SHETH LUJ AND SIR MV COLLEGE
Subject: Data Analysis with SAS / SPSS /R

Practical No: 10

Aim: Creating new variables using transformations and calculations in R. import dataset.

Code:

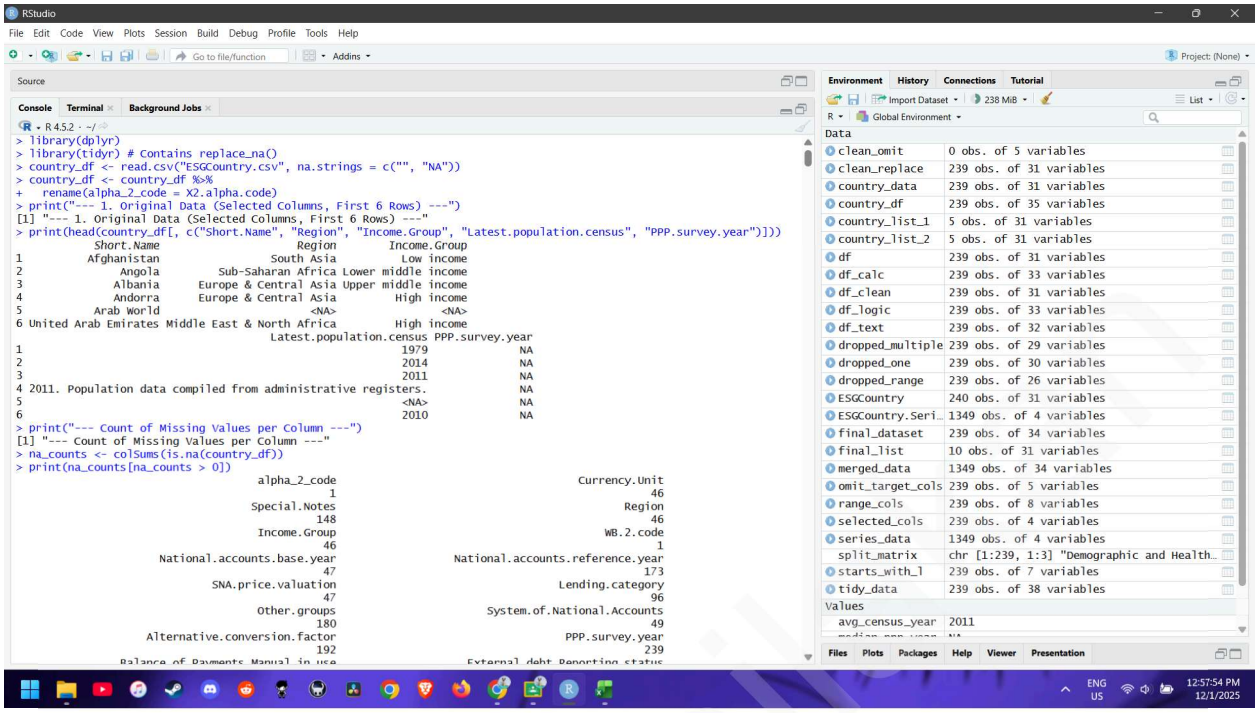
```
# R Script: Creating New Variables (Transformations & Calculations)

library(dplyr)
library(tidyr)

# 1. SETUP:  Import the Dataset and Clean for Calculation

df <- read.csv("ESGCountry.csv", na.strings = c("", "NA"))

# 1. Convert year columns to numeric (handling non-numeric text which becomes NA).
# 2. Fill NAs with the median for the year columns and a placeholder for text.
df_clean <- df %>%
  mutate(
    Latest.population.census = as.numeric(as.character(Latest.population.census)),
    PPP.survey.year = as.numeric(as.character(PPP.survey.year))
  ) %>%
  mutate(
    Latest.population.census = replace_na(Latest.population.census,
median(Latest.population.census, na.rm = TRUE)),
    PPP.survey.year = replace_na(PPP.survey.year, median(PPP.survey.year, na.rm = TRUE)),
    Income.Group = replace_na(Income.Group, "Not Reported")
  )

print("--- Cleaned Baseline Data ---")
print(head(df_clean %>% select(Short.Name, Income.Group, Latest.population.census,
PPP.survey.year)))

# 2. METHOD A: ARITHMETIC CALCULATIONS

df_calc <- df_clean %>%
  mutate(
    Census_Data_Age = 2025 - Latest.population.census,
    Survey_Year_Gap = Latest.population.census - PPP.survey.year
  )

print("--- Method A: Arithmetic Results (Data Age & Gap) ---")
print(df_calc %>% select(Short.Name, Latest.population.census, PPP.survey.year,
Census_Data_Age, Survey_Year_Gap) %>% head())
```

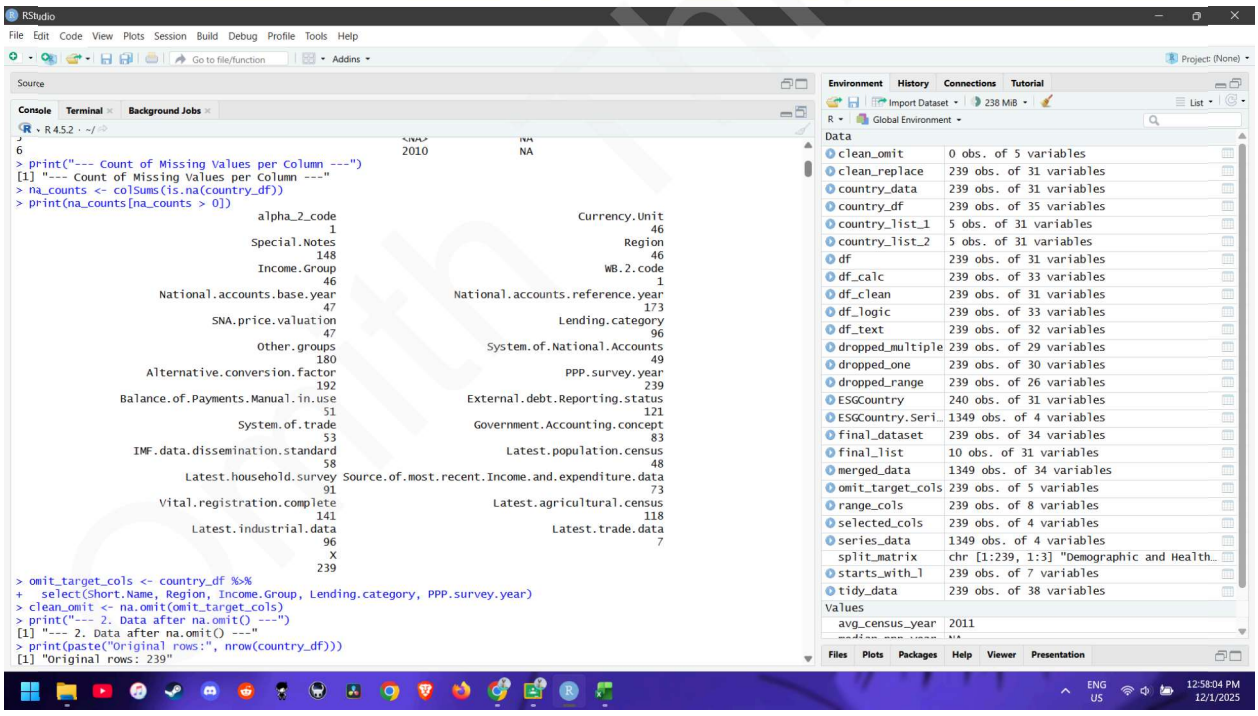
Omith Thilakan
S097

```
# 3. METHOD B: CONDITIONAL LOGIC (ifelse)

df_logic <- df_clean %>%
  mutate(
    Data_Age_Flag = ifelse(PPP.survey.year < 2000, "Outdated PPP Data", "Recent PPP Data"),
    Lending_Risk_Level = ifelse(Income.Group == "Low income", "High Risk", "Moderate/Low
Risk")
  )

print("--- Method B: Logic Results (Labels) ---")
print(df_logic %>% select(Short.Name, PPP.survey.year, Data_Age_Flag, Income.Group,
Lending_Risk_Level) %>% head())

# 4. METHOD C: TEXT TRANSFORMATION (paste)

df_text <- df_clean %>%
  mutate(
    Lending.category = replace_na(Lending.category, "Unspecified"),
    Region_Summary = paste("Country in the", Region, "region with", Lending.category,
"lending.")
  )

print("--- Method C: Text Transformation ---")
print(head(df_text$Region_Summary))

# 5. ALL TOGETHER (The Standard Workflow)

final_dataset <- df_clean %>%
  mutate(
    # Calc 1: Gap calculation
    Survey_Gap = Latest.population.census - PPP.survey.year,
    # Calc 2: Is the survey gap large?
    Has_Large_Gap = ifelse(Survey_Gap > 10, TRUE, FALSE),
    # Calc 3: Final combined status report
    Status_Report = paste0("Region: ", Region, " | Income: ", Income.Group, " | Gap: ",
Survey_Gap)
  )

print("--- Final Combined Dataset ---")
print(head(final_dataset %>% select(Short.Name, Income.Group, Survey_Gap,
Has_Large_Gap, Status_Report)))
```

SHETH LUJ AND SIR MV COLLEGE
Subject: Data Analysis with SAS / SPSS /R

Output:

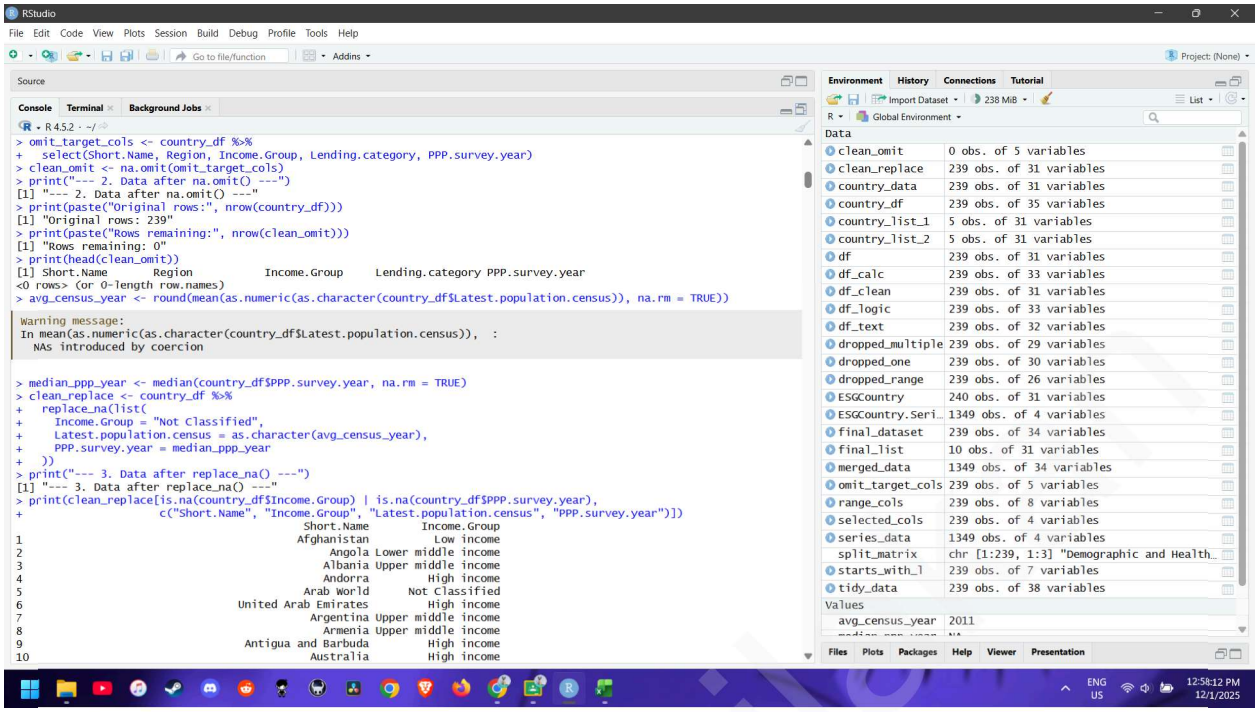




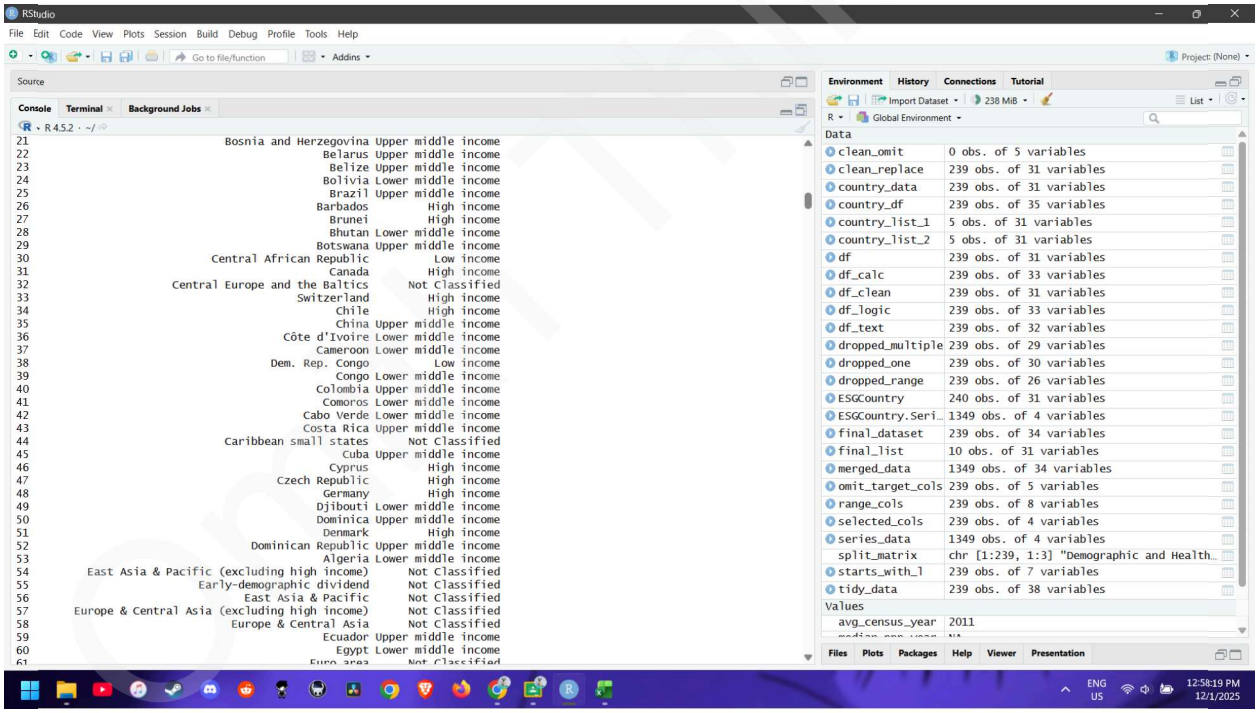
Omith Thilakan
S097

SHETH LUJ AND SIR MV COLLEGE

Subject: Data Analysis with SAS / SPSS /R



```
> omit_target_cols <- country_df %>%
+   select(Short.Name, Region, Income.Group, Lending.category, PPP.survey.year)
> clean_omit <- na.omit(omit_target_cols)
> print("--- 2. Data after na.omit() ---")
[1] "--- 2. Data after na.omit() ---"
> print(paste("Original rows:", nrow(country_df)))
[1] "Original rows: 239"
> print(paste("Rows remaining:", nrow(clean_omit)))
[1] "Rows remaining: 0"
> print(head(clean_omit))
[1] Short.Name       Region           Income.Group     Lending.category PPP.survey.year
<0 rows> (or 0-length row.names)
> avg_census_year <- round(mean(as.numeric(as.character(country_df$Latest.population.census)), na.rm = TRUE))
Warning message:
In mean(as.numeric(as.character(country_df$Latest.population.census)),  :
  NAs introduced by coercion

> median_ppp_year <- median(country_df$PPP.survey.year, na.rm = TRUE)
> clean_replace <- country_df %>%
+   replace_na(list(
+     Income.Group = "Not Classified",
+     Latest.population.census = as.character(avg_census_year),
+     PPP.survey.year = median_ppp_year
+   ))
> print("--- 3. Data after replace_na() ---")
[1] "--- 3. Data after replace_na() ---"
> print(clean_replace[is.na(country_df$Income.Group) | is.na(country_df$PPP.survey.year),
+                     c("Short.Name", "Income.Group", "Latest.population.census", "PPP.survey.year")])
                                    Short.Name        Income.Group
1                                 Afghanistan         Low income
2                                      Angola Lower middle income
3                                     Albania Upper middle income
4                                     Andorra         High income
5                                  Arab World      Not Classified
6                        United Arab Emirates         High income
7                                   Argentina Upper middle income
8                                     Armenia Upper middle income
9                           Antigua and Barbuda         High income
10                                  Australia         High income
```

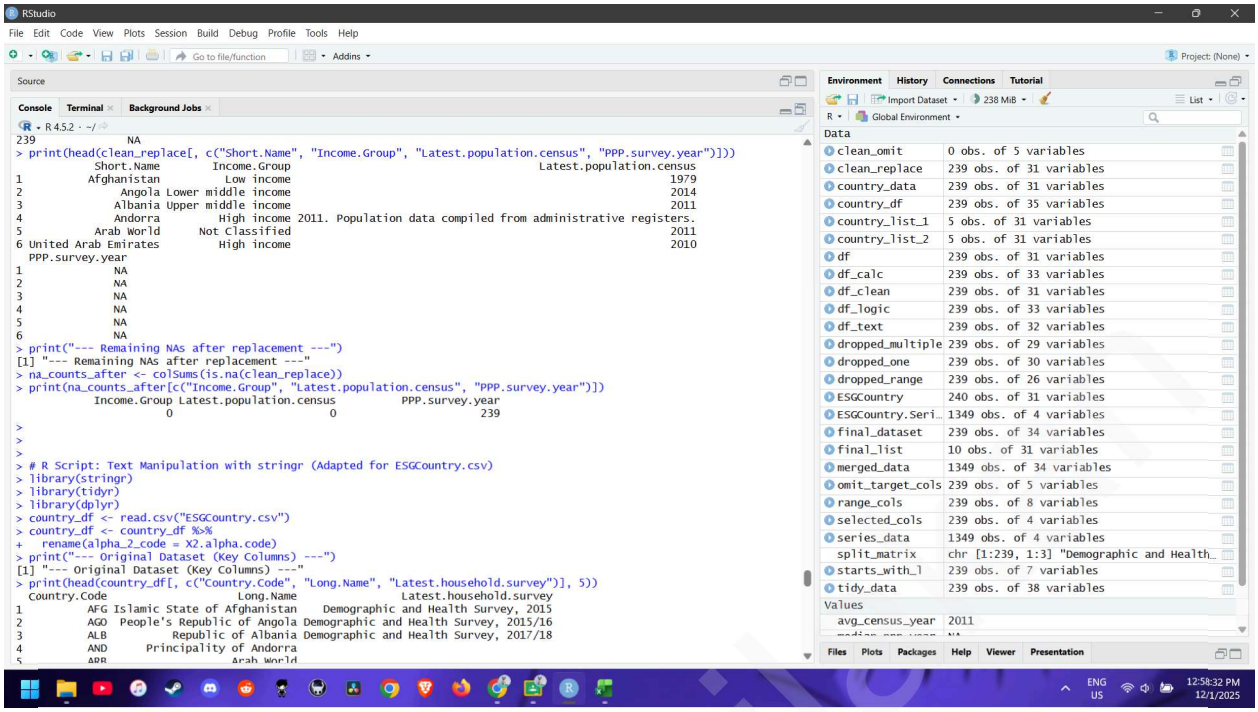


```
21                          Bosnia and Herzegovina Upper middle income
22                                        Belarus Upper middle income
23                                         Belize Upper middle income
24                                        Bolivia Lower middle income
25                                         Brazil Upper middle income
26                                       Barbados         High income
27                                         Brunei         High income
28                                         Bhutan Lower middle income
29                                       Botswana Upper middle income
30                       Central African Republic          Low income
31                                         Canada         High income
32           Central Europe and the Baltics      Not Classified
33                                    Switzerland         High income
34                                          Chile         High income
35                                          China Upper middle income
36                                  Côte d'Ivoire Lower middle income
37                                       Cameroon Lower middle income
38                                Dem. Rep. Congo          Low income
39                                          Congo Lower middle income
40                                       Colombia Upper middle income
41                                        Comoros Lower middle income
42                                     Cabo Verde Lower middle income
43                                     Costa Rica Upper middle income
44                       Caribbean small states      Not Classified
45                                           Cuba Upper middle income
46                                         Cyprus         High income
47                                 Czech Republic         High income
48                                        Germany         High income
49                                       Djibouti Lower middle income
50                                       Dominica Upper middle income
51                                        Denmark         High income
52                             Dominican Republic Upper middle income
53                                        Algeria Lower middle income
54 East Asia & Pacific (excluding high income)      Not Classified
55                    Early-demographic dividend      Not Classified
56                            East Asia & Pacific      Not Classified
57 Europe & Central Asia (excluding high income)      Not Classified
58                          Europe & Central Asia      Not Classified
59                                       Ecuador Upper middle income
60                                         Egypt Lower middle income
61                                      Euro area      Not Classified
```

Omith Thilakan

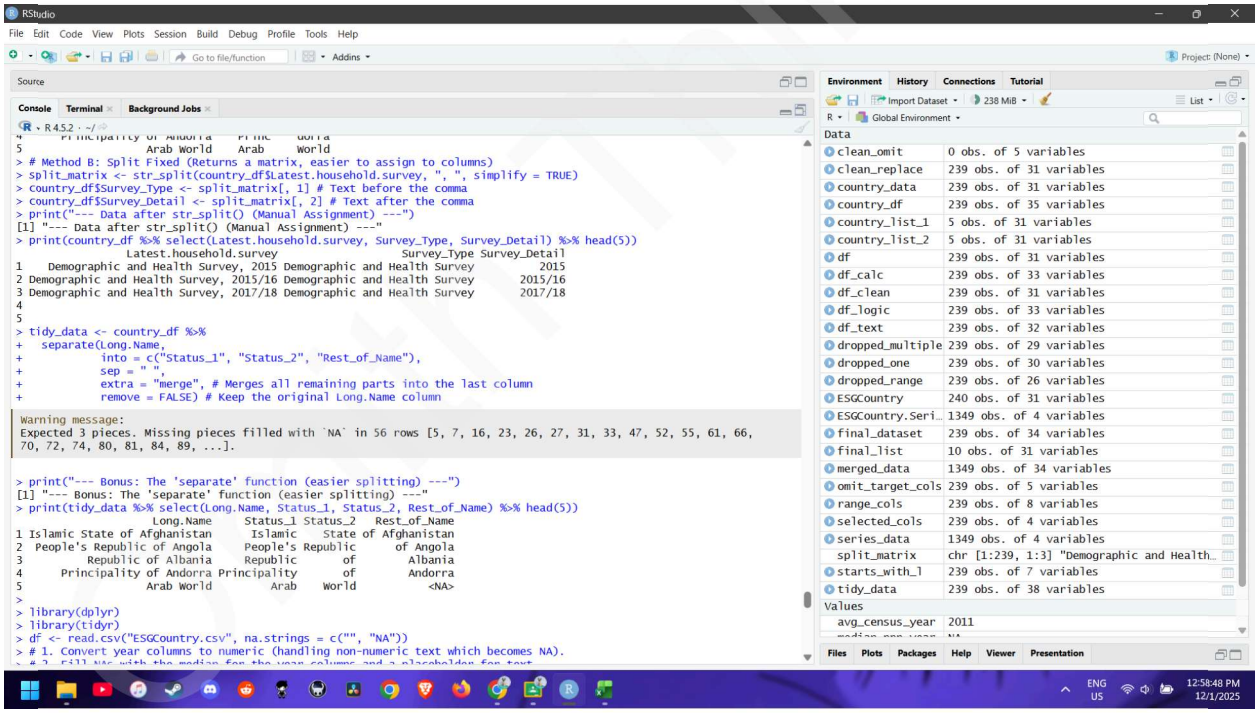
S097

SHETH LUJ AND SIR MV COLLEGE
Subject: Data Analysis with SAS / SPSS /R



```
239              NA
> print(head(clean_replace[, c("Short.Name", "Income.Group", "Latest.population.census", "PPP.survey.year")]))
          Short.Name       Income.Group                                    Latest.population.census
1        Afghanistan         Low income                                                        1979
2             Angola Lower middle income                                                        2014
3            Albania Upper middle income                                                        2011
4            Andorra         High income 2011. Population data compiled from administrative registers.
5         Arab World      Not Classified                                                        2011
6 United Arab Emirates         High income                                                        2010
  PPP.survey.year
1              NA
2              NA
3              NA
4              NA
5              NA
6              NA
> print("--- Remaining NAs after replacement ---")
[1] "--- Remaining NAs after replacement ---"
> na_counts_after <- colSums(is.na(clean_replace))
> print(na_counts_after[c("Income.Group", "Latest.population.census", "PPP.survey.year")])
          Income.Group Latest.population.census          PPP.survey.year
                     0                        0                      239
>
>
>
> # R Script: Text Manipulation with stringr (Adapted for ESGCountry.csv)
> library(stringr)
> library(tidyr)
> library(dplyr)
> country_df <- read.csv("ESGCountry.csv")
> country_df <- country_df %>%
+   rename(alpha_2_code = X2.alpha.code)
> print("--- Original Dataset (Key Columns) ---")
[1] "--- Original Dataset (Key Columns) ---"
> print(head(country_df[, c("Country.Code", "Long.Name", "Latest.household.survey")], 5))
  Country.Code                          Long.Name                        Latest.household.survey
1          AFG Islamic State of Afghanistan    Demographic and Health Survey, 2015
2          AGO  People's Republic of Angola Demographic and Health Survey, 2015/16
3          ALB          Republic of Albania Demographic and Health Survey, 2017/18
4          AND      Principality of Andorra
5          ARB                   Arab World
```

Environment	
Data	
clean_omit	0 obs. of 5 variables
clean_replace	239 obs. of 31 variables
country_data	239 obs. of 31 variables
country_df	239 obs. of 35 variables
country_list_1	5 obs. of 31 variables
country_list_2	5 obs. of 31 variables
df	239 obs. of 31 variables
df_calc	239 obs. of 33 variables
df_clean	239 obs. of 31 variables
df_logic	239 obs. of 33 variables
df_text	239 obs. of 32 variables
dropped_multiple	239 obs. of 29 variables
dropped_one	239 obs. of 30 variables
dropped_range	239 obs. of 26 variables
ESGCountry	240 obs. of 31 variables
ESGCountry.Seri...	1349 obs. of 4 variables
final_dataset	239 obs. of 34 variables
final_list	10 obs. of 31 variables
merged_data	1349 obs. of 34 variables
omit_target_cols	239 obs. of 5 variables
range_cols	239 obs. of 8 variables
selected_cols	239 obs. of 4 variables
series_data	1349 obs. of 4 variables
split_matrix	chr [1:239, 1:3] "Demographic and Health...
starts_with_l	239 obs. of 7 variables
tidy_data	239 obs. of 38 variables
Values	
avg_census_year	2011



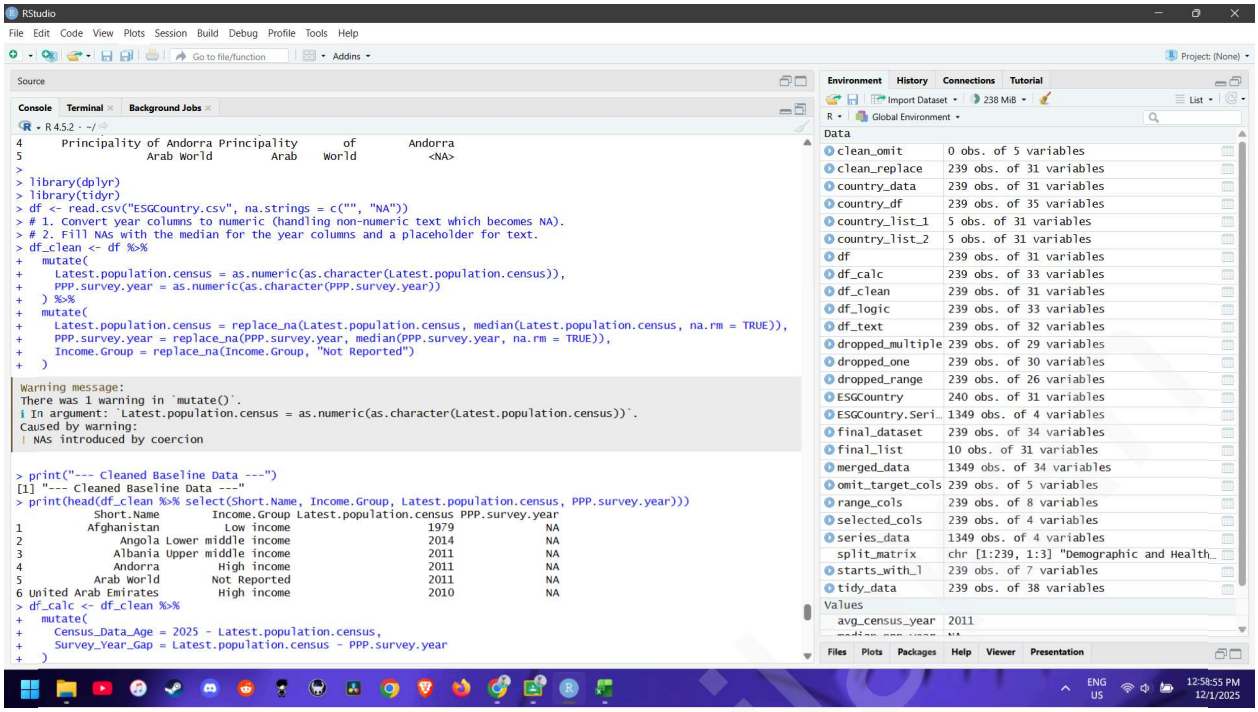
```
5                 Arab World    Arab     World
> # Method B: Split Fixed (Returns a matrix, easier to assign to columns)
> split_matrix <- str_split(country_df$Latest.household.survey, ", ", simplify = TRUE)
> country_df$Survey_Type <- split_matrix[, 1] # Text before the comma
> country_df$Survey_Detail <- split_matrix[, 2] # Text after the comma
> print("--- Data after str_split() (Manual Assignment) ---")
[1] "--- Data after str_split() (Manual Assignment) ---"
> print(country_df %>% select(Latest.household.survey, Survey_Type, Survey_Detail) %>% head(5))
                Latest.household.survey                 Survey_Type Survey_Detail
1    Demographic and Health Survey, 2015 Demographic and Health Survey         2015
2 Demographic and Health Survey, 2015/16 Demographic and Health Survey      2015/16
3 Demographic and Health Survey, 2017/18 Demographic and Health Survey      2017/18
4
5
> tidy_data <- country_df %>%
+   separate(Long.Name,
+            into = c("Status_1", "Status_2", "Rest_of_Name"),
+            sep = " ",
+            extra = "merge", # Merges all remaining parts into the last column
+            remove = FALSE) # Keep the original Long.Name column

 Warning message:
Expected 3 pieces. Missing pieces filled with `NA` in 56 rows [5, 7, 16, 23, 26, 27, 31, 33, 47, 52, 55, 61, 66,
70, 72, 74, 80, 81, 84, 89, ...].


> print("--- Bonus: The 'separate' function (easier splitting) ---")
[1] "--- Bonus: The 'separate' function (easier splitting) ---"
> print(tidy_data %>% select(Long.Name, Status_1, Status_2, Rest_of_Name) %>% head(5))
                Long.Name     Status_1 Status_2   Rest_of_Name
1 Islamic State of Afghanistan      Islamic    State of Afghanistan
2  People's Republic of Angola     People's Republic      of Angola
3         Republic of Albania     Republic        of        Albania
4      Principality of Andorra Principality        of        Andorra
5                 Arab World         Arab    World           <NA>
>
> library(dplyr)
> library(tidyr)
> df <- read.csv("ESGCountry.csv", na.strings = c("", "NA"))
> # 1. Convert year columns to numeric (handling non-numeric text which becomes NA).
```

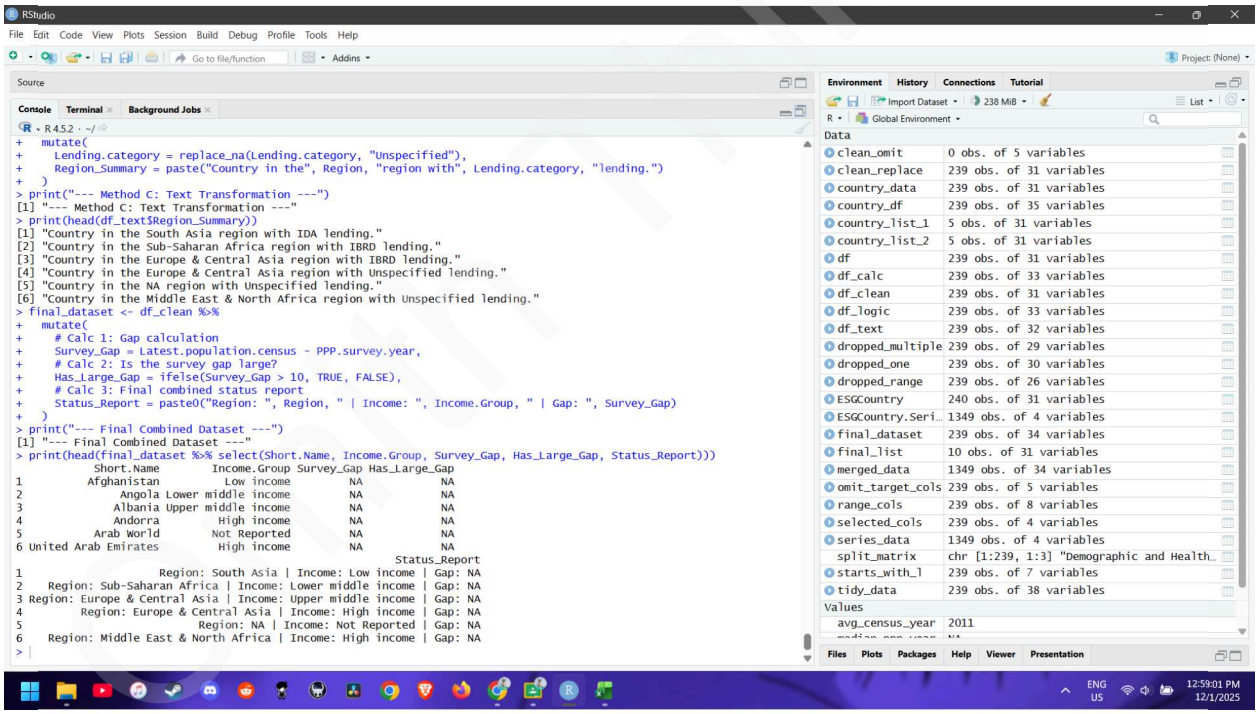
Environment	
Data	
clean_omit	0 obs. of 5 variables
clean_replace	239 obs. of 31 variables
country_data	239 obs. of 31 variables
country_df	239 obs. of 35 variables
country_list_1	5 obs. of 31 variables
country_list_2	5 obs. of 31 variables
df	239 obs. of 31 variables
df_calc	239 obs. of 33 variables
df_clean	239 obs. of 31 variables
df_logic	239 obs. of 33 variables
df_text	239 obs. of 32 variables
dropped_multiple	239 obs. of 29 variables
dropped_one	239 obs. of 30 variables
dropped_range	239 obs. of 26 variables
ESGCountry	240 obs. of 31 variables
ESGCountry.Seri...	1349 obs. of 4 variables
final_dataset	239 obs. of 34 variables
final_list	10 obs. of 31 variables
merged_data	1349 obs. of 34 variables
omit_target_cols	239 obs. of 5 variables
range_cols	239 obs. of 8 variables
selected_cols	239 obs. of 4 variables
series_data	1349 obs. of 4 variables
split_matrix	chr [1:239, 1:3] "Demographic and Health...
starts_with_l	239 obs. of 7 variables
tidy_data	239 obs. of 38 variables
Values	
avg_census_year	2011

Omith Thilakan
S097

SHETH LUJ AND SIR MV COLLEGE

Subject: Data Analysis with SAS / SPSS /R





Omith Thilakan

S097