DATA PREPARATION

TEJO SAI KRISHNA KOTTE

10/15/23

Importing the datasets

```
# Importing the two csv files
data1<-read.csv('survey1.csv')
data2<-read.csv('survey2.csv')
dim(data1)
```

```
## [1] 1190   27
```

```
dim(data2)
```

```
## [1] 69 27
```

Combining the datasets

```
# Combining the two dataframes to a single dataframe called survey
survey <- rbind(data1, data2)
dim(survey)
```

```
## [1] 1259   27
```

By comparing the dimensions we can see that we have all the rows and columns combined successfully

Data Cleaning

```
# Lets look for some missing data first
missing_data <- is.na(survey)

# Summarize missing values
summary_missing <- colSums(missing_data)

# Print the summary
print(summary_missing)
```

```
##                  Timestamp                        Age                     Gender
##                          0                          1                          0
##                    Country                      state              self_employed
##                          0                        514                         19
##            family_history                  treatment            work_interfere
##                          1                          1                        256
##              no_employees                remote_work              tech_company
##                          0                          0                          1
##                   benefits              care_options          wellness_program
##                          0                          0                          0
##                  seek_help                  anonymity                      leave
##                          0                          0                          0
## mental_health_consequence   phys_health_consequence                  coworkers
##                          0                          0                          0
##                 supervisor   mental_health_interview     phys_health_interview
##                          0                          0                          0
##         mental_vs_physical           obs_consequence                   comments
##                          0                          0                       1091
```

We can see that we have some columns that have missing values. We will first drop the comments,state,work_interfere columns because they have almost 90,50,25 percent of the missing values respectively. This makes no sense to analyse these columns

```
# droping the columns
survey <- select(survey, -comments, -state,-work_interfere,-mental_health_consequence,-phys_health_conse
dim(survey)
```

```
## [1] 1259   17
```

```
head(survey)
```

```
##       Timestamp   Age Gender        Country self_employed family_history
## 1 8/28/14 10:07 -1726   male United Kingdom            No             No
## 2 8/27/14 12:39   -29   Male  United States            No             No
## 3 8/30/14 20:55    -1      p  United States           Yes            Yes
## 4    $41,878.49     0      m         Canada           Yes             No
## 5 8/27/14 12:29     0   Male  United States            No             No
## 6 8/28/14 10:35     5   Male  United States            No             No
##   treatment   no_employees remote_work tech_company   benefits care_options
## 1       Yes         26-100           2           No         No           No
## 2        No More than 1000           1           No        Yes           No
## 3       Yes          5-Jan         Yes          Yes        Yes          Yes
## 4        No          44201          No          Yes         No          Yes
## 5       Yes       500-1000          No          Yes        Yes          Yes
## 6        No        100-500                      Yes Don't know     Not sure
##   wellness_program  seek_help  anonymity              leave
## 1               No         No Don't know Somewhat difficult
## 2       Don't know        Yes Don't know         Don't know
## 3              Yes        Yes        Yes          Very easy
## 4              Yes         No Don't know         Don't know
## 5               No Don't know        Yes         Don't know
## 6               No         No Don't know      Somewhat easy
##   mental_health_interview
```

```
## 1                       No
## 2                       No
## 3                      Yes
## 4                       No
## 5                       No
## 6                       No
```

we can see that we have successfully dropped the columns. We now drop the rows that have fewer missing values in the columns like Age,self_employed,family_history,treatment,tech_company

```
# Remove rows with missing values in Age,self_employed,family_history,treatment,tech_company columns
survey <- na.omit(survey, cols = "Age","self_employed","family_history","treatment","tech_company")
dim(survey)
```

```
## [1] 1237   17
```

We have successfully removed all the NA values. Lets recheck once.

```
# Lets look for some missing data first
missing_data <- is.na(survey)

# Summarize missing values
summary_missing <- colSums(missing_data)

# Print the summary
print(summary_missing)
```

```
##               Timestamp                     Age                  Gender
##                       0                       0                       0
##                 Country           self_employed         family_history
##                       0                       0                       0
##               treatment            no_employees            remote_work
##                       0                       0                       0
##            tech_company                benefits            care_options
##                       0                       0                       0
##        wellness_program               seek_help               anonymity
##                       0                       0                       0
##                   leave mental_health_interview
##                       0                       0
```

```
# Reordering the indexes
survey <- data.frame(survey, row.names = NULL)
dim(survey)
```

```
## [1] 1237   17
```

```
head(survey)
```

```
##       Timestamp   Age Gender        Country self_employed family_history
## 1 8/28/14 10:07 -1726   male United Kingdom            No             No
## 2 8/27/14 12:39   -29   Male  United States            No             No
```

```
## 3 8/30/14 20:55    -1      p  United States           Yes            Yes
## 4    $41,878.49     0      m         Canada           Yes             No
## 5 8/27/14 12:29     0   Male  United States            No             No
## 6 8/28/14 10:35     5   Male  United States            No             No
##   treatment   no_employees remote_work tech_company   benefits care_options
## 1       Yes         26-100           2           No         No           No
## 2        No More than 1000           1           No        Yes           No
## 3       Yes          5-Jan         Yes          Yes        Yes          Yes
## 4        No          44201          No          Yes         No          Yes
## 5       Yes       500-1000          No          Yes        Yes          Yes
## 6        No        100-500                      Yes Don't know     Not sure
##   wellness_program  seek_help  anonymity              leave
## 1               No         No Don't know Somewhat difficult
## 2       Don't know        Yes Don't know         Don't know
## 3              Yes        Yes        Yes          Very easy
## 4              Yes         No Don't know         Don't know
## 5               No Don't know        Yes         Don't know
## 6               No         No Don't know      Somewhat easy
##   mental_health_interview
## 1                      No
## 2                      No
## 3                     Yes
## 4                      No
## 5                      No
## 6                      No
```

```
tail(survey)
```

```
##           Timestamp Age Gender       Country self_employed family_history
## 1232  8/20/15 16:52  29   male United States            No            Yes
## 1233  8/25/15 19:59  36   Male            US            No            Yes
## 1234  9/12/15 11:17   0   male            UK            No             No
## 1235   9/26/15 1:07  32   Male United States            No            Yes
## 1236 11/30/15 21:25  46      f United States            No             No
## 1237   2/1/16 23:04  25   Male United States            No            Yes
##      treatment   no_employees remote_work tech_company   benefits care_options
## 1232       Yes        100-500         Yes          Yes        Yes          Yes
## 1233        No More than 1000          No           No Don't know           No
## 1234       Yes         26-100          No          Yes         No           No
## 1235       Yes         26-100         Yes          Yes        Yes          Yes
## 1236        No        100-500         Yes          Yes         No          Yes
## 1237       Yes         26-100          No           No        Yes          Yes
##      wellness_program seek_help  anonymity              leave
## 1232              Yes        No        Yes         Don't know
## 1233              Yes       Yes Don't know      Somewhat easy
## 1234               No        No Don't know      Somewhat easy
## 1235               No        No        Yes Somewhat difficult
## 1236               No        No Don't know         Don't know
## 1237               No        No        Yes         Don't know
##      mental_health_interview
## 1232                      No
## 1233                      No
## 1234                      No
## 1235                      No
```

```
## 1236                        No
## 1237                        No
```

we have successfully reordered the indexes of our dataframe. We can see that we have successfully removed all the null values while keeping most of our data. Now its time to check the structure of our data.

```
str(survey)
```

```
## 'data.frame':    1237 obs. of  17 variables:
##  $ Timestamp              : chr  "8/28/14 10:07" "8/27/14 12:39" "8/30/14 20:55" "$41,878.49" ...
##  $ Age                    : num  -1726 -29 -1 0 0 ...
##  $ Gender                 : chr  "male" "Male" "p" "m" ...
##  $ Country                : chr  "United Kingdom" "United States" "United States" "Canada" ...
##  $ self_employed          : chr  "No" "No" "Yes" "Yes" ...
##  $ family_history         : chr  "No" "No" "Yes" "No" ...
##  $ treatment              : chr  "Yes" "No" "Yes" "No" ...
##  $ no_employees           : chr  "26-100" "More than 1000" "5-Jan" "44201" ...
##  $ remote_work            : chr  "2" "1" "Yes" "No" ...
##  $ tech_company           : chr  "No" "No" "Yes" "Yes" ...
##  $ benefits               : chr  "No" "Yes" "Yes" "No" ...
##  $ care_options           : chr  "No" "No" "Yes" "Yes" ...
##  $ wellness_program       : chr  "No" "Don't know" "Yes" "Yes" ...
##  $ seek_help              : chr  "No" "Yes" "Yes" "No" ...
##  $ anonymity              : chr  "Don't know" "Don't know" "Yes" "Don't know" ...
##  $ leave                  : chr  "Somewhat difficult" "Don't know" "Very easy" "Don't know" ...
##  $ mental_health_interview: chr  "No" "No" "Yes" "No" ...
```

The structure looks good except for one column (TimeStamp).We will take a look at the summary of the data first.

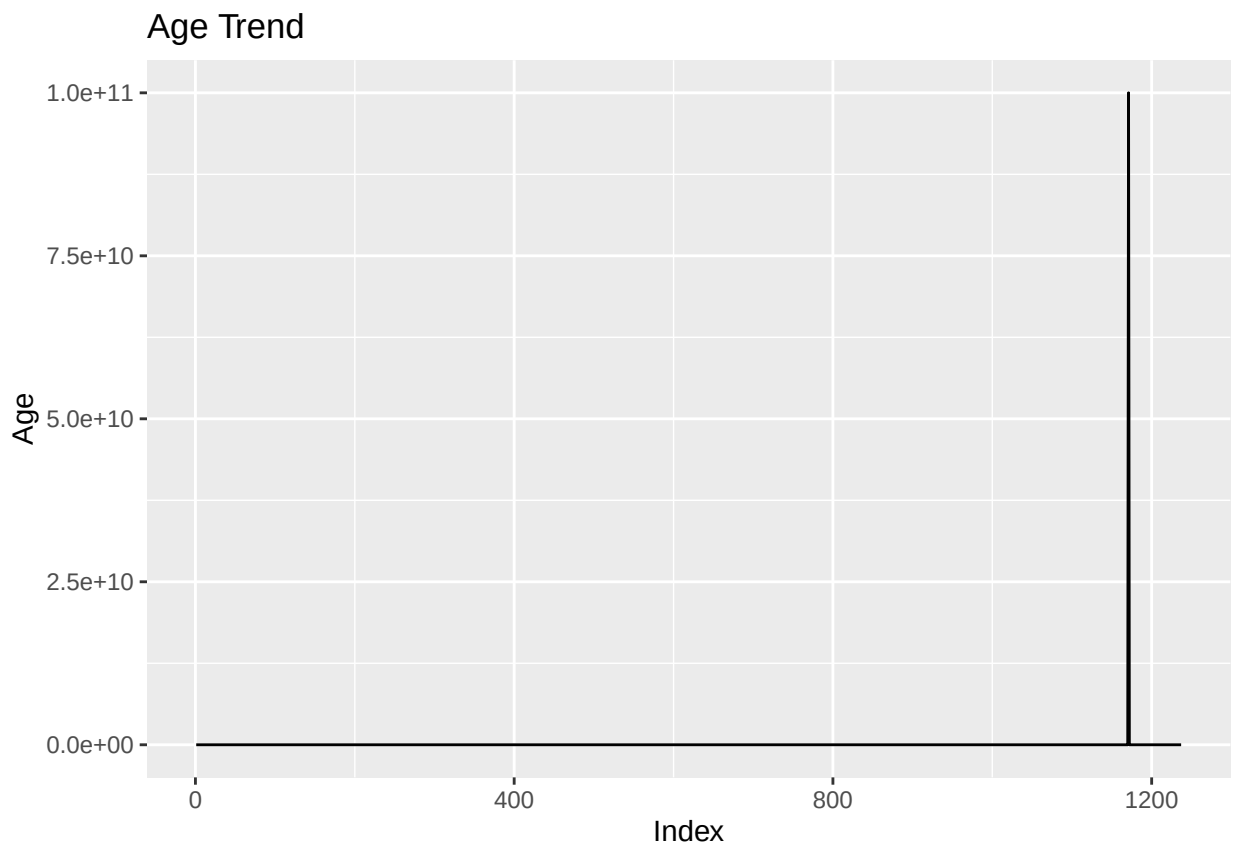
```
summary(survey)
```

```
##   Timestamp              Age                Gender            Country
##  Length:1237        Min.   :-1.726e+03   Length:1237        Length:1237
##  Class :character   1st Qu.: 2.700e+01   Class :character   Class :character
##  Mode  :character   Median : 3.100e+01   Mode  :character   Mode  :character
##                     Mean   : 8.084e+07
##                     3rd Qu.: 3.600e+01
##                     Max.   : 1.000e+11
##  self_employed      family_history      treatment          no_employees
##  Length:1237        Length:1237        Length:1237        Length:1237
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##  remote_work        tech_company         benefits          care_options
##  Length:1237        Length:1237        Length:1237        Length:1237
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
```

```
##
##  wellness_program    seek_help          anonymity            leave
##  Length:1237        Length:1237        Length:1237        Length:1237
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##  mental_health_interview
##  Length:1237
##  Class :character
##  Mode  :character
##
##
##
```

From the summary of Age column above we can see that the minimum value in the Age column is -1726. We now will clean the Age column as Age cannot be negative. And also the mean value and max value are also abnormal for Age column.

```
# Plotting using ggplot2
ggplot(data = survey, aes(x = seq_along(Age), y = Age)) +
  geom_line() +
  labs(x = "Index", y = "Age", title = "Age Trend")
```



```r
# Remove rows with empty Timestamp cells
survey <- survey %>%
  filter(Age != "")
# Filtering the Age column
survey <- survey %>%
  filter(Age > 0, Age < 100)
dim(survey)
```

```
## [1] 1228   17
```

As you can see there are some outliers in the Age column.We filtered out those using the filter method.

```r
#Summary on Age column
summary(survey$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   27.00   31.00   31.89   36.00   72.00
```

Now the Age column looks good. We will clean the Timestamp column.

```r
# Remove rows with empty Timestamp cells
survey <- survey %>%
  filter(Timestamp != "")
dim(survey)
```

```
## [1] 1226   17
```

In the Timestamp column, the data structure is not consistent. Some of the records have "Wednesday, August 27, 2014" date format. But most of the records have "9/9/14 13:49" this format.We shall remove the records with the latter structure.

```r
# Remove rows bad structure of Timestamp
survey <- survey %>%
  filter(Timestamp != "Wednesday, August 27, 2014")
dim(survey)
```

```
## [1] 1213   17
```

The Timestamp column is of char type we will now convert it into timestamp type

```r
# Convert "Timestamp" to datetime type using POSIXct.
survey$Timestamp <- as.POSIXct(survey$Timestamp, format = "%m/%d/%y %H:%M")
str(survey)
```

```
## 'data.frame':    1213 obs. of  17 variables:
##  $ Timestamp              : POSIXct, format: "2014-08-28 10:35:00" "2014-08-29 09:10:00" ...
##  $ Age                    : num  5 8 11 18 18 18 18 18 18 18 ...
##  $ Gender                 : chr  "Male" "A little about you" "male" "Male" ...
##  $ Country                : chr  "United States" "Bahamas, The" "United States" "Netherlands" ...
##  $ self_employed          : chr  "No" "Yes" "Yes" "No" ...
```

```
##  $ family_history         : chr  "No" "Yes" "No" "No" ...
##  $ treatment              : chr  "No" "Yes" "No" "No" ...
##  $ no_employees           : chr  "100-500" "5-Jan" "5-Jan" "25-Jun" ...
##  $ remote_work            : chr  "" "Yes" "Yes" "No" ...
##  $ tech_company           : chr  "Yes" "Yes" "Yes" "Yes" ...
##  $ benefits               : chr  "Don't know" "Yes" "No" "No" ...
##  $ care_options           : chr  "Not sure" "Yes" "Yes" "Not sure" ...
##  $ wellness_program       : chr  "No" "Yes" "No" "No" ...
##  $ seek_help              : chr  "No" "Yes" "No" "No" ...
##  $ anonymity              : chr  "Don't know" "Yes" "Yes" "Don't know" ...
##  $ leave                  : chr  "Somewhat easy" "Very easy" "Very easy" "Somewhat difficult" ...
##  $ mental_health_interview: chr  "No" "Yes" "No" "No" ...
```

You can see that we changed the format of our Timestamp label. Lets move on to the gender column.

```
# Remove rows with empty Gender cells
survey <- survey %>%
  filter(Gender != "")
dim(survey)
```

```
## [1] 1213   17
```

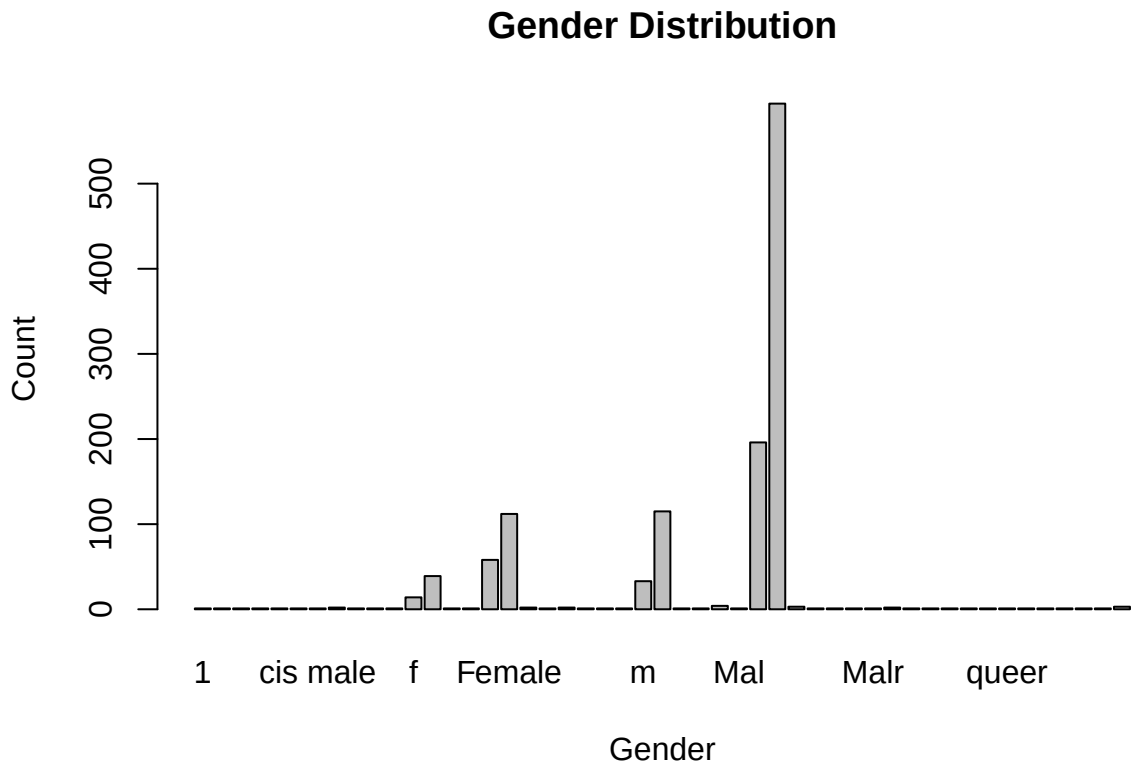
```
# Create a bar plot using base R
gender_counts <- table(survey$Gender)
print(gender_counts)
```

```
##
##                                          1
##                                          1
##                                          2
##                                          1
##                            A little about you
##                                          1
##                                   Agender
##                                          1
##                                 Androgyne
##                                          1
##                                Cis Female
##                                          1
##                                  cis male
##                                          1
##                                  Cis Male
##                                          2
##                                   Cis Man
##                                          1
##                             cis-female/femme
##                                          1
##                                      Enby
##                                          1
##                                          f
##                                         14
##                                          F
##                                         39
```

```
##                                   femail
##                                        1
##                                  Femake
##                                        1
##                                   female
##                                       58
##                                   Female
##                                      112
##                                 Female 
##                                        2
##                               Female (cis)
##                                        1
##                              Female (trans)
##                                        2
##                                    fluid
##                                        1
##                                 Genderqueer
##                                        1
##                              Guy (-ish) ^_^
##                                        1
##                                        m
##                                       33
##                                        M
##                                      115
##                                     Mail
##                                        1
##                                    maile
##                                        1
##                                     Make
##                                        4
##                                      Mal
##                                        1
##                                     male
##                                      196
##                                     Male
##                                      594
##                                    Male 
##                                        3
##                                 Male (CIS)
##                                        1
##                        male leaning androgynous
##                                        1
##                                 Male-ish
##                                        1
##                                     Malr
##                                        1
##                                      Man
##                                        2
##                                     msle
##                                        1
##                                      Nah
##                                        1
##                                   Neuter
##                                        1
```

```
##                                     non-binary
##                                              1
## ostensibly male, unsure what that really means
##                                              1
##                                         queer
##                                              1
##                                   queer/she/they
##                                              1
##                             something kinda male?
##                                              1
##                                    Trans woman
##                                              1
##                                  Trans-female
##                                              1
##                                         woman
##                                              1
##                                         Woman
##                                              3
```

```
barplot(gender_counts, main = "Gender Distribution", xlab = "Gender", ylab = "Count")
```



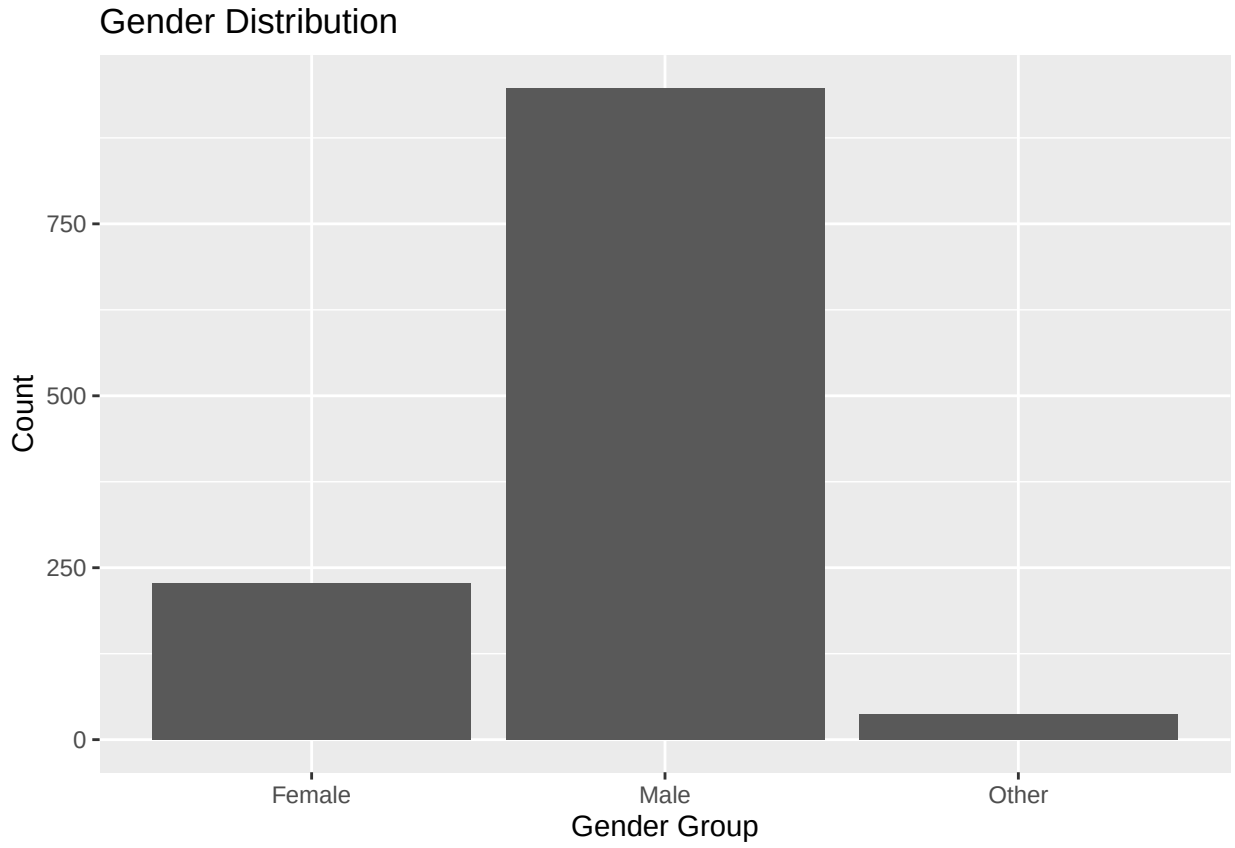
We will simply have three categories namely Male, Female and Other.

```
# Define a function to group genders
group_genders <- function(gender) {
  gender <- tolower(gender)  # Convert to lowercase
```

```
  if (gender %in% c("male", "m", "man", "make", "cis male", "cis man")) return("Male")
  if (gender %in% c("female", "f", "woman", "cis female", "cis woman")) return("Female")
  return("Other")
}

# Apply the grouping function to the Gender column
survey$GroupedGender <- sapply(survey$Gender, group_genders)

# Create a bar plot using ggplot2
ggplot(data = survey, aes(x = GroupedGender)) +
  geom_bar() +
  labs(x = "Gender Group", y = "Count", title = "Gender Distribution")
```




```
# Drop the old Gender column
survey <- select(survey, -Gender)
dim(survey)
```

```
## [1] 1213   17
```

We will now move on to the Country column.

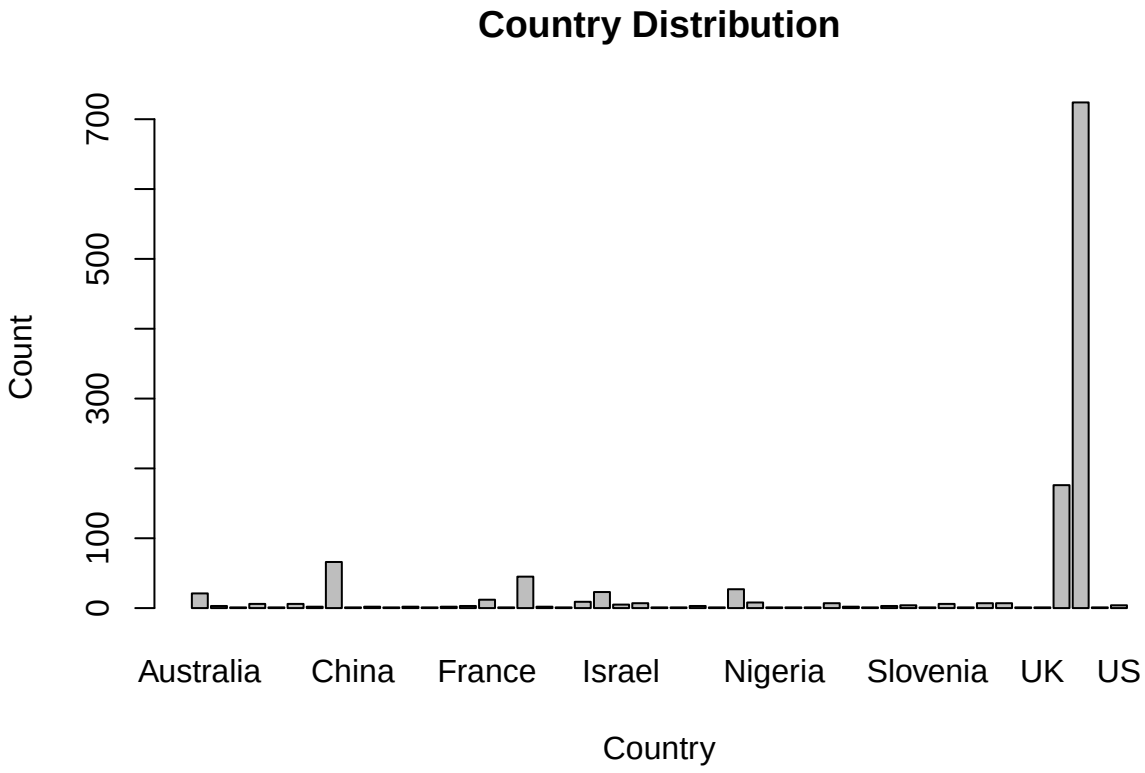
```
# Remove rows with empty Gender cells
survey <- survey %>%
  filter(Country != "")
dim(survey)
```

```
## [1] 1213   17
```

```r
# We will plot and see the unique countries
country_counts <- table(survey$Country)
print(country_counts)
```

```
##
##              Australia                Austria           Bahamas, The
##                     21                      3                      1
##                Belgium Bosnia and Herzegovina                 Brazil
##                      6                      1                      6
##               Bulgaria                 Canada                  China
##                      2                     66                      1
##               Colombia             Costa Rica                Croatia
##                      2                      1                      2
##         Czech Republic                Denmark                Finland
##                      1                      2                      3
##                 France                Georgia                Germany
##                     12                      1                     45
##                 Greece                Hungary                  India
##                      2                      1                      9
##                Ireland                 Israel                  Italy
##                     23                      5                      7
##                  Japan                 Latvia                 Mexico
##                      1                      1                      3
##                Moldova            Netherlands            New Zealand
##                      1                     27                      8
##                Nigeria                 Norway            Philippines
##                      1                      1                      1
##                 Poland               Portugal                Romania
##                      7                      2                      1
##                 Russia              Singapore               Slovenia
##                      3                      4                      1
##           South Africa                  Spain                 Sweden
##                      6                      1                      7
##            Switzerland               Thailand                     UK
##                      7                      1                      1
##         United Kingdom          United States                Uruguay
##                    176                    724                      1
##                     US
##                      4
```

```r
barplot(country_counts, main = "Country Distribution", xlab = "Country", ylab = "Count")
```



The country column looks good.The no_employees column has many inconsistent values. We will drop it.

```
# Drop no_employees column
survey <- select(survey, columns=-no_employees)
```

Now we shall move on to the self_employed column.

```
# Drop the rows with empty cells
survey <- survey %>%
  filter(self_employed != "")
dim(survey)
```

```
## [1] 1209   16
```

we have successfully removed 4 rows. Now we shall look into the values in the self_employed column.

```
# Get the counts of unique values in the "self_employed" column
self_employed_counts <- table(survey$self_employed)

# Print the counts
print(self_employed_counts)
```

```
##
##   No  Yes
## 1070  139
```

We can see that there are two unique values with the counts above. This column looks clean. We will now move onto the next column family History

```
# Drop the rows with empty cells
survey <- survey %>%
  filter(family_history != "")
dim(survey)
```

```
## [1] 1209   16
```

All the rows are clean Now we shall look into the values in the family_history column.

```
# Get the counts of unique values in the "self_employed" column
family_history_counts <- table(survey$family_history)

# Print the counts
print(family_history_counts)
```

```
##
##  No Yes
## 734 475
```

We can see that there are two unique values with the counts above. This column looks clean. We will now move onto the next column treatment.

```
# Drop the rows with empty cells
survey <- survey %>%
  filter(treatment != "")
dim(survey)
```

```
## [1] 1209   16
```

All the rows are clean Now we shall look into the values in the treatment column.

```
treatment_counts <- table(survey$treatment)

# Print the counts
print(treatment_counts)
```

```
##
##   -   N  No   Y Yes
##   2   2 595   1 609
```

We can see that we have two rows with - value. We will drop these two rows and then we will convert the N to No and Y to Yes respectively.

```
# Drop the rows with '-' value
survey <- survey %>%
  filter(treatment != "-")
dim(survey)
```

```
## [1] 1207   16
```

```r
# Replace values in the "treatment" column
survey$treatment <- ifelse(survey$treatment == "N", "No", ifelse(survey$treatment == "Y", "Yes", survey$
```

We will print te unique values once again to cross check.

```r
treatment_counts <- table(survey$treatment)

# Print the counts
print(treatment_counts)
```

```
##
##  No Yes
## 597 610
```

Now the treatment column looks good. We shall move onto the next column remote_work.

```r
# Drop the rows with empty cells
survey <- survey %>%
  filter(remote_work != "")
dim(survey)
```

```
## [1] 1206   16
```

Now we shall look into the values in the remote_work column.

```r
remote_work_counts <- table(survey$remote_work)

# Print the counts
print(remote_work_counts)
```

```
##
##   -  No Yes
##   2 847 357
```

We can see that we have two rows with - value. We will drop these two rows.

```r
# Drop the rows with '-' value
survey <- survey %>%
  filter(remote_work != "-")
dim(survey)
```

```
## [1] 1204   16
```

This column is clean. we will now move onto tech_company column.

```r
# Drop the rows with empty cells
survey <- survey %>%
  filter(tech_company != "")
dim(survey)
```

```
## [1] 1204   16
```

Now we shall look into the values in the tech_company column.

```
tech_company_counts <- table(survey$tech_company)

# Print the counts
print(tech_company_counts)
```

```
##
##   -  No Yes
##   1 220 983
```

We have one - value. we shall drop that row

```
# Drop the rows with '-' value
survey <- survey %>%
  filter(tech_company != "-")
dim(survey)
```

```
## [1] 1203   16
```

Now we will move onto the benefits column

```
# Drop the rows with empty cells
survey <- survey %>%
  filter(benefits != "")
dim(survey)
```

```
## [1] 1203   16
```

Now we shall look into the values in the benefits column.

```
benefits_counts <- table(survey$benefits)

# Print the counts
print(benefits_counts)
```

```
##
## Don't know         No   not sure   Not sure        Yes
##        388        352          2          2        459
```

```
# Clean the "benefits" column

# Define a function to group benefits
group_benefits <- function(benefit) {
  benefit <- tolower(benefit)  # Convert to lowercase
  if (benefit %in% c("don't know", " not sure")) return("Not sure")
  if (benefit %in% c(" no ")) return("No")
  if (benefit %in% c("yes")) return("Yes")
```

```
  return("Other")
}

# Apply the grouping function to the Gender column
survey$benefits <- sapply(survey$benefits, group_benefits)
benefits_counts <- table(survey$benefits)

# Print the counts
print(benefits_counts)
```

```
##
## Not sure    Other      Yes
##      388      356      459
```

We have successfully cleaned the benefits column. We will move onto the next column care_options. Now we will move onto the benefits column

```
# Drop the rows with empty cells
survey <- survey %>%
  filter(care_options != "")
dim(survey)
```

```
## [1] 1203   16
```

Now we shall look into the values in the care_options column.

```
care_options_counts <- table(survey$care_options)

# Print the counts
print(care_options_counts)
```

```
##
##       No Not sure      Yes
##      477      302      424
```

This column looks good. Moving onto wellness_program column.

```
# Drop the rows with empty cells
survey <- survey %>%
  filter(wellness_program != "")
dim(survey)
```

```
## [1] 1203   16
```

Now we shall look into the values in the wellness_program column.

```
wellness_program_counts <- table(survey$wellness_program)

# Print the counts
print(wellness_program_counts)
```

```
##
## Don't know          No         Yes
##         177         800         226
```

This column looks good, we shall move onto the next column seek_help

```
# Drop the rows with empty cells
survey <- survey %>%
  filter(seek_help != "")
dim(survey)
```

```
## [1] 1203   16
```

Now we shall look into the values in the seek_help column.

```
seek_help_counts <- table(survey$seek_help)

# Print the counts
print(seek_help_counts)
```

```
##
## Don't know          No    not sure    Not sure         Yes
##         345         609           1           1         247
```

We can use the group_benefits function we defined above to clean this column

```
survey$seek_help <- sapply(survey$seek_help, group_benefits)
seek_help_counts <- table(survey$seek_help)

# Print the counts
print(seek_help_counts)
```

```
##
## Not sure    Other      Yes
##      345      611      247
```

We shall omit all the rows with empty cells.

```
remove_empty_rows <- function(data, column_name) {
  data %>%
    filter(vars(column_name)!="")
}

survey <- remove_empty_rows(survey,anonymity)
survey <- remove_empty_rows(survey,leave)
survey <- remove_empty_rows(survey,mental_health_consequence)
survey <- remove_empty_rows(survey,phys_health_consequence)
survey <- remove_empty_rows(survey,coworkers)
survey <- remove_empty_rows(survey,supervisor)
survey <- remove_empty_rows(survey,mental_health_interview)
survey <- remove_empty_rows(survey,phys_health_interview)
survey <- remove_empty_rows(survey,mental_vs_physical)
survey <- remove_empty_rows(survey,obs_consequence)
dim(survey)
```

```
## [1] 1203   16
```

```
counts <- table(survey$anonymity)
print(counts)
```

```
##
## Don't know         No   not sure        Yes
##        779         58          1        365
```

We can use the group_benefits function we defined above to clean this column

```
survey$anonymity <- sapply(survey$anonymity, group_benefits)
counts <- table(survey$anonymity)

# Print the counts
print(counts)
```

```
##
## Not sure    Other      Yes
##      779       59      365
```

We have cleaned all of our required columns. Now we need to save this dataframe into a csv file for later usage. # Saving the dataframe into csv file.

```
write.csv(survey, file = "survey.csv", row.names = FALSE)
```