

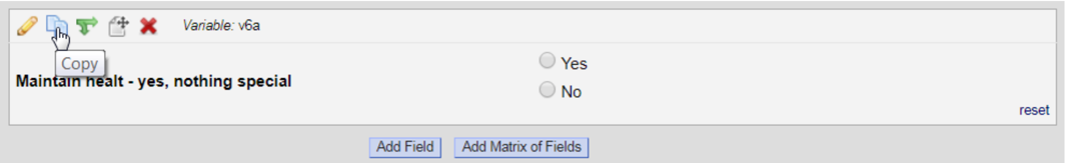
Custom Alignment Right / Vertical (RV) ▾
Align the position of the field on the page

Field Note (optional)
Small reminder text displayed underneath field

Save | Cancel

Figure 69 Radio button field with value labels

If the questionnaire includes several variables with the same set of value labels, the field with its value labels can easily be copied to a new variable with the "copy" instrument in the upper left corner of the field (see figure).



Variable: v6a

Copy

Maintain healt - yes, nothing special

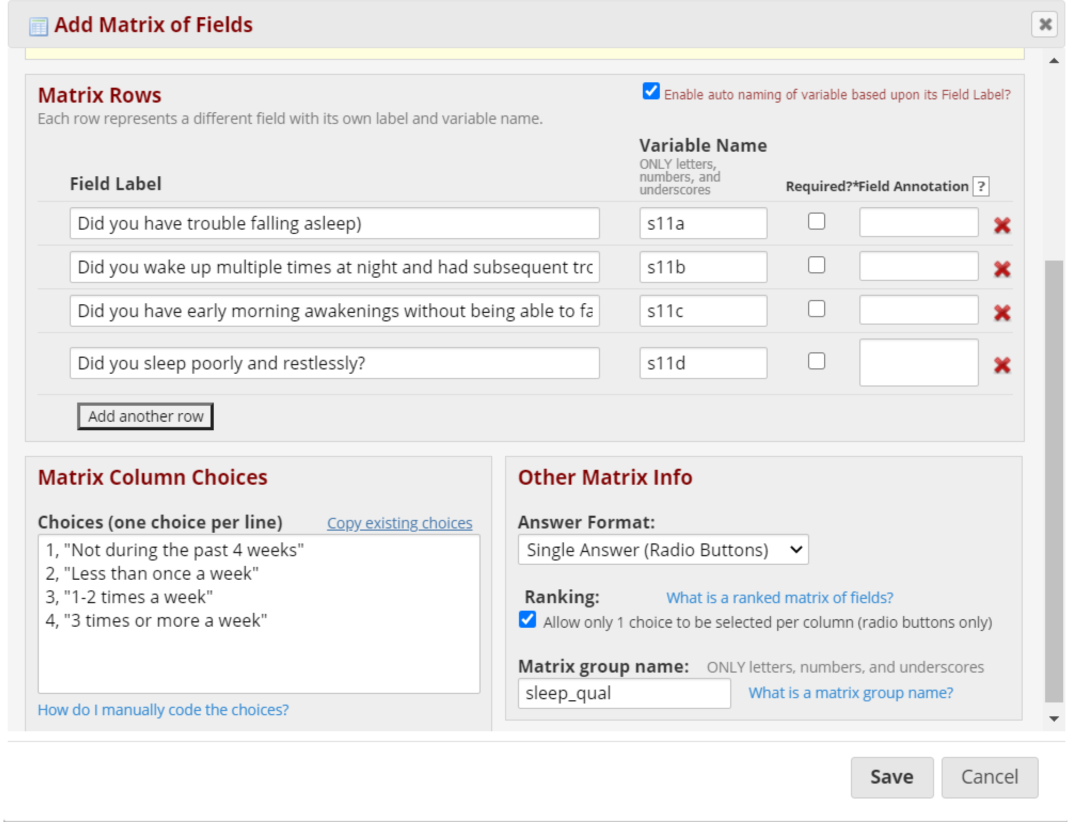
○ Yes
○ No

reset

Add Field | Add Matrix of Fields

Figure 70 Copying a field using the "copy" button

Another possibility if having several items with the same categories is to define the set of variables as a "Matrix of Fields". This will result in a set of lines with each variable text and a set of rows with each possible answer, only given above the matrix and with radio buttons on each line, very useful for setting up e.g. scales. The matrix needs a name, but each variable (corresponding with the line) may be named individually. It is also possible to mark that each column is only allowed to be used for one variable (marked as a cross in "Ranking"). This is relevant if you want the user to rank different statements in each item to each other (it would not be very relevant in the variable shown in the figure, though).



Add Matrix of Fields

Matrix Rows

☑ Enable auto naming of variable based upon its Field Label?

Each row represents a different field with its own label and variable name.

Field Label	Variable Name ONLY letters, numbers, and underscores	Required?*	Field Annotation ?
Did you have trouble falling asleep)	s11a	□	
Did you wake up multiple times at night and had subsequent trc	s11b	□	
Did you have early morning awakenings without being able to fa	s11c	□	
Did you sleep poorly and restlessly?	s11d	□	

Matrix Column Choices

Choices (one choice per line)

1, "Not during the past 4 weeks"
2, "Less than once a week"
3, "1-2 times a week"
4, "3 times or more a week"

Other Matrix Info

Answer Format: Single Answer (Radio Buttons)

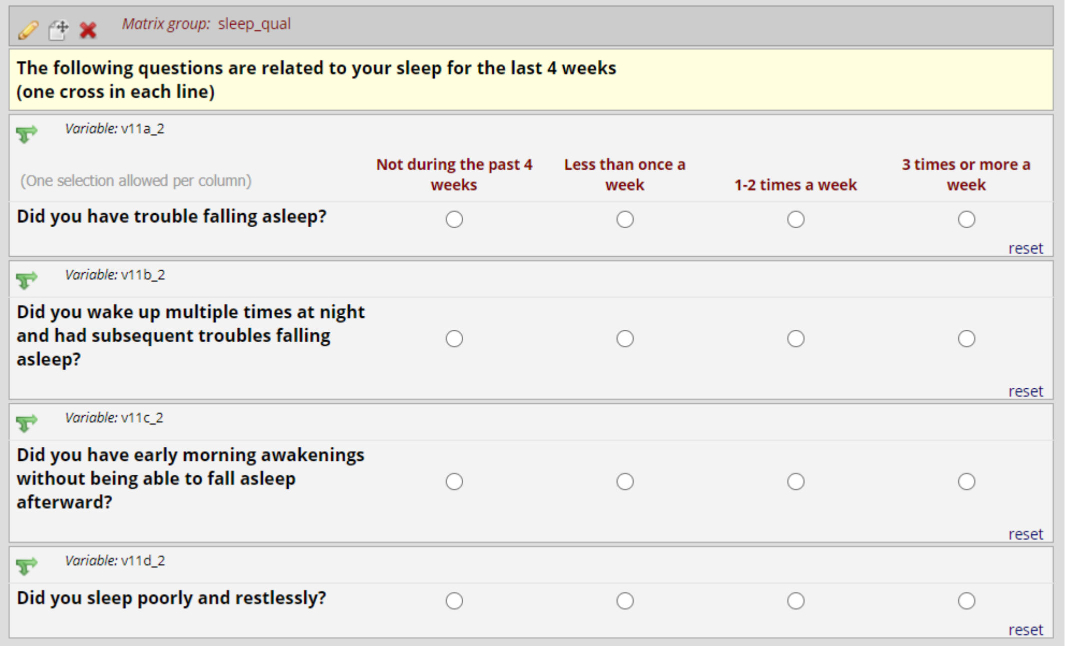
Ranking: ☑ Allow only 1 choice to be selected per column (radio buttons only)

Matrix group name: ONLY letters, numbers, and underscores
sleep_qual

Add another row
Copy existing choices
How do I manually code the choices?
What is a ranked matrix of fields?
What is a matrix group name?
Save
Cancel

Figure 71 Setting up a matrix of fields

The corresponding matrix is shown below.



Matrix group: sleep_qual

The following questions are related to your sleep for the last 4 weeks (one cross in each line)

Variable: v11a_2				
(One selection allowed per column)	**Not during the past 4 weeks**	**Less than once a week**	**1-2 times a week**	**3 times or more a week**
Did you have trouble falling asleep?	○	○	○	○
				reset
Variable: v11b_2				
Did you wake up multiple times at night and had subsequent troubles falling asleep?	○	○	○	○
				reset
Variable: v11c_2				
Did you have early morning awakenings without being able to fall asleep afterward?	○	○	○	○
				reset
Variable: v11d_2				
Did you sleep poorly and restlessly?	○	○	○	○
				reset

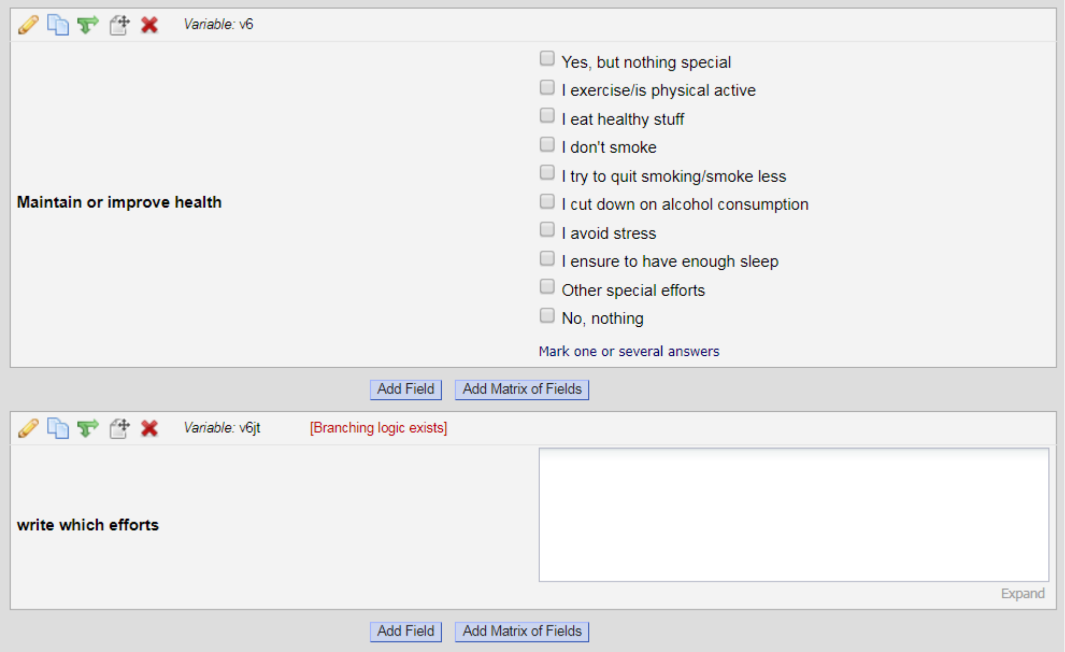
Figure 72 The resulting matrix of fields

28.3 Setting up a variable for multiple answers

It is also possible to define a box with variable that have more than one possible answer, this is called "checkboxes (multiple answers)" and for each check box REDCap automatically defines a set of multiple variables that are each marked as checked or unchecked when filling out the instrument. The variables are named by consecutive numbers (v6(1), v6(2)) for each box in the variable, but with the label attached in REDCap, making it fairly easy to follow the naming.

Using the action tag @noneoftheabove will ensure that if this check box is marked, then the others will be unchecked. This can be used when a set of possible answers are followed by the possibility of "none of the above".

It is also possible to set up branching logic (see next section), so that checking one particular box will open another variable for delivering further information (see figure below)



Variable: v6

Maintain or improve health

□ Yes, but nothing special
□ I exercise/is physical active
□ I eat healthy stuff
□ I don't smoke
□ I try to quit smoking/smoke less
□ I cut down on alcohol consumption
□ I avoid stress
□ I ensure to have enough sleep
□ Other special efforts
□ No, nothing

Mark one or several answers

Variable: v6jt [Branching logic exists]

write which efforts

Figure 73 Setting up a multiple checkbox variable with further branching logic

28.4 Branching and looping

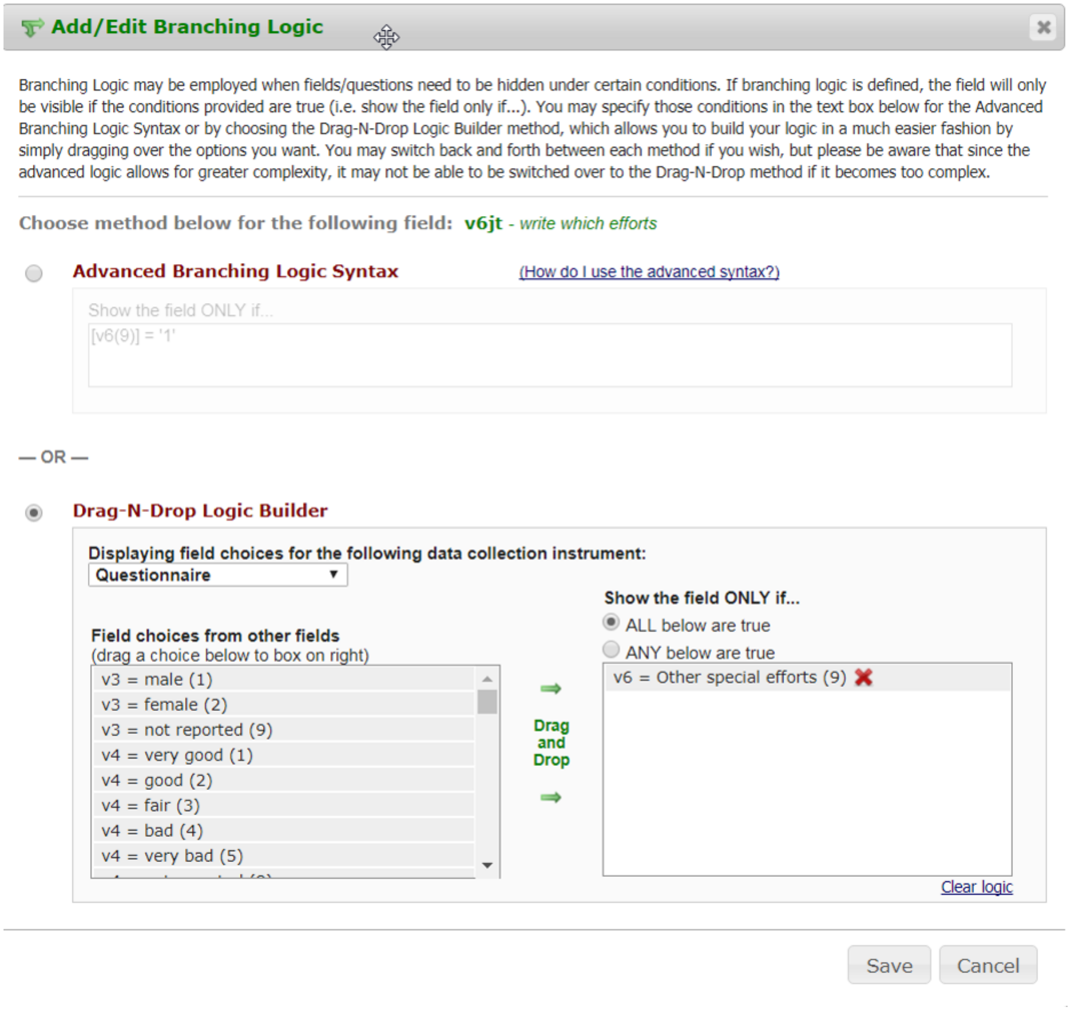
REDCap has the possibility to evaluate the input and change the way instruments and variables are opened. This is called branching and it is available for each defined variable as a green double arrow, that is opening a tool for defining how to proceed. The branching is defined in the variable that is the *end* of the branching (here the field that should be opened).

The branching is possible for all types of fields, but here only shown for the example of a multiple checkbox above.

Defining how the instrument shall work can be made by writing code (called "advanced branching logic syntax") or can be set up using a "drag-n-drop" builder. The relevant fields has in both instances to be defined before using them, so branching is best made after the setting up of all variables in the instrument.

For each variable in the instrument, all possible answers (value labels) are shown in the "drag-n-drop" window to the left, and the relevant answer (here that the variable 6(9) [other special efforts] is marked). If this is true (the variable is marked), the field (variable) is shown. In all other situations the field is not shown. It is possible to define that either all or any of the definitions should be true.

The variable in the instrument will show a red text stating that branching logic exists in the field when branching logics have been set up.



Add/Edit Branching Logic

Branching Logic may be employed when fields/questions need to be hidden under certain conditions. If branching logic is defined, the field will only be visible if the conditions provided are true (i.e. show the field only if...). You may specify those conditions in the text box below for the Advanced Branching Logic Syntax or by choosing the Drag-N-Drop Logic Builder method, which allows you to build your logic in a much easier fashion by simply dragging over the options you want. You may switch back and forth between each method if you wish, but please be aware that since the advanced logic allows for greater complexity, it may not be able to be switched over to the Drag-N-Drop method if it becomes too complex.

Choose method below for the following field: **v6jt** - *write which efforts*

○ **Advanced Branching Logic Syntax** (How do I use the advanced syntax?)

Show the field ONLY if...

[v6(9)] = '1'

— OR —

● **Drag-N-Drop Logic Builder**

Displaying field choices for the following data collection instrument:

Questionnaire ▼

Field choices from other fields
(drag a choice below to box on right)

- v3 = male (1)
- v3 = female (2)
- v3 = not reported (9)
- v4 = very good (1)
- v4 = good (2)
- v4 = fair (3)
- v4 = bad (4)
- v4 = very bad (5)

Show the field ONLY if...

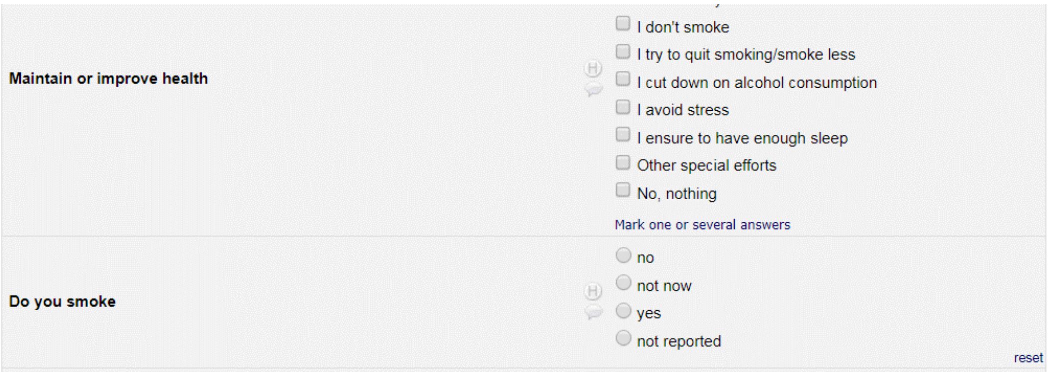
● ALL below are true
○ ANY below are true

v6 = Other special efforts (9) ✖

Figure 74 The branching logic tool

The same branching behaviour could be constructed by using the syntax (in the figure, the syntax giving the same result as in the drag-n-drop builder is shown ([v6(9)]="1")) in the top part of the window. While the drag-n-drop tool is easy to use, the syntax allows for much more complex branching.

Below is the resulting questionnaire in REDCap before filling in the question and if a marking is placed in the field "other special efforts", then a text box, normally not shown, is opened below to allow the keyer to fill in which other efforts.



Maintain or improve health

☐ I don't smoke
☐ I try to quit smoking/smoke less
☐ I cut down on alcohol consumption
☐ I avoid stress
☐ I ensure to have enough sleep
☐ Other special efforts
☐ No, nothing

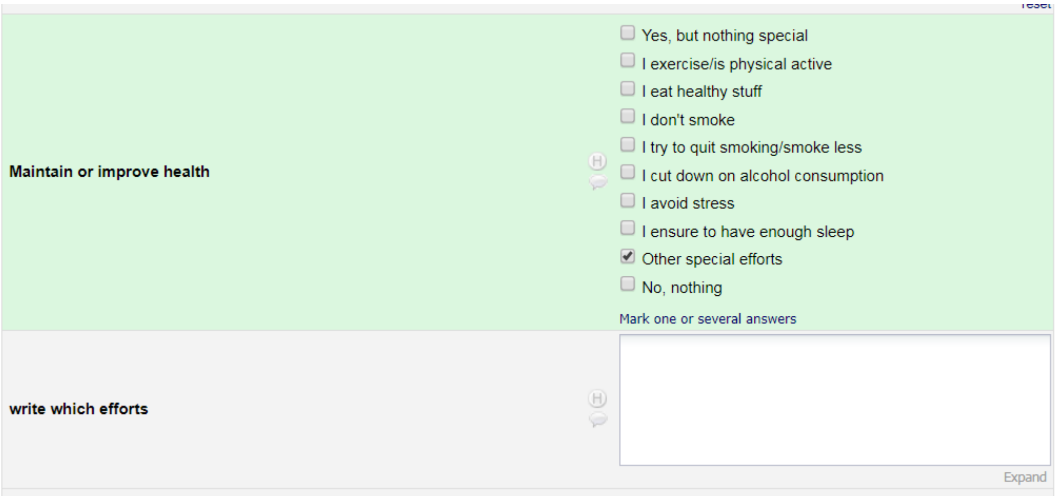
Mark one or several answers

Do you smoke

○ no
○ not now
○ yes
○ not reported

reset

Figure 75 This is how the instrument looks like before marking - the next question is related to smoking



Maintain or improve health

☐ Yes, but nothing special
☐ I exercise/is physical active
☐ I eat healthy stuff
☐ I don't smoke
☐ I try to quit smoking/smoke less
☐ I cut down on alcohol consumption
☐ I avoid stress
☐ I ensure to have enough sleep
☑ Other special efforts
☐ No, nothing

Mark one or several answers

write which efforts

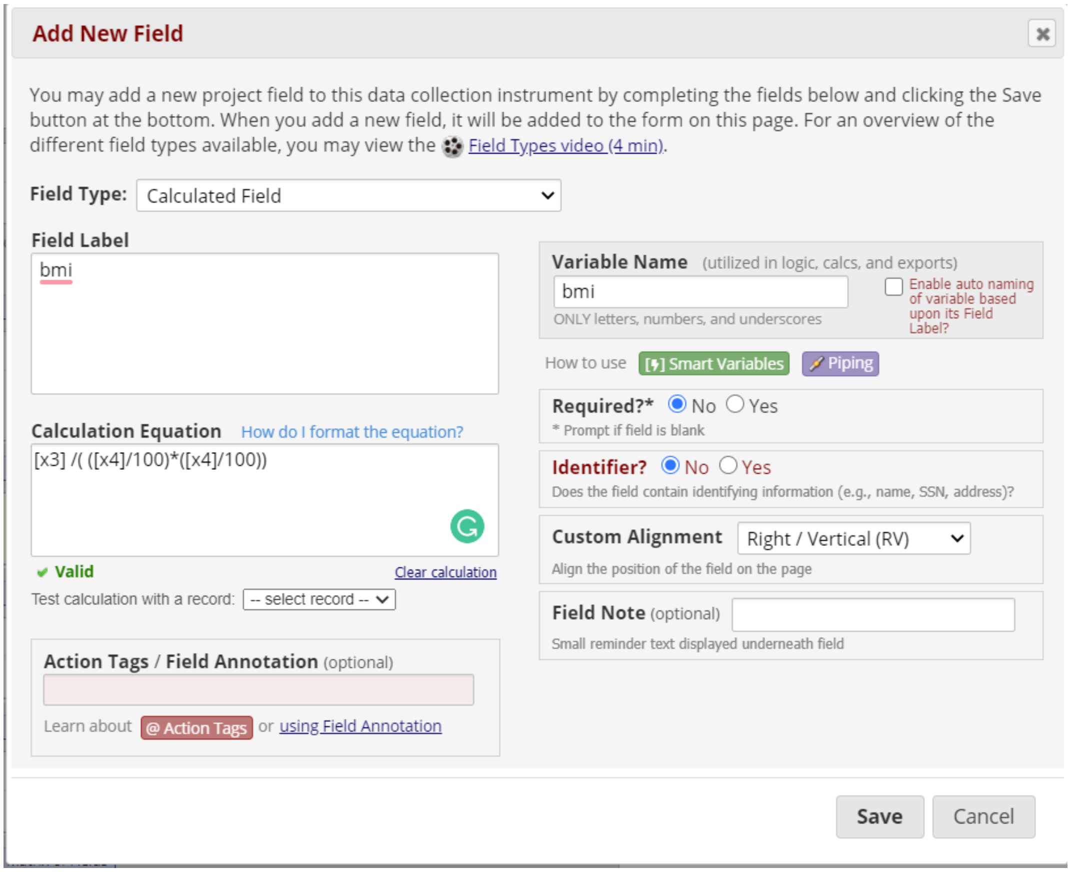
Expand

Figure 76 When marking the answer to which a branch is defined, REDCap opens the field with the branching code

The result of using the designer for setting up the instruments is a ready-to-use tool of questions. Behind this is written a data-structure, that can be downloaded from the project home > data dictionary. It is shown in the form of a comma separated (.csv) file, and consists of a line for each variable in a special sequence. It can be edited or written directly, but it should be made with caution and is normally not advisable. However, it allows for moving data structures between projects. I have written a manual (in Danish) on moving data from an existing STATA file into a REDCap structure using the dictionary file if necessary (typically in order to work on a data set, that was started outside REDCap).

28.5 Setting up calculated variables

REDCap has the ability to make real-time calculations on data entry forms. These calculations will then present the results in non-accessible variables. One example could be the calculation of BMI from self-reported height and weight. In calculations, the names of variables are placed in hard brackets and in this example height is divided by 100 in order to transform height in cm to meter.



Add New Field

You may add a new project field to this data collection instrument by completing the fields below and clicking the Save button at the bottom. When you add a new field, it will be added to the form on this page. For an overview of the different field types available, you may view the Field Types video (4 min).

Field Type: Calculated Field

Field Label: bmi

Calculation Equation: [x3] /(([x4]/100)*([x4]/100))

✓ Valid

Test calculation with a record: -- select record --

Action Tags / Field Annotation (optional)

Learn about @ Action Tags or using Field Annotation

Variable Name (utilized in logic, calcs, and exports): bmi

ONLY letters, numbers, and underscores

☐ Enable auto naming of variable based upon its Field Label?

How to use [⚡] Smart Variables Piping

Required?* ● No ○ Yes

* Prompt if field is blank

Identifier? ● No ○ Yes

Does the field contain identifying information (e.g., name, SSN, address)?

Custom Alignment: Right / Vertical (RV)

Align the position of the field on the page

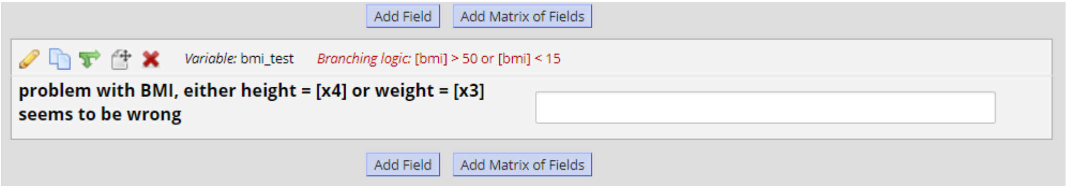
Field Note (optional)

Small reminder text displayed underneath field

Figure 77 Setting up a calculated field for BMI based on self reported weight in variable x3 and height (in cm) in x4

The resulting calculation is shown in red. Most calculation should be made when analysing data later on, but it can help to find implausible values within height and weight that is not found in the validation of these variables itself, for instance for a height of 140 and a weight of 200. They are allowed by the validation of each variable but leads to a BMI of 100.

While the calculated field in itself is not having the possibility of validation, you can make a field using branching logic to alert the keyer on the data - the field showing the problem is made with piping, showing the values of height and weight in the label text. The information is taken from the actual information in x3 and x4 and the keyed number is presented by setting up the variable name in sharp brackets.



Variable: bmi_test *Branching logic:* [bmi] > 50 or [bmi] < 15

problem with BMI, either height = [x4] or weight = [x3] seems to be wrong

Figure 78 Setting up a field in the questionnaire that is only shown when the BMI is outside values and piping the actual values for weight [x3] and height [x4]

28.6 Setting up a Notes Box

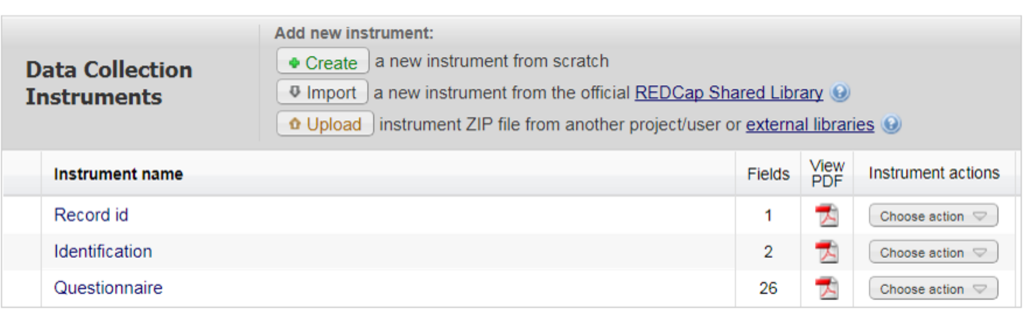
A notes box is a large text box for collecting larger amount of textual information. It is handy for asking the respondent for further information. The size of the box will expand as needed.

28.7 Setting up a New Section

Apart from the variables described above, it is possible to have text information between the variables in the instrument by including a "New Section", that includes a Field label. The text of the Label can be used to introduce new sections etc. and they are not handled as part of the dataset itself. Beware that it is not possible to have the Section as the last field, so you will have to put it between two variables afterwards.

29 Building and testing

Setting up a questionnaire can be seen as collecting a number of instruments that together constituting the questionnaire. The number of fields is shown for each instrument, and it is possible to create new instruments manually or importing/uploading instruments either from other projects or from external libraries. If the questionnaire is divided in sections, it would be advisable to divide the project accordingly.



Data Collection Instruments

Add new instrument:

- Create a new instrument from scratch
- Import a new instrument from the official REDCap Shared Library
- Upload instrument ZIP file from another project/user or external libraries

Instrument name	Fields	View PDF	Instrument actions
Record id	1		Choose action
Identification	2		Choose action
Questionnaire	26		Choose action

Figure 79 The different instruments are shown in the collection. For each instrument the number of fields (variables) are shown to the right

Testing the instruments can be done from the online designer, clicking on "Preview instrument" to the right. Branching logic, calculated fields, piping etc are not however not working in preview. In order to test these parts, you should go to "Add/Edit Records" and click "Add new record" to test the instruments and keying in of data. At this point, the project should be kept in "development" phase, so that data are not seen as real.

It is possible to see a pdf-version of how the instruments will look like from the "Project Setup" pane (within "Design your data collection instruments", click on "Download PDF of all instruments").

By testing the instruments, it will be clear that some changes are needed, and this is done using the "Online Designer" and clicking on the instrument name. The field that is to be edited is opened by clicking on the yellow pencil (or the green arrow if changes are to be made on the branching logic). The field can be deleted by clicking on the red cross.

29.1 Status form

REDCap has a special way of keeping track of the status of each instrument in the form of a traffic light system of incomplete (red), unverified (yellow) and complete (green) instruments. The idea is, that unless otherwise programmed, the keying in of data is either incomplete, if something is missing (fields not entered) or unverified (keyed in, but not overlooked by the administrator). Other possibilities for status are present (partial survey response, complete survey response, many statuses etc).

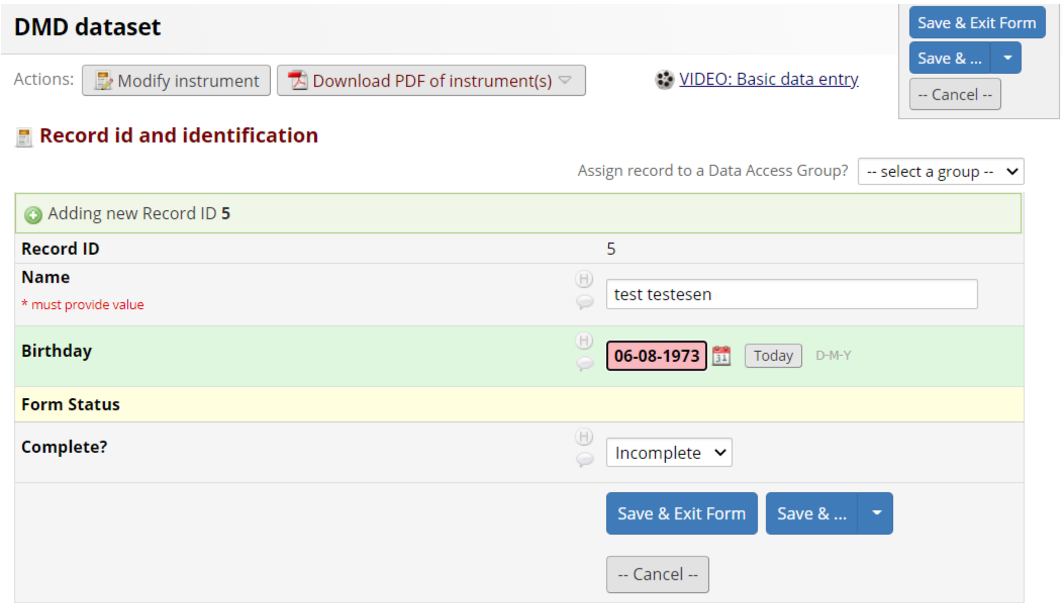
The status of each instrument is collected in automatically created variables, typically named "instrument_complete" and coded 0, 1 or 2 according to the traffic colour.

The keyer should be instructed to either change the form status to unverified or complete depending on whether the administrator want to overlook the information before accepting it in the dataset.

For using REDCap to collect survey information the status is normally set to complete when the collection is complete.

This makes it possible to quickly find instruments for particular records that are not complete and when the administrator is satisfied, they are marked as complete. The instrument is opened by clicking the coloured mark.

The record ends with hitting "Save & Exit Form" or using the possibilities to go to the next form or the next record.



DMD dataset

Record id and identification

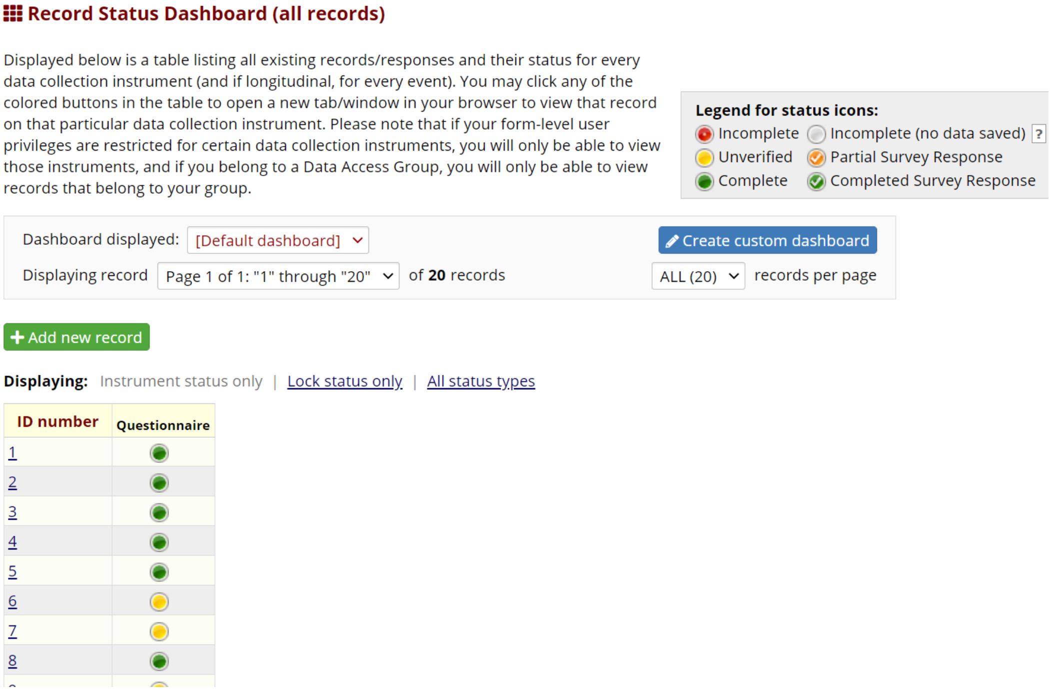
Assign record to a Data Access Group? -- select a group --

Adding new Record ID **5**

Record ID	5
Name * must provide value	test testesen
Birthday	06-08-1973 D-M-Y
Form Status	
Complete?	Incomplete

Figure 80 Each instrument has as the last part a "Form Status", that can be incomplete, unverified or complete.

The form status can also be seen in the "Record Status Dashboard", showing the traffic light information for all records or for the individual record. If the variables were divided in several instruments, there would be a traffic light for each instrument for the records making it easy to identify the missing information.



Record Status Dashboard (all records)

Displayed below is a table listing all existing records/responses and their status for every data collection instrument (and if longitudinal, for every event). You may click any of the colored buttons in the table to open a new tab/window in your browser to view that record on that particular data collection instrument. Please note that if your form-level user privileges are restricted for certain data collection instruments, you will only be able to view those instruments, and if you belong to a Data Access Group, you will only be able to view records that belong to your group.

Legend for status icons:
🔴 Incomplete ⚪ Incomplete (no data saved) ?
🟡 Unverified 🟠 Partial Survey Response
🟢 Complete ✅ Completed Survey Response

Dashboard displayed: [Default dashboard]
Displaying record Page 1 of 1: "1" through "20" of **20** records

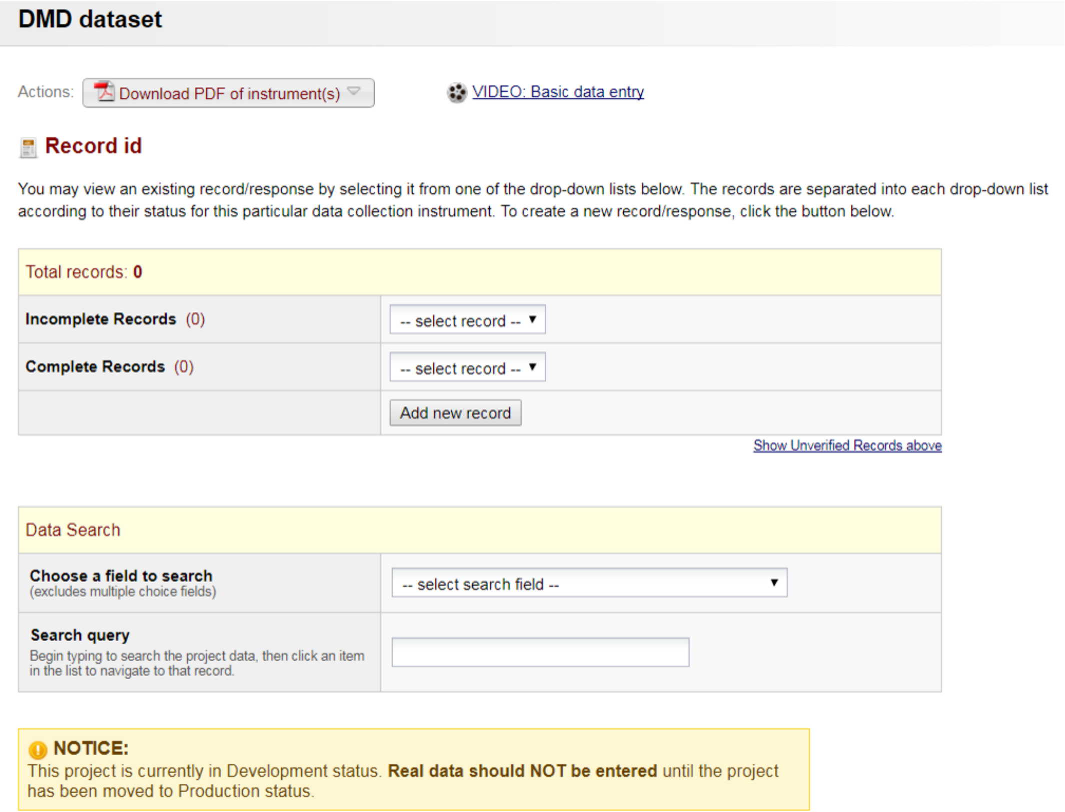
ALL (20) records per page

Displaying: Instrument status only | Lock status only | All status types

ID number	Questionnaire
1	🟢
2	🟢
3	🟢
4	🟢
5	🟢
6	🟡
7	🟡
8	🟢
9	🟡

Figure 81 The Record Status Dashboard for all records, showing that records 6, 7 and 9 are unverified.

The data will not be considered ready until marked as complete by either the keyer or the administrator. This allows for the administrator to evaluate each part of the dataset and decide whether data should be considered complete or whether the administrator should try to handle potential problems. REDCap has the possibility to find incomplete records and evaluate them, marking them as complete when verified.



DMD dataset

Record id

You may view an existing record/response by selecting it from one of the drop-down lists below. The records are separated into each drop-down list according to their status for this particular data collection instrument. To create a new record/response, click the button below.

Total records: **0**

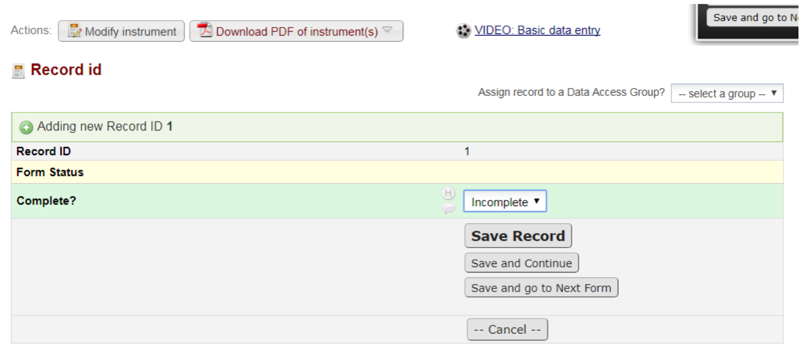
Incomplete Records (0)	-- select record --
Complete Records (0)	-- select record --
	Add new record

Data Search

Choose a field to search (excludes multiple choice fields)	-- select search field --
Search query Begin typing to search the project data, then click an item in the list to navigate to that record.	

NOTICE:
This project is currently in Development status. **Real data should NOT be entered** until the project has been moved to Production status.

Figure 82 Finding the incomplete records and evaluating them



Record id

Assign record to a Data Access Group? -- select a group --

Adding new Record ID **1**

Record ID	1
Form Status	
Complete?	Incomplete

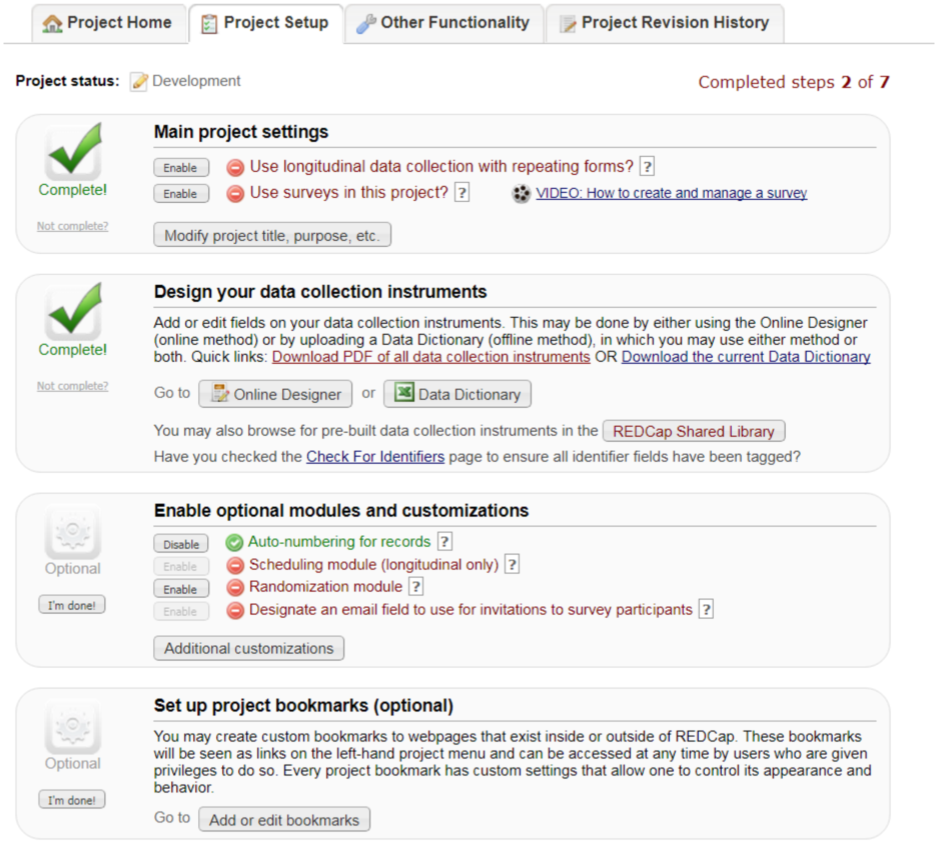
Figure 83 Marking the records as complete after evaluation

29.2 Finalizing the instrument

Once the instruments have been developed and collected you can mark the project as complete on the project set-up pane and move the project into production. It is possible (see figures below) to invoke other tools, but it is not necessary in order to be able to use the instruments in the project.

It is possible to allow the keyer for making notes in relation to, data and I would recommend to have this option marked in order to have keyers to comment on their choices when keying is not in accordance with the instrument. In that instance, when the keyer finds a record with for instance a value outside the anticipated range, it can be commented on and the administrator can look into the data and decide how to code it.

Also to be considered is to enable logging of the data in order to see who and how changes to data were made, and to allow a "today" option in the instruments with dates, so that the keyer can just press a button to record today's date.



Project status: Development

Completed steps 2 of 7

Main project settings

Complete!

Not complete?

Enable ⊖ Use longitudinal data collection with repeating forms? ?

Enable ⊖ Use surveys in this project? ? VIDEO: How to create and manage a survey

Modify project title, purpose, etc.

Design your data collection instruments

Complete!

Not complete?

Add or edit fields on your data collection instruments. This may be done by either using the Online Designer (online method) or by uploading a Data Dictionary (offline method), in which you may use either method or both. Quick links: Download PDF of all data collection instruments OR Download the current Data Dictionary

Go to Online Designer or Data Dictionary

You may also browse for pre-built data collection instruments in the REDCap Shared Library

Have you checked the Check For Identifiers page to ensure all identifier fields have been tagged?

Enable optional modules and customizations

Optional

I'm done!

Disable ✓ Auto-numbering for records ?

Enable ⊖ Scheduling module (longitudinal only) ?

Enable ⊖ Randomization module ?

Enable ⊖ Designate an email field to use for invitations to survey participants ?

Additional customizations

Set up project bookmarks (optional)

Optional

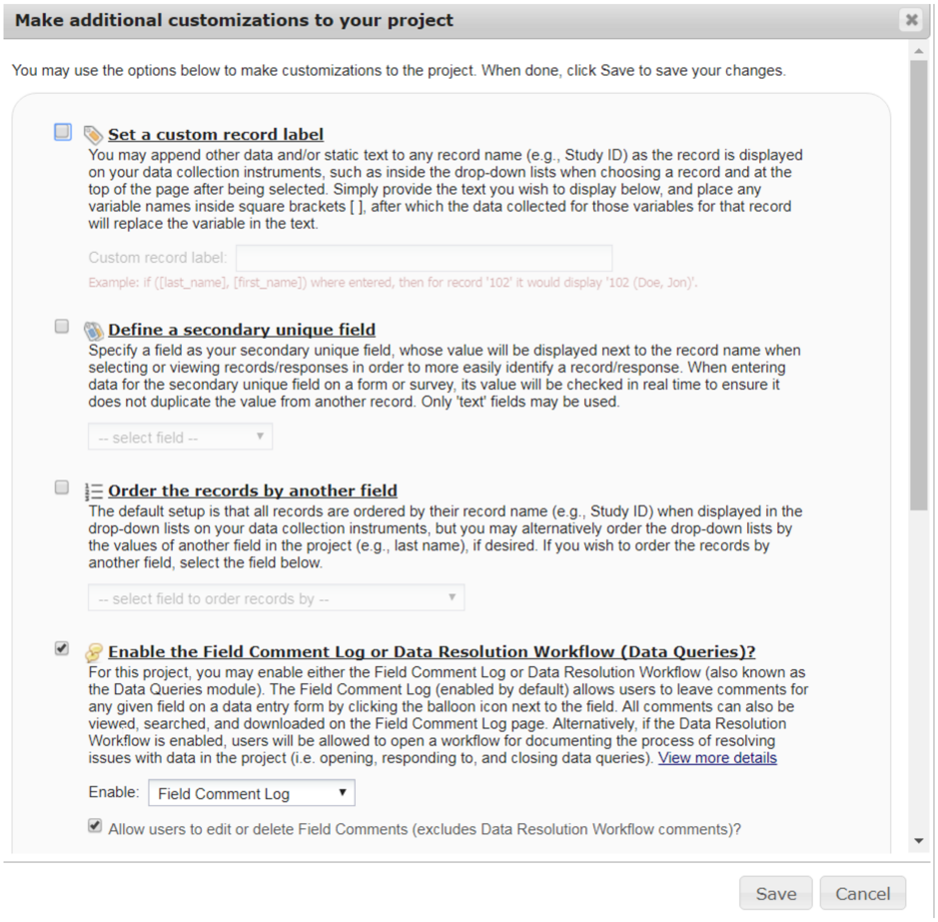
I'm done!

You may create custom bookmarks to webpages that exist inside or outside of REDCap. These bookmarks will be seen as links on the left-hand project menu and can be accessed at any time by users who are given privileges to do so. Every project bookmark has custom settings that allow one to control its appearance and behavior.

Go to Add or edit bookmarks

Figure 84 The project page marked as finished and with the possibility to further include tools to

the project in order to have randomization, an e-mail for sending out a link for the questionnaire etc. (first part of page)



Make additional customizations to your project

You may use the options below to make customizations to the project. When done, click Save to save your changes.

□ **Set a custom record label**
You may append other data and/or static text to any record name (e.g., Study ID) as the record is displayed on your data collection instruments, such as inside the drop-down lists when choosing a record and at the top of the page after being selected. Simply provide the text you wish to display below, and place any variable names inside square brackets [ ], after which the data collected for those variables for that record will replace the variable in the text.

Custom record label:

Example: if ([last_name], [first_name]) where entered, then for record '102' it would display '102 (Doe, Jon)'.

□ **Define a secondary unique field**
Specify a field as your secondary unique field, whose value will be displayed next to the record name when selecting or viewing records/responses in order to more easily identify a record/response. When entering data for the secondary unique field on a form or survey, its value will be checked in real time to ensure it does not duplicate the value from another record. Only 'text' fields may be used.

-- select field --

□ **Order the records by another field**
The default setup is that all records are ordered by their record name (e.g., Study ID) when displayed in the drop-down lists on your data collection instruments, but you may alternatively order the drop-down lists by the values of another field in the project (e.g., last name), if desired. If you wish to order the records by another field, select the field below.

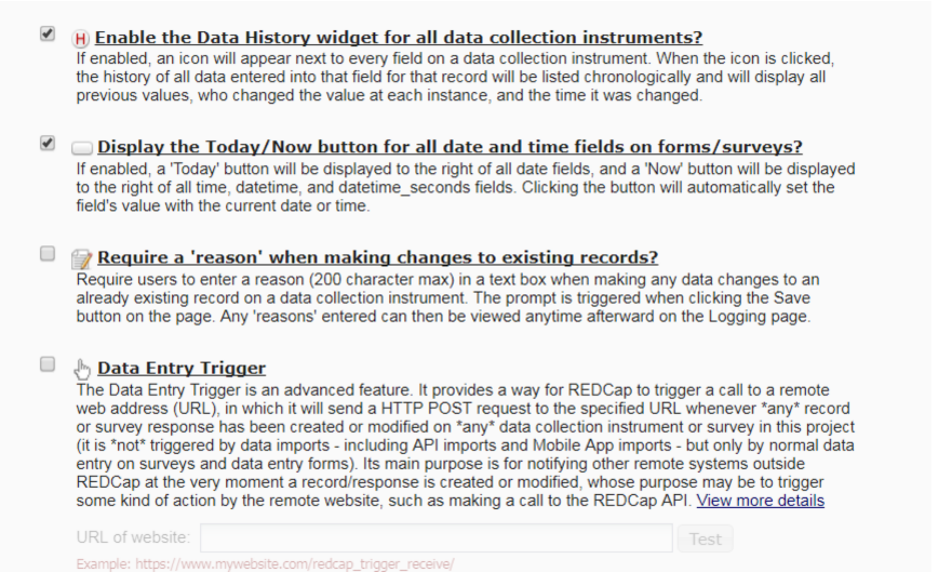
-- select field to order records by --

☑ **Enable the Field Comment Log or Data Resolution Workflow (Data Queries)?**
For this project, you may enable either the Field Comment Log or Data Resolution Workflow (also known as the Data Queries module). The Field Comment Log (enabled by default) allows users to leave comments for any given field on a data entry form by clicking the balloon icon next to the field. All comments can also be viewed, searched, and downloaded on the Field Comment Log page. Alternatively, if the Data Resolution Workflow is enabled, users will be allowed to open a workflow for documenting the process of resolving issues with data in the project (i.e. opening, responding to, and closing data queries). View more details

Enable: Field Comment Log

☑ Allow users to edit or delete Field Comments (excludes Data Resolution Workflow comments)?

Figure 85 The second part of the project page, keep the "enable Field Comment log" in order to allow comments from the keyer for the field



☑ **Enable the Data History widget for all data collection instruments?**
If enabled, an icon will appear next to every field on a data collection instrument. When the icon is clicked, the history of all data entered into that field for that record will be listed chronologically and will display all previous values, who changed the value at each instance, and the time it was changed.

☑ **Display the Today/Now button for all date and time fields on forms/surveys?**
If enabled, a 'Today' button will be displayed to the right of all date fields, and a 'Now' button will be displayed to the right of all time, datetime, and datetime_seconds fields. Clicking the button will automatically set the field's value with the current date or time.

□ **Require a 'reason' when making changes to existing records?**
Require users to enter a reason (200 character max) in a text box when making any data changes to an already existing record on a data collection instrument. The prompt is triggered when clicking the Save button on the page. Any 'reasons' entered can then be viewed anytime afterward on the Logging page.

□ **Data Entry Trigger**
The Data Entry Trigger is an advanced feature. It provides a way for REDCap to trigger a call to a remote web address (URL), in which it will send a HTTP POST request to the specified URL whenever *any* record or survey response has been created or modified on *any* data collection instrument or survey in this project (it is *not* triggered by data imports - including API imports and Mobile App imports - but only by normal data entry on surveys and data entry forms). Its main purpose is for notifying other remote systems outside REDCap at the very moment a record/response is created or modified, whose purpose may be to trigger some kind of action by the remote website, such as making a call to the REDCap API. View more details

Figure 86 The third part of the project page, keep the "Enable the Data History widget" in order to log changes to the data.

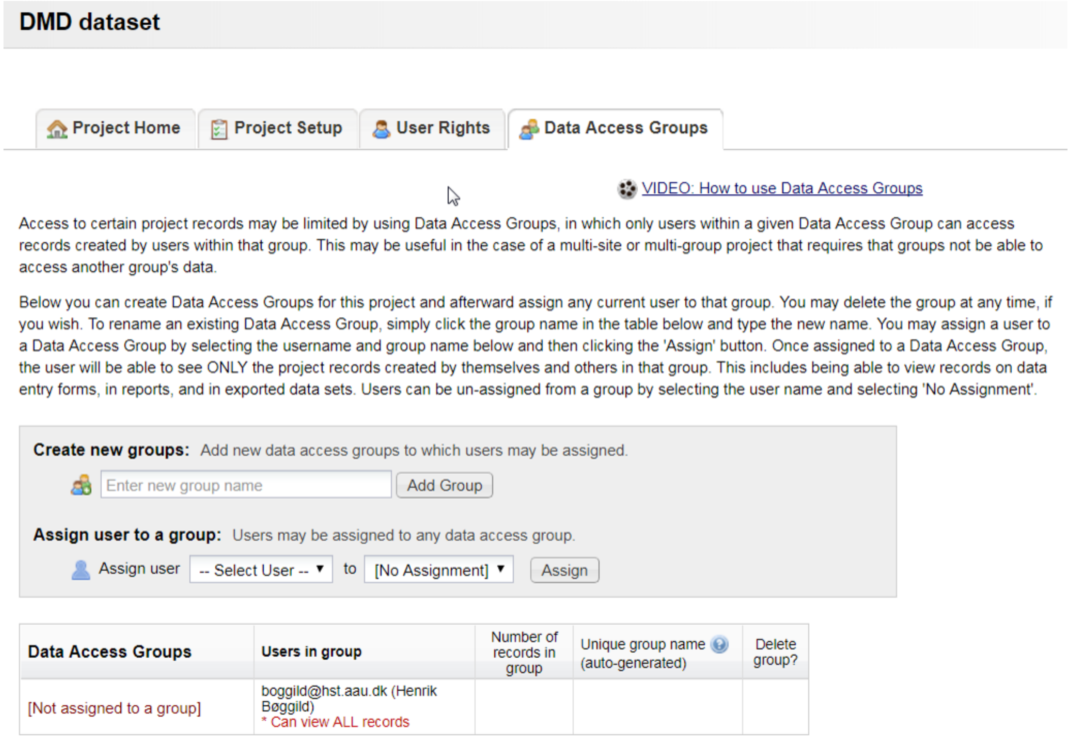
29.3 Setting up data access

REDCap has the possibility to assign users to different groups. By that you can grant administrator rights to some users, while others are only allowed to key in and verify data. This is handled from the User Rights and Data Access Groups panes in the project. It is possible to give rights for a period of time only.

The e-mail of each user is added from "User Rights" and it is possible to create roles with different access rights and assign the users to them.

In Data Access Groups, access is given to a range of records, which can be used to give access for multi-site collaborations, where different users will only have access to view their own data.

Users can be removed from the project from the users right page by clicking on the username and "editing user privileges" and from there clicking on the "remove user" button.



Access to certain project records may be limited by using Data Access Groups, in which only users within a given Data Access Group can access records created by users within that group. This may be useful in the case of a multi-site or multi-group project that requires that groups not be able to access another group's data.

Below you can create Data Access Groups for this project and afterward assign any current user to that group. You may delete the group at any time, if you wish. To rename an existing Data Access Group, simply click the group name in the table below and type the new name. You may assign a user to a Data Access Group by selecting the username and group name below and then clicking the 'Assign' button. Once assigned to a Data Access Group, the user will be able to see ONLY the project records created by themselves and others in that group. This includes being able to view records on data entry forms, in reports, and in exported data sets. Users can be un-assigned from a group by selecting the user name and selecting 'No Assignment'.

Create new groups: Add new data access groups to which users may be assigned.

Enter new group name

Assign user to a group: Users may be assigned to any data access group.

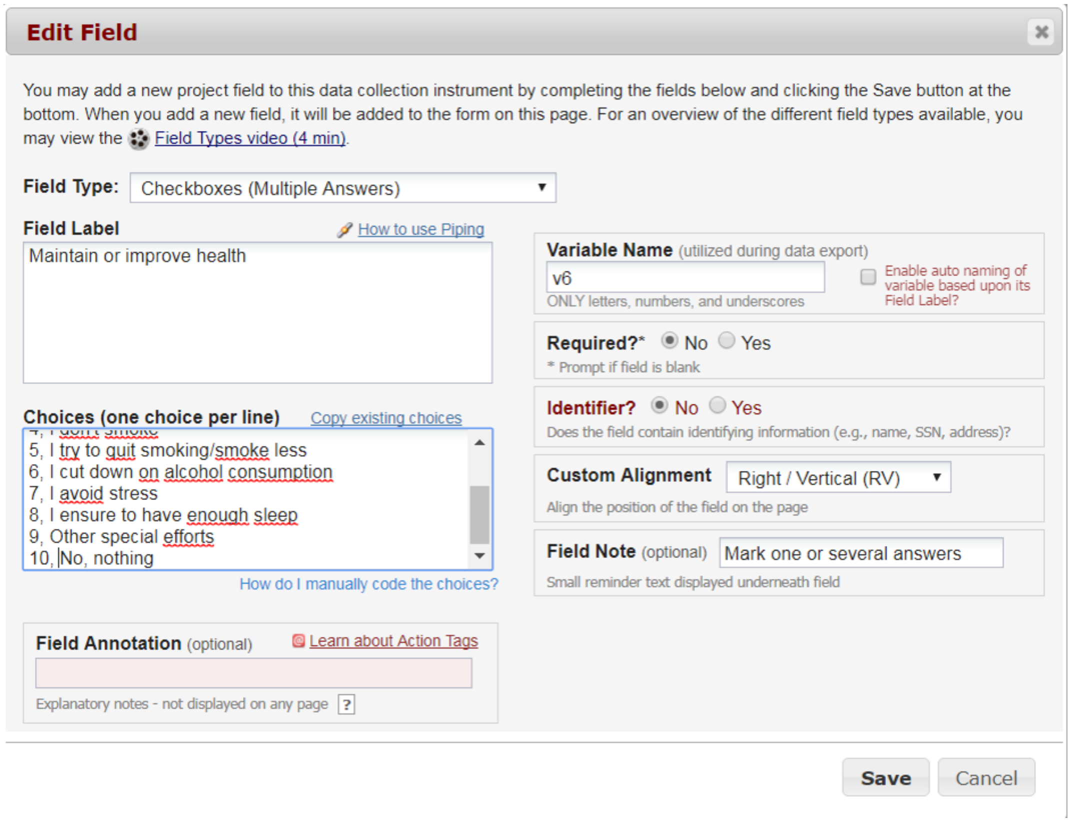
Assign user -- Select User -- to [No Assignment]

Data Access Groups	Users in group	Number of records in group	Unique group name (auto-generated)	Delete group?
[Not assigned to a group]	boggild@hst.aau.dk (Henrik Bøggild) * Can view ALL records			

Figure 87 Setting up user rights and defining data access groups

29.4 Editing instruments

You will probably need to edit your instruments during setting up the questionnaire. Open the instrument where all information related to the field is collected, and change the information, coding or working for the variable. This should be done before moving the project into production.



Edit Field

You may add a new project field to this data collection instrument by completing the fields below and clicking the Save button at the bottom. When you add a new field, it will be added to the form on this page. For an overview of the different field types available, you may view the Field Types video (4 min).

Field Type: Checkboxes (Multiple Answers)

Field Label — How to use Piping

Maintain or improve health

Choices (one choice per line) — Copy existing choices

```
4, I don't smoke
5, I try to quit smoking/smoke less
6, I cut down on alcohol consumption
7, I avoid stress
8, I ensure to have enough sleep
9, Other special efforts
10, No, nothing
```

How do I manually code the choices?

Variable Name (utilized during data export): v6

ONLY letters, numbers, and underscores

□ Enable auto naming of variable based upon its Field Label?

Required?* ● No ○ Yes

* Prompt if field is blank

Identifier? ● No ○ Yes

Does the field contain identifying information (e.g., name, SSN, address)?

Custom Alignment Right / Vertical (RV)

Align the position of the field on the page

Field Note (optional): Mark one or several answers

Small reminder text displayed underneath field

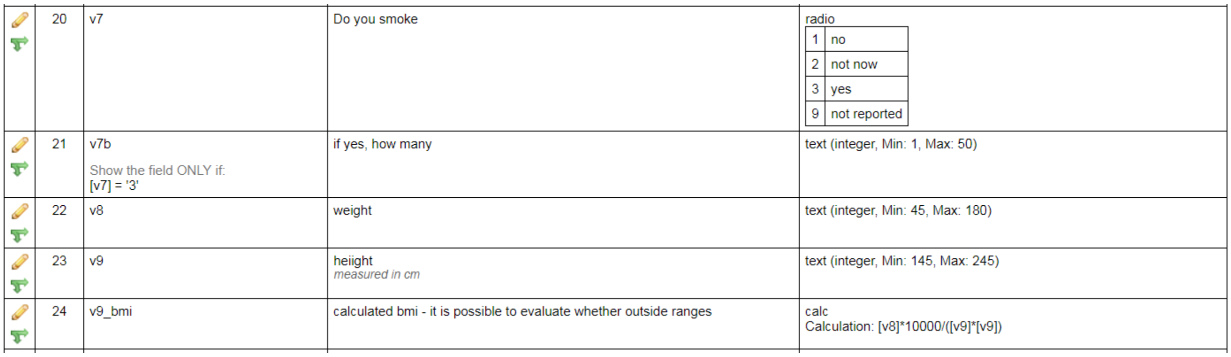
Field Annotation (optional) — Learn about Action Tags

Explanatory notes - not displayed on any page ?

Figure 88 Editing a field

30 Codebook in REDCap

When the instruments have been gathered and the project is ready for production, a codebook can be produced directly from REDCap with information on all variables. This includes field type, variable (field) label and value labels, defined range and branching logic. This is a major documentation of the keying process. The codebook is produced from the "project home" pane and can be printed, for instance in pdf-format.



✎	20	v7	Do you smoke	radio 1 no 2 not now 3 yes 9 not reported
✎	21	v7b Show the field ONLY if: [v7] = '3'	if yes, how many	text (integer, Min: 1, Max: 50)
✎	22	v8	weight	text (integer, Min: 45, Max: 180)
✎	23	v9	heiight *measured in cm*	text (integer, Min: 145, Max: 245)
✎	24	v9_bmi	calculated bmi - it is possible to evaluate whether outside ranges	calc Calculation: [v8]*10000/([v9]*[v9])

Figure 89 Example of part of a codebook

31 Moving into production

When the instruments are ready, tested and documented, the project can be marked as such by moving it into production. This means that changes to the instruments should not be made after moving into production.

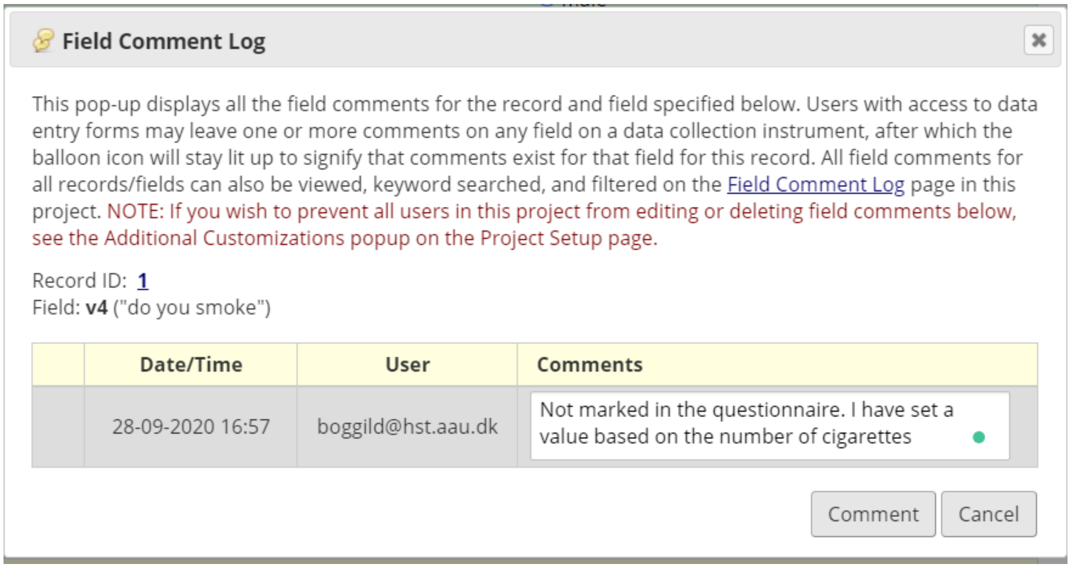
If needed, it is still possible to change the instruments, but be careful as the changes will not be reflected in already entered data, and you may end up having used two different versions to create data. According to the set-up of REDCap you will enter Draft Mode with the project and suggest changes, that need to have approval for changes by an administrator. When approved, changes will take effect on the production project.

During the move into production you can choose to keep or delete all data from the developmental phase. Normally you would choose to delete all data in order to start the keying from scratch.

31.1 Keying in REDCap

When the instruments are in production you or other keyers can key in data. The new record is started in the "Addedit records". When all variables in the instruments are filled in, the "form status" is marked according to the instructions from the administrator (unverified or complete) and the record is finished.

If the feature is turned on for the instrument (see above on finalizing the instrument, chapter 29.2), those keying in can give comments during entering data. This is shown as a small "speech bubble" to the left of the field itself. When clicked, the record and field are marked, the user is identified by their e-mail and the date/time of commenting is stated with a text-field, that is saved with the data. Existing comments are shown by a yellow colour of the speech bubble.



Field Comment Log

This pop-up displays all the field comments for the record and field specified below. Users with access to data entry forms may leave one or more comments on any field on a data collection instrument, after which the balloon icon will stay lit up to signify that comments exist for that field for this record. All field comments for all records/fields can also be viewed, keyword searched, and filtered on the Field Comment Log page in this project. NOTE: If you wish to prevent all users in this project from editing or deleting field comments below, see the Additional Customizations popup on the Project Setup page.

Record ID: 1
Field: **v4** ("do you smoke")

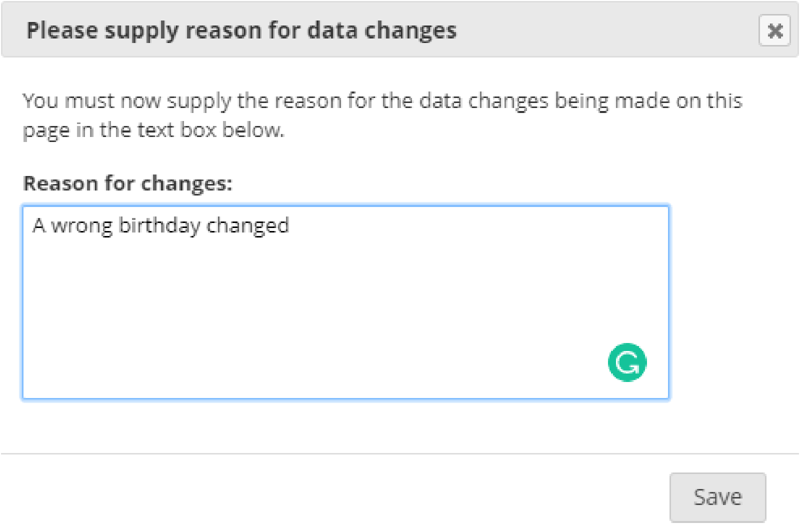
	Date/Time	User	Comments
	28-09-2020 16:57	boggild@hst.aau.dk	Not marked in the questionnaire. I have set a value based on the number of cigarettes

Figure 90 Field comment to a record and variable.

The administrator can find a log of all comments for the instrument in the "Field Comment Log" application at the left of REDCap and evaluate whether the decisions of the keyer is as expected.

31.2 Changing data in REDCap

After keying faults may be discovered and data need to be changed. This should be documented as well. By invoking the option of documenting changes in data (see chapter 29.2), the keyer is presented with a box asking for a reason for the change in data when changing data already keyed in. The box appears when the new data is saved. This is relevant for documenting the need for changing a keyed in information, whether by the keyer or (more often) the administrator. A log of changes can be produced as documentation.



Please supply reason for data changes

You must now supply the reason for the data changes being made on this page in the text box below.

Reason for changes:

A wrong birthday changed

Figure 91 REDCap has the possibility to invoke asking the keyer to supply reason for changes in data

32 Double data entry

I have argued that data should be double entered (see 7.3). REDCap do not have a built in simple, automatic way of double entering and comparing data, but the administrator of REDCap can install a double data entry module, and two persons are then being given rights to re-enter and compare records. The two records can be merged after verification.

A more simple way of doing double data entry is to use the same set of instruments to copy the project, create a second set of data, using the same unique identifier, and then compare the two datasets afterwards, for instance using the statistical program.

Other modules include a Data Quality application, where rules can be build for checking the data for errors during collection.

33 Exporting and documenting data

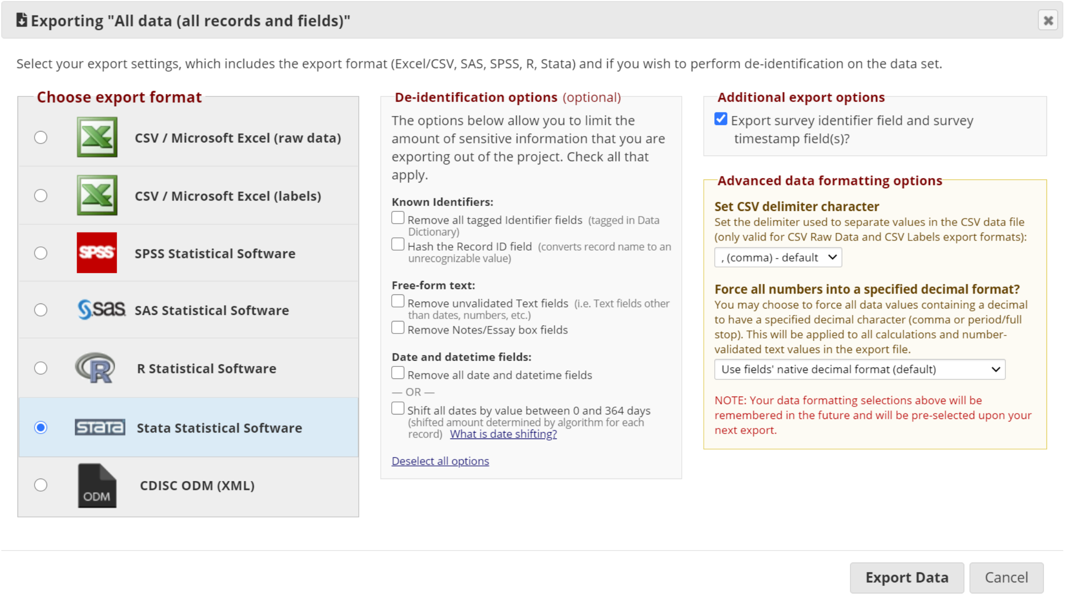
When data has been keyed in and verified, they can be documented by creating reports of each variable. Variables can also be compared ("Check data quality"), errors can be changed at this point directly in the data in REDCap and this is automatically documented.

Data can then be exported for use in statistical software (CSV (comma-separated variables, either with or without the labels), SPSS, SAS, R or STATA) or a CDISC ODM format, a special standard for data sharing (cdisc.org).

Exporting data will in most instances produce two files, one with the data and one with the labels. For STATA, R and SAS for instance, exporting produces a dataset in CSV-format (.csv) and a program file (do-file in STATA, SAS-file in SAS, R-file in R). Both files are needed and should be downloaded. Then the do-file should be opened in STATA and run to read the data.

In SPSS three files are produced, a mapper file, a SPS-program file and again the data in CSV-format.

During export it is possible to remove tagged identifier fields, to hash the record field into a recognizable value, to remove text files and notes and to either remove dates or shifting them (scrambling). All these possibilities are for the possibility to de-identify the data before exporting.



Exporting "All data (all records and fields)"

Select your export settings, which includes the export format (Excel/CSV, SAS, SPSS, R, Stata) and if you wish to perform de-identification on the data set.

Choose export format

○ CSV / Microsoft Excel (raw data)

○ CSV / Microsoft Excel (labels)

○ SPSS Statistical Software

○ SAS Statistical Software

○ R Statistical Software

● Stata Statistical Software

○ CDISC ODM (XML)

De-identification options (optional)

The options below allow you to limit the amount of sensitive information that you are exporting out of the project. Check all that apply.

Known Identifiers:

□ Remove all tagged Identifier fields (tagged in Data Dictionary)

□ Hash the Record ID field (converts record name to an unrecognizable value)

Free-form text:

□ Remove unvalidated Text fields (i.e. Text fields other than dates, numbers, etc.)

□ Remove Notes/Essay box fields

Date and datetime fields:

□ Remove all date and datetime fields

— OR —

□ Shift all dates by value between 0 and 364 days (shifted amount determined by algorithm for each record) What is date shifting?

Deselect all options

Additional export options

☑ Export survey identifier field and survey timestamp field(s)?

Advanced data formatting options

Set CSV delimiter character

Set the delimiter used to separate values in the CSV data file (only valid for CSV Raw Data and CSV Labels export formats):

, (comma) - default

Force all numbers into a specified decimal format?

You may choose to force all data values containing a decimal to have a specified decimal character (comma or period/full stop). This will be applied to all calculations and number-validated text values in the export file.

Use fields' native decimal format (default)

NOTE: Your data formatting selections above will be remembered in the future and will be pre-selected upon your next export.

Figure 92 Exporting data from REDCap

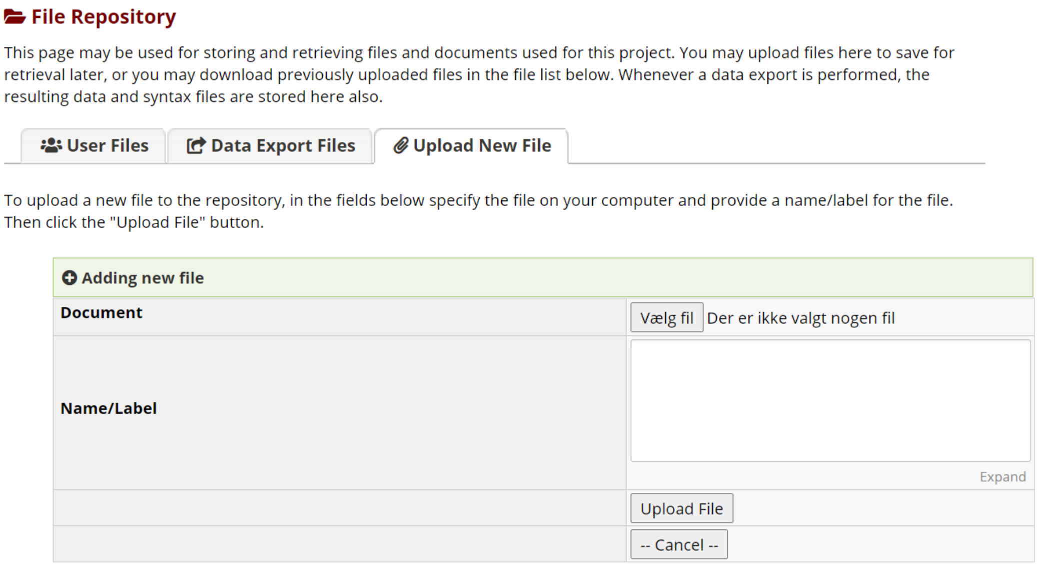
After exporting the original data is still in REDCap.

34 Archiving in RedCap

When data collection are complete and data have been exported the project can move to inactive status (from which it can easily be moved back into production), it can be archived (from which it can also be un-archived if needed), or it can be deleted.

As REDCap documents access, it may be relevant to keep identifiable data inside REDCap. This would naturally be relevant for the keyed dataset themselves, but may also include for instance identifiable data collected through the process.

From the application "File Repository" you can upload files and they will be subject to logging, so it is possible to see who have accessed the data.



File Repository

This page may be used for storing and retrieving files and documents used for this project. You may upload files here to save for retrieval later, or you may download previously uploaded files in the file list below. Whenever a data export is performed, the resulting data and syntax files are stored here also.

To upload a new file to the repository, in the fields below specify the file on your computer and provide a name/label for the file. Then click the "Upload File" button.

Adding new file

Document	Vælg fil Der er ikke valgt nogen fil
Name/Label	

Figure 93 File repository with the possibility to have access to files covered by logging

REDCap has other possibilities, that has not been covered in this text.

Part IV

Data management and documentation using statistical software

35 DMD work in SPSS, STATA, SAS and other statistical programs

35.1 Statistical programs in short

I recommend using a statistical program that has the possibility to use and save program files, eventually besides the possibility of using curtains to run commands (SPSS and STATA). The program will document the work you do with your data. The programming possibility have different names in different programs, it is called "syntax" in SPSS, program or pgm-files in SAS, and DO-files in STATA. Other statistical programs like R also have this possibility – while Microsoft Excel (that can also produce relevant statistical output) hasn't this feature. If you use Excel you won't have documentation of your work easily at hand.

In this section, I will show how to work with the first three mentioned.

35.2 Always use a program file

The reason for recommending a program-file is that you can produce documentation of the programming while working, and in the case of problems it is fairly easy to re-run the program-file and produce the same results again. Or, if having to change variables or values, re-running the program will quickly return the results based on these new values. Furthermore, if data is expanded (more records obtained), it is straightforward to re-run the program with the new dataset and update results.

It will also ease programming work, as those parts working and giving the anticipated outcome, can be copy-pasted to other parts of the analyses and be reused with small changes in variable names and procedures.

You will probably also learn smart ways of working with your programs, for instance using loops or arrays, and sections of this could also be relevant in other programming tasks and will easily be copy-pasted from your repository of smart-working codes.

Finally, program-files are smaller and more navigable than outcome-files.

35.2.1 Naming conventions for programs

You would probably end up needing more than one program file, eventually when you get feed-back from co-workers and decide to change parts of the tables or figures. If you change the command file after producing output, I would recommend a new version of the program file (save under another name) in order to keep documentation of the work. This new version should be named in accordance with your own system, for instance by adding a version number. In the file itself, you may keep a track of the versions by including a date and comments for the changes being made.

35.3 Document the data files being used

If the program (aka syntax- or DO-file) is to work according to the principles laid out in this book, it must start by documenting which data is read. You will often have more than one version of the dataset, and it should be clear without question which version has been used to produce the results.

The dataset might be the result of using ED and the E-file should then be exported (documented!) and placed in a relevant part of the computer. The same goes for data exported from REDCap. By including the path for the file, it is documented which version is used. It might work as easy as this, where the commands for the syntax-file in SPSS, the DO file in STATA and the program file i SAS will read the actual file:

SPSS:

```
Get file='k:\data\xxx'.
```

STATA:

```
use “k:\data\xxx’,clear
```

SAS:

```
libname dmd ‘k:\data’;
data xny;
set dmd.xxx;
run;
```

When datasets have been exported/imported they should always be controlled, for instance by comparing codebooks. See chapter 10 and 39.2. This comparison should also be documented, at least by writing a note/comment of it in your project diary or it could be stated in a comment in the program itself (see next subchapter 35.4).

The exported dataset would eventually need to be further cleaned, new variables with labels generated and errors corrected. The resulting final dataset might be saved under a new name in order to shorten the work with the following analyses – and in that case, the program-file should include a “save as”-command, again including the actual path for the file.

```
SPSS:
Save outfile='k:\data\xny' /compressed.
```

```
STATA:
save "k:\data\xny', replace
```

```
SAS:
libname dmd 'k:\data';
data dmd.xny;
set dmd.xxx; run;
```

When re-reading this file, either in the same or in a new program-file, the new file should again be included in the program-file with the new path and name to document which file is used.

35.4 Comment in the program-file

To make sure that you remember why the program-file was programmed the way it was, include comments to your program – both to help navigate ("the next part is doing this and that") and to help reading the actual program ("this part set a pointer x to zero, then find the records that satisfy the condition y and raise the pointer by one"). This will help both yourself and another reader to know what was intended – and to find errors in the program if it is not working as anticipated. Comments are ignored by the statistical program and will be recognized by a special set of characters in the beginning (and eventually also at the end) of the program line.

```
*in SPSS a star is used.
```

```
*in STATA a star can also be used
/*but a section of comments can also be marked (spanning more than one line) by marking both the start and the end of the line*/
// further this combination can also be used
```

```
*in SAS a star is also used to mark a comment that run until the semicolon;
/*can also be used in SAS, */
```

35.5 Divide the program-file in two – or use two distinct files

The dataset imported from ED or REDCap will not necessarily be ready before doing the statistical analyses, and new variables might also be needed at a later stage of the analyses (you "suddenly" discover that you need a bivariate outcome variable for logistic regression). To help both yourself and the reader, this new variable should be placed in the part used for data management, and not within the statistical part, even though it might feel awkward to work several pages back. But it will both help to keep the structure of the documentation, and force you to think about the statistical work. Furthermore, it will help you when later want to find out how the dichotomized variable was made - because it is with all the other new variables, their labels and value labels.

This might suggest having two program-files, one for data management and one for statistical analyses – or you might have the two parts of one program-file, clearly separated, the management part first and the analyses part last. When realizing a need for a new variable, you return to the management part and create and document it, and then use it in the statistical part. You will have to re-run the entire file before using the results.

36 Preparing the exported file

When the data has been exported from ED or REDCap, or a file received from someone else, and it have been read into the statistical program, it should be controlled. Is the number of records as anticipated, are all variables present (comparing code-books of the old and new file is suggested) – and again, document that this work has been done and what the results of the comparison was, even if everything is in order. This could easily be stated in comments in the program-file.

Then known issues should be handled. If using STATA, missing values are to be recorded as .a, .b, .c etc., and this should be documented. If using negative values in ED, you might consider recoding then in one sentence

```
STATA:
/*change missing values in all variables/*
recode _all -1=.a -2=.b
*or
mvdecode _all, mv(-1=.a -2=.b)
```

Secure that all variables are labelled, both for variables and for values (see below). If this is not in place, you should label variables. In SPSS this can be done manually in the data window, but that way documentation is not preserved and I would suggest always doing it using the program file.

If doing it manually in the STATA variable window, documentation is produced in the review list and can be documented in the do-file afterwards

SPSS: variable labels v15 ’xxx’. value labels v16e 1 ’no’ 2 ’yes’ 9 ’not stated’.
STATA: label variable v15 ’xxx’ /*value labels are defined and then attached to the variable*/ label define yn 1 ’no’ 2 ’yes’ 9 ’not stated’ label values v16e yn
SAS: /*labels in SAS can be used both temporary and permanently, the latter is suggested – see manual on my website*/

37 Correcting data

You might have identified problems during keying, that should be handled, for instance the person being 2,35 m in height, this should be documented as well (here just shown for SPSS and STATA). This will allow future readers to see what changes were made - and why.

In the example below I’ve used the id-number to identify the record (id=6) that is changed. In Statistics Denmark this is seen as identification, and you would not be allowed for instance to export your program-file. Instead, Statistics Denmark would recommend to recode directly using the wrong and right figure, but I would not suggest this, at changes may be unpredictable.

SPSS:

```
*a very high person was identified and height have been verified.
DO IF (id = 6).
RECODE q8 (999=235).
END IF.
EXECUTE.
```

STATA:

```
*a very high person was identified and height have been verified
replace q8 = 235 if id == 6
```

38 Transforming variables

Every time a new variable or a recoded variable is made, you should control that it turned out the way you anticipated. If the new variable is a direct result of an existing one (recoding five answer categories into two), a simple cross-tabulation of the old and the new variable will show that the values are placed in distinct groups (see chapter 8.1). If the new variable is the result of more than one recoding, the control should be based on all variables.

Remember also to think of handling of missing values, are they to result in missing value in the new variable, then they too should be controlled and this is not standard procedure in statistical programs, you will have to define that missing should be included.

If a recoding is missing, those values are stated as missing, and the cross-tab will have to include also missing values in order to make sure that all "old" values are in fact recoded.

39 Documenting data using labels

All new variables should be labelled and their values should also have value labels if measured on nominal or ordinal scales. This might best be programmed during data management to make sure that it is coded while present.

Statistical programs work differently here. I have shown principles in the table above. Besides this, STATA has a smart feature, allowing labels to include both text and the numbers in the same label.

Stata: numlabel _all, add

39.1 Changing labels to another language

You might work in your own language during DMD, writing the comments and labels in your native language, but when reports and manuscripts are prepared, they may be needed in another tongue, typically English. By using labels for variable and values, it is easy to change the output.

Include a new set of variable and value labels in a part of your program-file, and then run this part when you need to change the language. The new labels will be inserted when re-running analyses and will then be included in tables and figures.

STATA can use several set of labels, that can be switched. Use label language *languagename* to switch between set of labels, that you have been setting up.

Also in SAS, different label libraries can be called, and if two formats are defined, it is possible to change between them by telling SAS which format should be used.

39.2 Codebook

When variables have been produced, you should set up a codebook for the dataset - in SPSS and STATA by using the command codebook, in SAS by using single commands. The resulting list of variables, labels and description of data can be printed (*.pdf-format) and used for documentation.

40 Output results directly into tables

Researchers may copy the results from the statistical program in order to produce the tables and figures in accordance with the requirements of the journal. But this destroys the audit trail. Miscopied information, missing update etc. may produce tables, that are not actually giving the correct information.

Errors may be made when copying information from the statistical program into the tables for the paper or report, or if using another program to produce a specific figure.

Instead, the statistical programs should be used to either direct produce the tables for publication or to export the necessary results in a way, that other programs can import without having to write them manually.

40.1 Direct production of tables

STATA version 17 has been expanded with a very large set of commands to collect and use results from all types of statistical procedures for setting up tables, that can be fine-tuned both in relation to both what information the table is including and the use of e.g. table borders, marking of statistical significant results etc. The set of commands for this is the prefix collect:. The new table command is also very customizable.

For previous versions of STATA, general ado/program files have been written in order to help produce for instance the table 1 with description of the variables for the sample, eventually divided by the exposure. I have been using the ado extension "tabout" (type search tabout and install it), it has many possibilities, an easier ado-file is "tabone", that exports one-way tabulations, fewer possibilites, but easier to learn.

in SAS the automated output for table 1 is normally handled in macros, an example is the Drexel University BSC Table macro (https://zenodo.org/record/3698290#.X3Rd_Nov5PY)), but also many other macros have been produced.

It takes time to learn to program and use these possibilities, but the time is well invested in order to secure the last part of the audit trail and the possibility also to document the production of tables.

For other types of tables than "table 1", the statistical programs (except STATA version 17, that can also use collect: to produce tables of others procedures) will normally store for instance estimated regression coefficients and their standard errors, so you will be able to calculate e.g. confidence intervals from these without having to copy from the screen to obtain the same results as the programs present.

If working with STATA before version 17, "estout" is an ado-file, that can produce both unadjusted and adjusted models from stored estimates from regression models.

Again, this takes time to learn and it is tempting just to copy-paste from the screen when the deadline approaches, but being pressed on time actually raise the risk of making errors - and you may not notice.

Instead, learn to use a couple of ado-files/macros and you will in most instances be able to produce what you want. Material on the Internet is very helpful for this.

Furthermore, it will make it easy to update tables if data or procedures are changed - and you will not be tempted to leave results as they are just because you know the large amount of work that awaits you updating the manually edited tables.

40.2 Export results to new files

If this is not working, use the possibilities to export results in new data files from your statistical program. This could for instance be the mean and SD or estimated coefficients and their standard error, and the results could then be opened and calculated from, or the results could be exported in e.g. CSV-files and then be formatted in Excel or Word to obtain the results you need. By exporting the results, data are not changed in the process and you can still document the work flow.

You can also produce output in other file-formats by writing data directly into cells in e.g. Excel. In STATA this is conducted using the command putexcel.

While this takes a little longer than manually copying results the first times, it will end up being easier when the results are re-run, as you will then not have to re-copy results (with the possibility of having overseen some of the changes, resulting in faulty results.)

40.3 Producing figures

All the statistical programs mentioned here can produce graphs (figures, plots) and most graphs can be finetuned to obtain the resulting figure that you want.

In STATA and SAS the fine-tuning is included in the program-files, in SPSS the fine-tuning is made directly on the graph, and this is not documented. It will, however, not change the underlying data of the graph, so the audit-trail is kept. But the work done to tune the figure is not.

When the desired figure is produced, it can be saved or exported in a format that can be used for production. Most journals have a list of acceptable graphical formats, that can be used. If possible, use a vector-based format, as the figure can then be changed in size without problem. The exporting of the file should of course be documented.

You can also export small portions of your data and for instance use R to produce the figures you need. In R you can find "packages" that will produce virtual every type of graph, that you need.

40.4 Sending output to other programs

Although this is a little on the side of this text, be aware that both STATA and SAS can send results directly into other programs.

In STATA this is handled by having STATA writing into e.g. Word or PDF-files using commands like putdocx or putpdf.

In SAS using the ODS-system will allow SAS to similarly produce a variety of output.

41 Always re-run the final syntax-file

Even with the best of intentions, some of the work will not be included in the syntax file or might be executed in a different order – and this might affect the results.

It is therefore important to close all files including the datafile and then re-run the final syntax file from start to end, this way securing that the right dataset is read, the right variables transformed and labelled, analyses are made based on these data, and that tables and figures are the result of these. This will ensure that the results are actually reflecting the program-file.

42 Epilogue

In the words of Cartwright & Seale from the preface, all this work is painstaking and you need to be obsessional with DMD to maintain the audit trail, but it is the only way to secure that your data is as good as they get - and that the results are more right and more true than if we were cutting corners.

It pays off in the long run.