Data Visualization with ggplot2 in R

Activity #2: Customizing Themes

0.0 Getting Started

Load the appropriate packages. Recreate the base plot you generated in Activity #1. Store the plot as `plot`.

```
library(ggplot2)
library(palmerpenguins)
```

1.0 Using pre-packaged themes

Add a pre-packaged theme to your base plot:

- theme_grey()
- theme_bw()
- theme_linedraw()
- theme_light()
- theme_dark()
- theme_minimal()
- theme_classic()

2.0 Customizing your own theme

2.1 Add fonts

Add fonts that R can use for customizing plots.

```
#### For Windows Users ####
install.packages('extrafont')
library('extrafont')

font_import()
# You will be prompted to continue [y/n]
# Type 'y' and press enter

loadfonts(device="win")


#### For Mac Users ####
install.packages('extrafont')
library('extrafont')
```

```
font_import()
# You will be prompted to continue [y/n]
# Type 'y' and press enter

loadfonts()
```

Check what fonts are available with `fonts()`.

2.2 Customizing theme elements

Using the `theme()` function, customize the following theme elements:

- plot.title
- plot.subtitle
- panel.background
- panel.border
- panel.background
- panel.grid
- axis.title
- axis.text

You should use some combination of the following adjustments in your theme:

- color
- fill
- size
- linewidth
- family
- face (bold, italic, etc.)

Be creative. Effective data communication requires important information to be interpretable *and* visually inviting.

2.3 Saving your theme

Store your theme as `my_theme`.

2.4 Reusing your theme

Run the following code:

```
plot + my_theme
```

3.0 Examples

Flipper Length by Body Mass in Penguins

Species type in color