Report: WHO Tuberculosis Data

Part 1: Tidying Up the Data

1. Setting Up the Environment

- Loading Libraries: The `tidyverse` library is loaded for data manipulation and visualization.

```
library(tidyverse)
```

2. Loading the Dataset

- Data Inspection: The WHO dataset is loaded into the R environment. This dataset contains information about tuberculosis cases across different countries, age groups, and genders.

```
who
```

3. Data Transformation Steps (Section 12.6)

- Step 1: Transforming the dataset using `pivot_longer.` This step reshapes the data from wide to long format, selecting columns related to tuberculosis cases and renaming them.

```
who1 <- who %>% pivot_longer(cols = new_sp_m014:newrel_f65, names_to = "key",
values_to = "cases", values_drop_na = TRUE)
```

Output of the transformed data (`who1`) is shown.

- Step 2: Updating the 'key' column to replace 'newrel' with 'new_rel' using string manipulation functions.

```
who2 <- who1 %>% mutate(key = stringr::str_replace(key, "newrel", "new_rel")
```

Preview of `who2` shows the updated 'key' values.

-Step 3: Separating the 'key' column into multiple columns ('new', 'type', 'sexage') using `separate`.

```
who3 <- who2 %>% separate(key, c("new", "type", "sexage"), sep = "_")
```

Viewing the count of 'new' column in `who3` for verification.

- Step 4: Dropping unnecessary columns ('new', 'iso2', 'iso3') to simplify the dataset

```
who4 <- who3 %>% select(-new, -iso2, -iso3)
```

- Step 5: Further separating the 'sexage' column into 'sex' and 'age' for clearer categorization.

```
who5 <- who4 %>% separate(sexage, c("sex", "age"), sep = 1)
```

The tidied data (`who5`) is now ready for analysis and visualization.

4. Saving the Tidied Data to an Excel File

- Preparing for File Export: Installing and loading the `openxlsx` package for Excel file operations.

```
   if (!requireNamespace("openxlsx", quietly = TRUE)) {

  install.packages("openxlsx")

 }

 library(openxlsx)
```

- Exporting Data: Writing the modified data frame (`who5`) to an Excel file saved in the main directory.

```
   excel_file_path <- "./who5.xlsx"

 write.xlsx(who5, excel_file_path, rowNames = FALSE)
```

The data is now saved in an Excel file named 'who5.xlsx' for further use.

Part 2: Exploratory Data Analysis (EDA)

1. Loading the 'nycflights' Dataset

- Installation and Loading: The `nycflights13` package, which contains data about flights departing from New York City in 2013, is installed and loaded.

```
    install.packages("nycflights13")

  library(nycflights13)
```

- Dataset Inspection: The `flights` dataset from `nycflights13` is loaded for analysis.

```
    flights <- nycflights13::flights
```

2. Data Analysis and Questions

- QUESTION 1: Busiest New York Airport

- Task: Determine which of the three New York airports is the busiest based on the number of departures.

- Method: Count the number of departures for each airport and identify the one with the highest count.

- Result:

```
   departure_counts <- table(flights$origin)

   busiest_airport <- names(departure_counts)[which.max(departure_counts)]
```

```
cat("The busiest airport is:", busiest_airport, "\n")
```

- ***QUESTION 3: Flights from LGA on Thanksgiving Day, 2013***

- Task: Count how many flights departed from LaGuardia Airport (LGA) on Thanksgiving Day in 2013.

- Method: Filter for flights departing from LGA on the specific date and count them.

- Result:

```
    thanksgiving_flights <- subset(flights, origin == "LGA" & month == 11 & day == 28)

num_thanksgiving_flights <- nrow(thanksgiving_flights)

cat("The number of flights departed from LGA on Thanksgiving Day, 2013:", num_thanksgiving_flights, "\n")
```

- ***QUESTION 8: Carrier with Fewest Flights***

- Task:Identify the carrier with the smallest number of flights.

- Method: Count flights for each carrier and find the one with the minimum count.

- Result:

```
carrier_counts <- table(flights$carrier)

smallest_carrier <- names(which.min(carrier_counts))

cat("The carrier with the smallest number of flights is:", smallest_carrier, "\n")
```

- ***QUESTION 9: Shortest Average Delay at JFK***

  - **Task: Find out which airline had the shortest average delay per flight departing from John F. Kennedy Airport (JFK).**

  - **Method: Calculate the average delay for each airline departing from JFK and identify the one with the shortest delay.**

  - **Result:**

```
    jfk_flights <- subset(flights, origin == "JFK")

average_delay_per_carrier <- tapply(jfk_flights$dep_delay, jfk_flights$carrier, mean, na.rm = TRUE)

shortest_delay_carrier <- names(which.min(average_delay_per_carrier))

cat("The airline with the shortest average delay per flight departing from JFK is:", shortest_delay_carrier, "\n")
```

- ***QUESTION 10: Longest Average Delay at LGA***

  - **Task: Determine which airline had the longest average delay per flight departing from LaGuardia Airport (LGA).**

  - **Method: Calculate the average delay for each airline departing from LGA and find the one with the longest delay.**

  - **Result:**

```
    lga_flights <- subset(flights, origin == "LGA")

average_delay_per_carrier <- tapply(lga_flights$dep_delay, lga_flights$carrier, mean, na.rm = TRUE)

longest_delay_carrier <- names(which.max(average_delay_per_carrier))

cat("The airline with the longest average delay per flight departing from LGA is:", longest_delay_carrier, "\n")
```

Part 3: Exploratory Data Analysis - Visualization

1. Plotting Average Departure Delay from JFK by Month

- Objective: To visualize the average departure delay for flights departing from John F. Kennedy Airport (JFK) for each month.

- Approach: Filter the flights originating from JFK, calculate the average delay per month, and plot these averages using a bar chart.

```
    library(ggplot2)

  jfk_flights <- subset(flights, origin == "JFK")

  average_delay_per_month <- aggregate(dep_delay ~ month, data = jfk_flights, FUN = mean, na.rm = TRUE)

  ggplot(average_delay_per_month, aes(x = month, y = dep_delay)) +

   geom_bar(stat = "identity", fill = "skyblue", color = "black") +

   labs(title = "Average Departure Delay for Flights Departing from JFK per Month",

      x = "Month",

      y = "Average Departure Delay (minutes)") +

   theme_minimal()
```

- Visualization: The bar plot displays the monthly average departure delay from JFK, offering insights into how delay trends vary over the year.

2. Boxplot of Departure Delays for the 5 Busiest Carriers

- Objective: To examine the statistical distribution of departure delays across the five busiest carriers.

- Approach: Identify the top five carriers based on the number of flights, filter the data for these carriers, and use a boxplot to visualize the distribution of departure delays.

```
top_carriers <- names(sort(table(flights$carrier), decreasing = TRUE))[1:5]

top_carrier_flights <- filter(flights, carrier %in% top_carriers)

ggplot(top_carrier_flights, aes(x = carrier, y = dep_delay)) +

  geom_boxplot(fill = "skyblue", color = "black") +

  labs(title = "Statistical Distribution of Departure Delays for the 5 Busiest Carriers",

       x = "Carrier",

       y = "Departure Delay (minutes)") +

  theme_minimal()
```

- Visualization: The boxplot provides a detailed view of the spread and central tendency of departure delays for each of the top five carriers, highlighting outliers and variability in delays.

Part 4: Exploratory Data Analysis - Advanced Questions

1. Analysis of Tuberculosis Cases in 2012

- QUESTION 1: Countries with the Largest and Smallest Number of TB Cases in 2012

- Task: Identify the countries with the largest and smallest number of TB cases in 2012.

- Method: Filter the data for the year 2012 and identify the countries with the maximum and minimum TB cases.

```
    tb_2012 <- subset(who5, year == 2012)

largest_country <- tb_2012$country[which.max(tb_2012$cases)]

smallest_country <- tb_2012$country[which.min(tb_2012$cases)]

cat("Country with the largest number of TB cases in 2012:", largest_country, "\n")

cat("Country with the smallest number of TB cases in 2012:", smallest_country, "\n")
```

2. Tuberculosis Cases in Australia Over Time

- ***QUESTION 2: TB Cases per Gender in Australia Over Time***

- Task: Plot the number of TB cases per gender in Australia over the dataset's covered period.

- Method: Filter data for Australia, then create a bar plot showing TB cases per gender across different years.

```
   australia_data <- who5 %>% filter(country == "Australia")

ggplot(australia_data, aes(x = year, y = cases, fill = sex)) +
  geom_bar(stat = "identity", position = "stack") +

  labs(title = "TB Cases per Gender in Australia Over Time",

       x = "Year",

       y = "Number of TB Cases",

       fill = "Gender") +

  theme_minimal()
```

3. Tuberculosis Cases in Afghanistan (2000-2013)

- QUESTION 3: Total TB Cases per Gender in Afghanistan (2000-2013)

- Task: Visualize the total number of TB cases per gender in Afghanistan between 2000 and 2013.

- Method: Filter the dataset for Afghanistan and the specified period, separate the 'sex' column into 'gender' and 'age_group', and plot the data.

```
   afghanistan_data <- who5 %>%

  filter(country == "Afghanistan" & year >= 2000 & year <= 2013) %>%

  separate(sex, into = c("gender", "age_group"), sep = 1)

ggplot(afghanistan_data, aes(x = year, y = cases, fill = gender)) +

  geom_bar(stat = "identity", position = "stack") +

  labs(title = "Total TB Cases per Gender in Afghanistan (2000-2013)",

       x = "Year",

       y = "Total Number of TB Cases",

       fill = "Gender") +
    theme_minimal()
```

4. TB Cases by Age Group in North American Countries

- QUESTION 4: Percentage of TB Cases per Age Group in North America

- Task: Compute and plot the percentage of TB cases per age group for the USA, Canada, and Mexico over the longest period covered by the dataset.

- Method: Aggregate and calculate percentages of TB cases by age group for each country, and visualize with a pie chart.

```
      north_america_data <- who5[who5$country %in% c("USA", "Canada", "Mexico"), ]

   age_group_cases <- aggregate(cases ~ age + country, data = north_america_data, sum)

   total_cases_per_country <- aggregate(cases ~ country, data = north_america_data, sum)

   age_group_cases <- merge(age_group_cases, total_cases_per_country, by = "country")

   age_group_cases$percentage <- (age_group_cases$cases.x / age_group_cases$cases.y) * 100

   ggplot(age_group_cases, aes(x = "", y = percentage, fill = age)) +

    geom_bar(stat = "identity", width = 1) +

    coord_polar("y", start = 0) +

    facet_wrap(~ country) +

    theme_void() +

    labs(fill = "Age Group", title = "Percentage of TB Cases by Age Group in North America")
```

5. Reduction in TB Cases (1997-2010)

- QUESTION 10: Country with the Sharpest Reduction in TB Cases (1997-2010)

- Task: Determine which country had the sharpest reduction in the total number of TB cases between 1997 and 2010 in percentage terms.

- Method: Filter the data for the specified years, calculate the total number of cases per country for each year, then compute the percentage change for each country. Identify the country with the largest negative percentage change.

```
library(dplyr)
library(tidyr)
filtered_data <- who5 %>% filter(year %in% c(1997, 2010))
total_cases_by_country <- filtered_data %>%
  group_by(country, year) %>%
  summarize(total_cases = sum(cases, na.rm = TRUE))
cases_by_year_country <- total_cases_by_country %>%
  pivot_wider(names_from = year, values_from = total_cases)
cases_by_year_country <- cases_by_year_country %>%
  mutate(percentage_change = ((`2010` - `1997`) / `1997`) * 100)
sharpest_reduction_country <- cases_by_year_country %>%
  arrange(percentage_change) %>%
  slice(1)
print(sharpest_reduction_country)
```

- Interpretation: The output identifies the country that experienced the most significant reduction in TB cases between 1997 and 2010, providing valuable insights into public health progress and the effectiveness of TB control measures in that region.