

To add images, replace `tutheaderbl1.png` with the file name of any image you upload to your GitHub repository.

## Tutorial Aims

1. The first section
2. The second section
3. The third section

**You can read this text, then delete it and replace it with your text about your tutorial: what are the aims, what code do you need to achieve them?**

We are using `<a href="#section_number">text</a>` to create anchors within our text. For example, when you click on section one, the page will automatically go to where you have put `<a name="section_number"></a>`.

To create subheadings, you can use `#`, e.g. `# Subheading 1` creates a subheading with a large font size. The more hashtags you add, the smaller the text becomes. If you want to make text bold, you can surround it with `__text__`, which creates **text**. For italics, use only one understore around the text, e.g. `_text_`, *text*.

## Subheading 1

## Subheading 2

## Subheading 3

This is some introductory text for your tutorial. Explain the skills that will be learned and why they are important. Set the tutorial in context.

You can get all of the resources for this tutorial from this GitHub repository. Clone and download the repo as a zip file, then unzip it.

## 1. The first section

At the beginning of your tutorial you can ask people to open **RStudio**, create a new script by clicking on **File/ New File/ R Script** set the working directory and load some packages, for example **ggplot2** and **dplyr**. You can surround package names, functions, actions (“File/ New...”) and small chunks of code with backticks, which defines them as inline code blocks and makes them stand out among the text, e.g. **ggplot2**.

When you have a larger chunk of code, you can paste the whole code in the **Markdown** document and add three backticks on the line before the code chunks starts and on the line after the code chunks ends. After the three backticks that go before your code chunk starts, you can specify in which language the code is written, in our case **R**.

To find the backticks on your keyboard, look towards the top left corner on a Windows computer, perhaps just above **Tab** and before the number one key. On a Mac, look around the left **Shift** key. You can also just copy the backticks from below.

```
# Set the working directory
setwd("your_filepath")

# Load packages
library(ggplot2)
library(dplyr)
```

## 2. The second section

You can add more text and code, e.g.

```
# Create fake data
x_dat <- rnorm(n = 100, mean = 5, sd = 2) # x data
y_dat <- rnorm(n = 100, mean = 10, sd = 0.2) # y data
xy <- data.frame(x_dat, y_dat) # combine into data frame
```

Here you can add some more text if you wish.

```
xy_fil <- xy %>% # Create object with the contents of `xy`
  filter(x_dat < 7.5) # Keep rows where `x_dat` is less than 7.5
```

And finally, plot the data:

```
ggplot(data = xy_fil, aes(x = x_dat, y = y_dat)) + # Select the data to use
  geom_point() + # Draw scatter points
  geom_smooth(method = "loess") # Draw a loess curve
```

At this point it would be a good idea to include an image of what the plot is meant to look like so students can check they've done it right. Replace `IMAGE_NAME.png` with your own image file:

## 3. The third section

More text, code and images.

This is the end of the tutorial. Summarise what the student has learned, possibly even with a list of learning outcomes. In this tutorial we learned:

- how to generate fake bivariate data

- how to create a scatterplot in ggplot2

- some of the different plot methods in ggplot2 We can also provide some useful links, include a contact form and a way to send feedback.

For more on `ggplot2`, read the official `ggplot2` cheatsheet.

Everything below this is footer material - text and links that appears at the end of all of your tutorials.

Check out our [Useful links](#) page where you can find loads of guides and cheatsheets.

If you have any questions about completing this tutorial, please contact us on [ourcodingclub@gmail.com](mailto:ourcodingclub@gmail.com)

We would love to hear your feedback on the tutorial, whether you did it in the classroom or online! Follow our coding adventures on Twitter!

Subscribe to our mailing list:

```
<div class="block">
  <!-- subscribe form start -->
  <div class="form-group">
    <form action="https://getsimpleform.com/messages?form_api_token=de1ba2f2f947822946fb6e835437ec7"
    <div class="form-group">
      <input type='text' class="form-control" name='Email' placeholder="Email" required/>
    </div>
    <div>
      <button class="btn btn-default" type='submit'>Subscribe</button>
    </div>
  </form>
</div>
</div>
```