# R RIGHT ANSWERS

## 1. Air Quality Analysis: Inbuilt dataset: airquality in R

```
library(dplyr)
data("airquality")

# A. Filter records for July (Month = 7)
july_data <- airquality %>%
  filter(Month == 7)
print(july_data)

# B. Group by Month and calculate average Ozone
ozone_avg <- airquality %>%
  group_by(Month) %>%
  summarise(Avg_Ozone = mean(Ozone, na.rm = TRUE))  # Corrected: na.rm must be inside mean()
print(ozone_avg)

# C. Use pipe to fetch records with Ozone > 50
high_ozone <- airquality %>%
  filter(Ozone > 50)
print(high_ozone)
```

## 3. Car Performance Analysis: Inbuilt dataset: mtcars in R

```
library(dplyr)
library(ggplot2)

# Add readable transmission labels
mtcars$Transmission <- ifelse(mtcars$am == 0, "Automatic", "Manual")

# A. Compare fuel efficiency (mpg) of automatic vs manual cars
avg_mpg <- mtcars %>%
  group_by(Transmission) %>%
  summarise(Average_MPG = mean(mpg))
print(avg_mpg)

# Bar plot
ggplot(avg_mpg, aes(x = Transmission, y = Average_MPG, fill = Transmission)) +
  geom_bar(stat = "identity") +
  labs(title = "Fuel Efficiency by Transmission Type",
       x = "Transmission Type",
       y = "Average MPG") +
  theme_minimal()
```

## 5. Dataset: PlantGrowth

```
library(dplyr)
library(ggplot2)
data("PlantGrowth")

# A. Compute average weight
avg_weight <- PlantGrowth %>%
  group_by(group) %>%
  summarise(Avg_Weight = mean(weight))
print(avg_weight)
```

**# B. Bar chart**

```
ggplot(avg_weight, aes(x = group, y = Avg_Weight, fill = group)) +
  geom_bar(stat = "identity") +
  labs(title = "Average Plant Weight by Group",
       x = "Treatment Group",
       y = "Average Weight") +
  theme_minimal()
```

## 7. Iris Flower Classification: Dataset: iris

```
library(dplyr)
library(ggplot2)
data("iris")
```

**# A. Petal stats**

```
avg_petal <- iris %>%
  group_by(Species) %>%
  summarise(
    Avg_Petal_Length = mean(Petal.Length),
    Avg_Petal_Width = mean(Petal.Width)
  )
print(avg_petal)
```

**# B. Sepal scatter plot**

```
ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, color = Species)) +
  geom_point(size = 3) +
  labs(title = "Sepal Dimensions by Species",
       x = "Sepal Length",
       y = "Sepal Width") +
  theme_minimal()
```

## 9. Distribution of Petal Length: Dataset: iris

```
library(ggplot2)
data("iris")
```

## # Histogram

```
ggplot(iris, aes(x = Petal.Length)) +
  geom_histogram(binwidth = 0.5, fill = "skyblue", color = "black") +
  labs(title = "Histogram of Petal Length",
       x = "Petal Length",
       y = "Frequency") +
  theme_minimal()
```

**# Density plot**

```
ggplot(iris, aes(x = Petal.Length)) +
  geom_density(fill = "lightgreen", alpha = 0.6) +
  labs(title = "Density Plot of Petal Length",
       x = "Petal Length",
       y = "Density") +
  theme_minimal()
```

## 11. Dataset: mtcars

```
library(ggplot2)
library(dplyr)
data("mtcars")
```

```
# A. Filter high horsepower
high_hp_cars <- mtcars %>%
  filter(hp > 150)
print(high_hp_cars)

# B. Scatter plot
ggplot(mtcars, aes(x = hp, y = mpg)) +
  geom_point(color = "steelblue", size = 3) +
  labs(title = "Horsepower vs. Fuel Efficiency",
       x = "Horsepower (hp)",
       y = "Miles per Gallon (mpg)") +
  theme_minimal()
```

## 13. CO2 Emissions: Dataset: CO2

```
library(dplyr)
library(ggplot2)
data("CO2")

# A. Average CO2 uptake by treatment
avg_uptake <- CO2 %>%
  group_by(Treatment) %>%
  summarise(Avg_Uptake = mean(uptake))
print(avg_uptake)

# B. Scatter: uptake vs concentration by plant type
ggplot(CO2, aes(x = conc, y = uptake, color = Type)) +
  geom_point(size = 3) +
  labs(title = "CO2 Uptake by Concentration and Plant Type",
       x = "CO2 Concentration (ppm)",
       y = "CO2 Uptake",
       color = "Plant Type") +
  theme_minimal()
```

## 15. Supermarket Sales Data Cleaning

```
sales_data <- data.frame(
  Transaction_ID = c(101, 102, 103, 104),
  Date = as.Date(c("2024-03-01", "2024-03-02", "2024-03-03", "2024-03-04")),
  Product = c("Apples", "Bread", "Milk", "Cheese"),
  Category = c("Fruits", "Bakery", "Dairy", "Dairy"),
  Quantity = c(2, NA, -1, 1),
  Price = c(1.5, 2.0, 3.0, 5.0),
  Total_Sales = c(3.0, NA, -3.0, 5.0)
)

# 1. Handle missing values

# Replace missing Quantity with the median
sales_data$Quantity[is.na(sales_data$Quantity)] <- median(sales_data$Quantity, na.rm = TRUE)

# 2. Correct negative Quantity values
sales_data$Quantity[sales_data$Quantity < 0] <- abs(sales_data$Quantity[sales_data$Quantity < 0])

# 3. Recompute Total_Sales where it's 0 or wrong
sales_data$Total_Sales <- sales_data$Quantity * sales_data$Price  # Always recompute
```

```
# 4. Summarize total sales per category
library(dplyr)
category_summary <- sales_data %>%
  group_by(Category) %>%
  summarise(Total_Sales_Sum = sum(Total_Sales))
print(category_summary)
```

## Golden Question: General Manipulation and Visualization

```
library(dplyr)
library(ggplot2)
data("mtcars")
```

### # Manipulation

```
manipulated_data <- mtcars %>%
  select(mpg, cyl, hp, gear) %>%
  filter(hp > 100) %>%
  mutate(Efficiency = mpg / cyl) %>%
  group_by(gear) %>%
  summarise(
    Avg_MPG = mean(mpg),
    Avg_HP = mean(hp),
    Count = n()
  ) %>%
  arrange(desc(Avg_MPG))
print(manipulated_data)
```

```
# Plot 1: Scatter HP vs MPG
ggplot(mtcars, aes(x = hp, y = mpg)) +
  geom_point(size = 3, color = "darkred") +
  labs(title = "Horsepower vs MPG",
       x = "Horsepower (hp)",
       y = "Miles Per Gallon (mpg)") +
  theme_minimal()
```

```
# Plot 2: Boxplot of MPG by Gear
ggplot(mtcars, aes(x = factor(gear), y = mpg)) +
  geom_boxplot(fill = "orange") +
  labs(title = "Distribution of MPG by Number of Gears",
       x = "Number of Gears",
       y = "Miles Per Gallon (mpg)") +
  theme_minimal()
```