

# PS2010 Workshop Code Book

Luke Kendrick

2025-08-06

# Contents

<b>Preface</b>	<b>3</b>
0.1 About this Book . . . . .	3
0.2 <b>License</b> . . . . .	3
0.3 <b>Citation</b> . . . . .	3
<b>1 Workshop 1: Data Handling Skills</b>	<b>4</b>
1.1 Aims: . . . . .	4
1.2 Exercise 1: Import the Data . . . . .	4
1.3 Exercise 2: Inspect Your Data . . . . .	5
1.4 Exercise 3: Change a Variable Name . . . . .	5
1.5 Exercise 4: Remove a Variable . . . . .	5
1.6 Exercise 5: Filter Cases . . . . .	6
1.7 Exercise 6: Guess Who? . . . . .	6
1.8 Exercise 7: Create a New Variable . . . . .	7

# Preface

This is the PS2010 Psychological Research Methods and Analysis Workshop Codebook.

## 0.1 About this Book

This code book contains information, exercises, and code for the PS2010 workshop sessions.

This resource is a work in progress, and we're continually updating and improving it.

If you spot an error or something that doesn't look quite right, please get in touch:

luke.kendrick@rhul.ac.uk.

## 0.2 License

This work is licensed under Creative Commons Attribution-ShareAlike 4.0 International License (CC-BY-SA 4.0). You are free to share and adapt the material for non-commercial purposes, with appropriate credit and under the same license. If you adapt the material, you must distribute your contributions under the same license.

## 0.3 Citation

Kendrick, L. T. (2025). PS2010 Workshop Code Book (Version 1.0). <https://luke-kendrick.github.io/starttr/>

# Chapter 1

## Workshop 1: Data Handling Skills

### 1.1 Aims:

- Practice importing a .csv data file into RStudio using `read_csv()`
- Practice inspecting your data in RStudio.
- Use different data wrangling functions to develop your data handling skills.
- Check basic summary statistics.

### 1.2 Exercise 1: Import the Data

#### Import the Data File `guess_who.csv`

Before you begin, you will need the tidyverse package loaded.

```
install.packages("tidyverse") #install tidyverse if you do not have it.  
library(tidyverse) #loads tidyverse.
```

Next import the data file and store it as an object called `dataset`.

```
dataset <- read_csv("guess_who.csv")
```

If you see an error saying cannot find function `read_csv()` this usually means you have not loaded (or installed) the tidyverse package.

## 1.3 Exercise 2: Inspect Your Data

**Take a look at your newly imported data file**

Check the top right panel (the environment) and also use the code below to inspect your data set.

```
view(dataset) # this will open the data in a new tab.  
names(dataset) # this will show the variable names.
```

It is really important to look at the variable names as you'll be using them in code later on.

Answer Question 2.1 - 2.2 on your worksheet.

## 1.4 Exercise 3: Change a Variable Name

**One of the variable names is quite long. This can be annoying if we have to keep typing it out**

Change the variable name `do_you_own_a_pet` to `pet`. The `rename()` function will let you rename a variable.

```
dataset <- dataset %>%  
  rename(pet = do_you_own_a_pet)
```

Check it has worked:

```
names(mydata) # ask for the variable names again
```

## 1.5 Exercise 4: Remove a Variable

**We do not really care about the age variable for the next few exercises.**

Let's remove it.

The code below will create a new object (once we start removing things, it is best to keep the original data file called `dataset` in the environment)

```
mydata <- dataset %>%  
  select(-age)
```

This code will:

- Create a new object called `mydata`.
- Take our original data called `dataset`
- “And then” `%>%`
- Use the `select()` function to remove `age` by placing a minus symbol `-` in front of it.

From now on, we will use the object called `mydata` and not the original data set.

## 1.6 Exercise 5: Filter Cases

**We can select particular cases in our data set**

For example, I could ask: how many people were from the city of Birmingham using the code below:

```
mydata %>%
  filter(city == "birmingham") %>%
  count()
```

Check the console (bottom left panel) for the answer.

This code will:

- Take `mydata` and then...
- Filter it by the `city` variable.
- We use a double equals symbol `==` to specify an exact match.
- I’ve added “birmingham” in speech marks. Note it is lowercase as to match the data set and then...
- `count()` the number of data points.

Adapt the code above to answer question 5.1 on the worksheet.

## 1.7 Exercise 6: Guess Who?

**We can filter based on multiple criteria.**

The code below will show us someone who is from Brighton, has a dog, and does not drink coffee.

```
mydata %>%  
  filter(city == "brighton", pet = "dog", coffee = 0)
```

We can also use less than/more than symbols to filter data, For example, this will show all people who have a art enjoyment score of less than 20:

```
mydata %>%  
  filter(maths < 20)
```

Use what you have learned above and adapt your code to play GUESS WHO? and complete questions 6.1-6.2 on the worksheet.

## 1.8 Exercise 7: Create a New Variable

**Sometimes we might want to compute new scores or variables**

Add up the three enjoyment scores for `maths`, `science`, and `art` to create an overall score called `total_score`.

```
mydata %>%  
  mutate(total_score = maths + science + art)
```

This code will:

- Take `mydata` and then...
- Use the `mutate()` function to create a new variable named `total_score` which should equal `= maths + science + art`.

View the data set and look for the new column to see it has worked.

```
view(mydata)
```