Homework 7

GONG Kuiyuan

Student ID: 39-246182

Student name: GONG Kuiyuan

Preferred name: Eddie

Answers:

Task 1:

```
library(pacman)
p_load(datasauRus)

library(pacman)
p_load(dplyr)
```

Task 2: We could use either `?` or `help()` to access the documentation.

```
?datasaurus_dozen
#help(datasaurus_dozen)
```

Task 3: Combining the information from the documentation, we get to know that `datasaurus_dozen` is a data frame that consists of 1846 rows and 3 columns. `dataset` indicates the sources of dataset, where `x` and `y` are values, respectively.

```
names(datasaurus_dozen)
```

```
[1] "dataset" "x"       "y"
```

```r
print(datasaurus_dozen)
```

```
# A tibble: 1,846 x 3
   dataset     x     y
   <chr>   <dbl> <dbl>
 1 dino     55.4  97.2
 2 dino     51.5  96.0
 3 dino     46.2  94.5
 4 dino     42.8  91.4
 5 dino     40.8  88.3
 6 dino     38.7  84.9
 7 dino     35.6  79.9
 8 dino     33.1  77.6
 9 dino     29.0  74.5
10 dino     26.2  71.4
# i 1,836 more rows
```

```r
str(datasaurus_dozen)
```

```
tibble [1,846 x 3] (S3: tbl_df/tbl/data.frame)
 $ dataset: chr [1:1846] "dino" "dino" "dino" "dino" ...
 $ x      : num [1:1846] 55.4 51.5 46.2 42.8 40.8 ...
 $ y      : num [1:1846] 97.2 96 94.5 91.4 88.3 ...
 - attr(*, "spec")=List of 2
  ..$ cols   :List of 3
  .. ..$ dataset: list()
  .. .. ..- attr(*, "class")= chr [1:2] "collector_character" "collector"
  .. ..$ x      : list()
  .. .. ..- attr(*, "class")= chr [1:2] "collector_double" "collector"
  .. ..$ y      : list()
  .. .. ..- attr(*, "class")= chr [1:2] "collector_double" "collector"
  ..$ default: list()
  .. ..- attr(*, "class")= chr [1:2] "collector_guess" "collector"
  ..- attr(*, "class")= chr "col_spec"
```

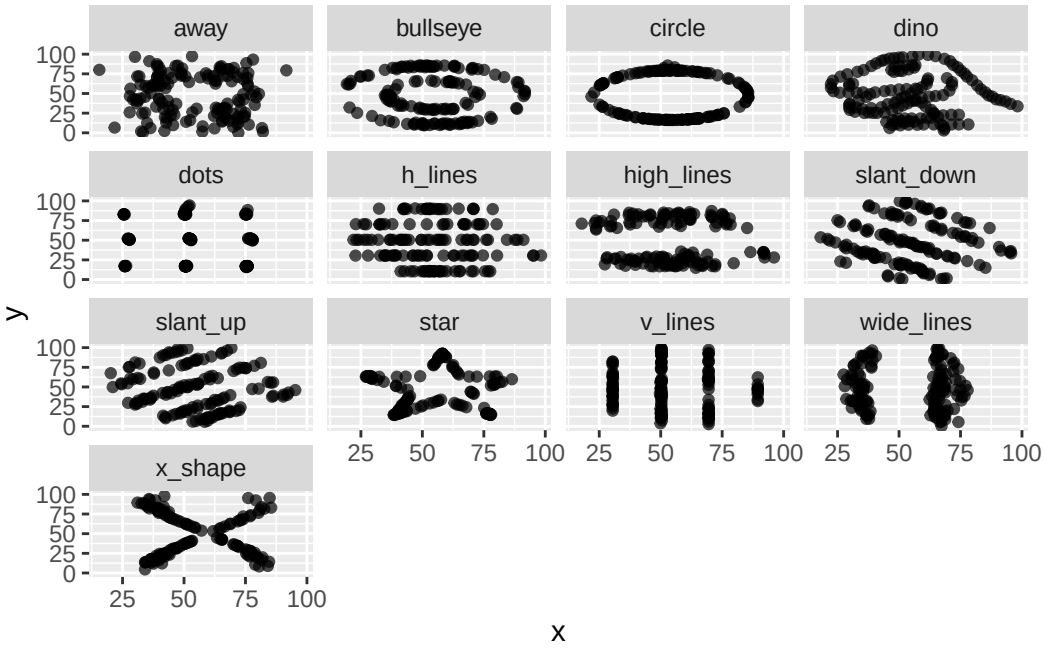
```r
#ncol(datasaurus_dozen)
#nrow(datasaurus_dozen)
datasaurus_dozen |>
  distinct(dataset)
```

```
# A tibble: 13 x 1
   dataset
   <chr>
 1 dino
 2 away
 3 h_lines
 4 v_lines
 5 x_shape
 6 star
 7 high_lines
 8 dots
 9 circle
10 bullseye
11 slant_up
12 slant_down
13 wide_lines
```

Task 4:

```
library(pacman)
p_load(ggplot2)

ggplot(
  data = datasaurus_dozen,
  aes(x = x, y = y)
  ) +
  geom_point(color = "black", alpha = 0.7
  ) +
  facet_wrap(~ dataset)
```



Task 5:

```
summary_stats <- datasaurus_dozen |>
  group_by(dataset) |>
  summarise(
    mean_x = mean(x),
    mean_y = mean(y),
    sd_x = sd(x),
    sd_y = sd(y),
    correlation = cor(x, y)
  )
print(summary_stats)
```

```
# A tibble: 13 x 6
   dataset    mean_x mean_y  sd_x  sd_y correlation
   <chr>       <dbl>  <dbl> <dbl> <dbl>       <dbl>
 1 away         54.3   47.8  16.8  26.9     -0.0641
 2 bullseye     54.3   47.8  16.8  26.9     -0.0686
 3 circle       54.3   47.8  16.8  26.9     -0.0683
 4 dino         54.3   47.8  16.8  26.9     -0.0645
 5 dots         54.3   47.8  16.8  26.9     -0.0603
 6 h_lines      54.3   47.8  16.8  26.9     -0.0617
 7 high_lines   54.3   47.8  16.8  26.9     -0.0685
```

```
 8 slant_down   54.3   47.8  16.8  26.9     -0.0690
 9 slant_up     54.3   47.8  16.8  26.9     -0.0686
10 star         54.3   47.8  16.8  26.9     -0.0630
11 v_lines      54.3   47.8  16.8  26.9     -0.0694
12 wide_lines   54.3   47.8  16.8  26.9     -0.0666
13 x_shape      54.3   47.8  16.8  26.9     -0.0656
```

Task 6: The numbers we got after these operations are very similar with each other. They basically share the same mean, standard deviation and correlation. However, those plots from `Task 4` indicate that there may not be any correlation between `x` and `y` in the first place since the shapes of plots could ba a dinosaur or a star.

Task 7: I also stored my plot in `p`.

```
library(pacman)
p_load(palmerpenguins)

names(penguins)
```

```
[1] "species"           "island"            "bill_length_mm"
[4] "bill_depth_mm"     "flipper_length_mm" "body_mass_g"
[7] "sex"               "year"
```

```
penguins |>
  distinct(species) |>
  print()
```

```
# A tibble: 3 x 1
  species
  <fct>
1 Adelie
2 Gentoo
3 Chinstrap
```

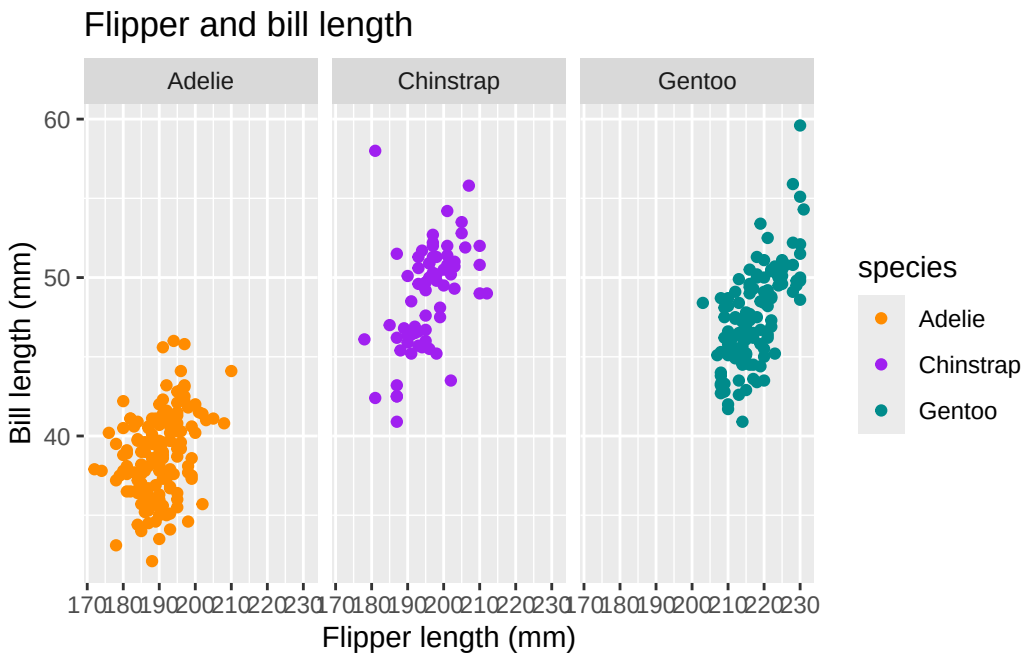
```
penguins_colors <- c("darkorange", "purple", "cyan4")

p <-
  ggplot(
  data = penguins,
  aes(x = flipper_length_mm,
  y = bill_length_mm,
```

```
  color = species)
) +
  geom_point() +
  labs(
    x = "Flipper length (mm)",
    y = "Bill length (mm)",
    title = "Flipper and bill length"
  ) +
  facet_wrap(~ species) +
  scale_color_manual(
  values = penguins_colors
  )

print(p)
```

```
Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_point()`).
```



Task 8: I choose the `theme_clean()` from `ggthemes` because I feel like it could make the plot much cleaner and easier for us to observe the data pattern.

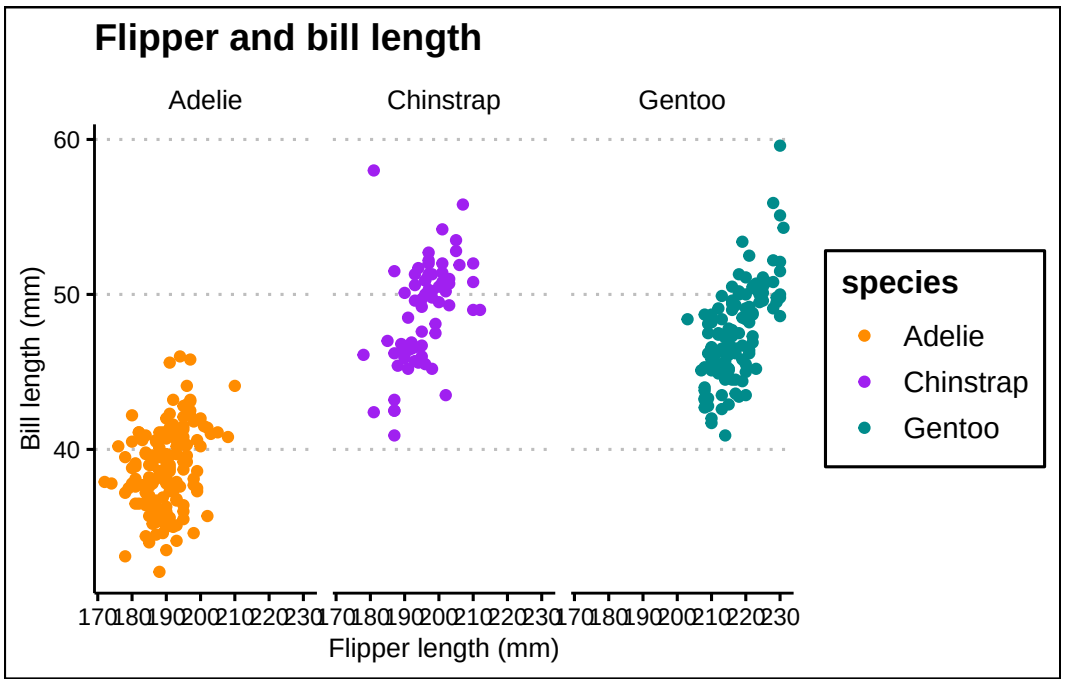
```
library(pacman)
p_load(ggthemes)

theme_set(theme_clean())
```

Task 9:

```
print(p)
```

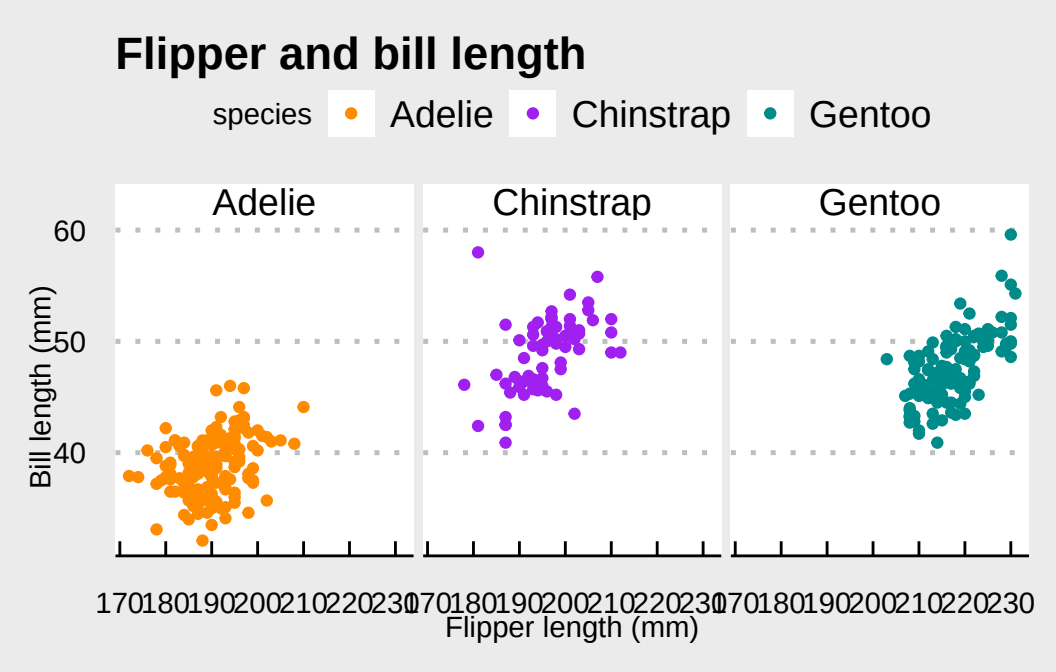
```
Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_point()`).
```



Task 10:

```
p <- p + theme_economist_white()
print(p)
```

```
Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_point()`).
```



Task 11:

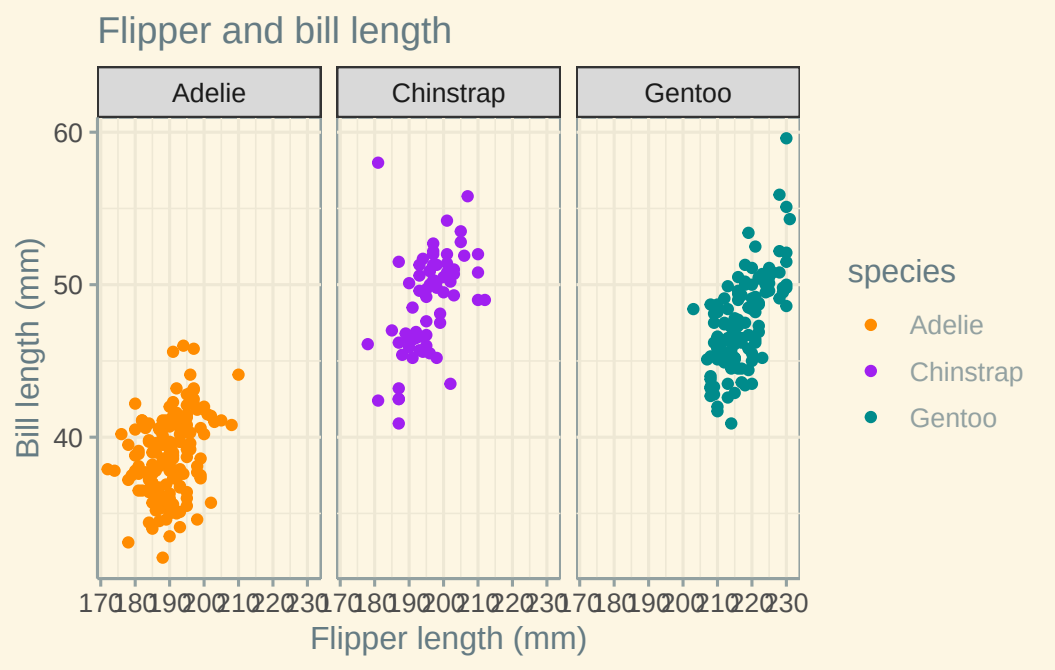
```
penguins_colors <- c(
  Chinstrap = "purple",
  Adelie = "darkorange",
  Gentoo = "cyan4"
)
```

Task 12: Although I changed the order of species together with its color, it doesn't affect the color assigned to each species unless I change the code into `Adelie = purple`. The color assigned to `Adelie` will be purple in this case.

```
p <-
  ggplot(
  data = penguins,
  aes(x = flipper_length_mm,
  y = bill_length_mm,
  color = species)
) +
  geom_point() +
  labs(
    x = "Flipper length (mm)",
    y = "Bill length (mm)",
```

```
    title = "Flipper and bill length"
  ) +
  facet_wrap(~ species) +
  scale_color_manual(
  values = penguins_colors
  ) +
  theme_solarized()
print(p)
```

```
Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_point()`).
```



Task 13: Now, we can see that the order of the plot is different, which starts with the `Gentoo` instead of the `Adelie`.

```
penguins$species <- factor(penguins$species, levels = c("Gentoo", "Adelie", "Chinstrap"))

p <-
  ggplot(
  data = penguins,
  aes(x = flipper_length_mm,
  y = bill_length_mm,
```

```
  color = species)
) +
  geom_point() +
  labs(
    x = "Flipper length (mm)",
    y = "Bill length (mm)",
    title = "Flipper and bill length"
  ) +
  facet_wrap(~ species) +
  scale_color_manual(
  values = penguins_colors
  )
print(p)
```

```
Warning: Removed 2 rows containing missing values or values outside the scale range
(`geom_point()`).
```