Loading Libraries

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v tibble  3.1.4     v purrr   0.3.4
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(rvest)
```

```
##
## Attaching package: 'rvest'

## The following object is masked from 'package:readr':
##
##     guess_encoding
```

```
library(magrittr)
```

```
##
## Attaching package: 'magrittr'

## The following object is masked from 'package:purrr':
##
##     set_names

## The following object is masked from 'package:tidyr':
##
##     extract
```

```
library(ggmap)
```

```
## Warning: package 'ggmap' was built under R version 4.1.2

## Google's Terms of Service: https://cloud.google.com/maps-platform/terms/.

## Please cite ggmap if you use it! See citation("ggmap") for details.

##
## Attaching package: 'ggmap'

## The following object is masked from 'package:magrittr':
##
##     inset
```

```
library(stringr)
```

Loading Dataset

```
atheletes <- read.csv("./Olympics/athlete_events.csv", stringsAsFactors = F)
regions <- read.csv("./Olympics/noc_regions.csv", stringsAsFactors = F)
```

Data Exploration

```
summary(atheletes)
```

```
##        ID             Name               Sex                 Age
##  Min.   :     1   Length:271116      Length:271116      Min.   :10.00
##  1st Qu.: 34643   Class :character   Class :character   1st Qu.:21.00
##  Median : 68205   Mode  :character   Mode  :character   Median :24.00
##  Mean   : 68249                                         Mean   :25.56
##  3rd Qu.:102097                                         3rd Qu.:28.00
##  Max.   :135571                                         Max.   :97.00
##                                                         NA's   :9474
##      Height           Weight           Team               NOC
##  Min.   :127.0   Min.   : 25.0   Length:271116      Length:271116
##  1st Qu.:168.0   1st Qu.: 60.0   Class :character   Class :character
##  Median :175.0   Median : 70.0   Mode  :character   Mode  :character
##  Mean   :175.3   Mean   : 70.7
##  3rd Qu.:183.0   3rd Qu.: 79.0
##  Max.   :226.0   Max.   :214.0
##  NA's   :60171   NA's   :62875
##     Games                Year          Season              City
##  Length:271116      Min.   :1896   Length:271116      Length:271116
##  Class :character   1st Qu.:1960   Class :character   Class :character
##  Mode  :character   Median :1988   Mode  :character   Mode  :character
##                     Mean   :1978
##                      3rd Qu.:2002
##                      Max.   :2016
##
##     Sport              Event              Medal
##  Length:271116      Length:271116      Length:271116
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
```

```
summary(regions)
```

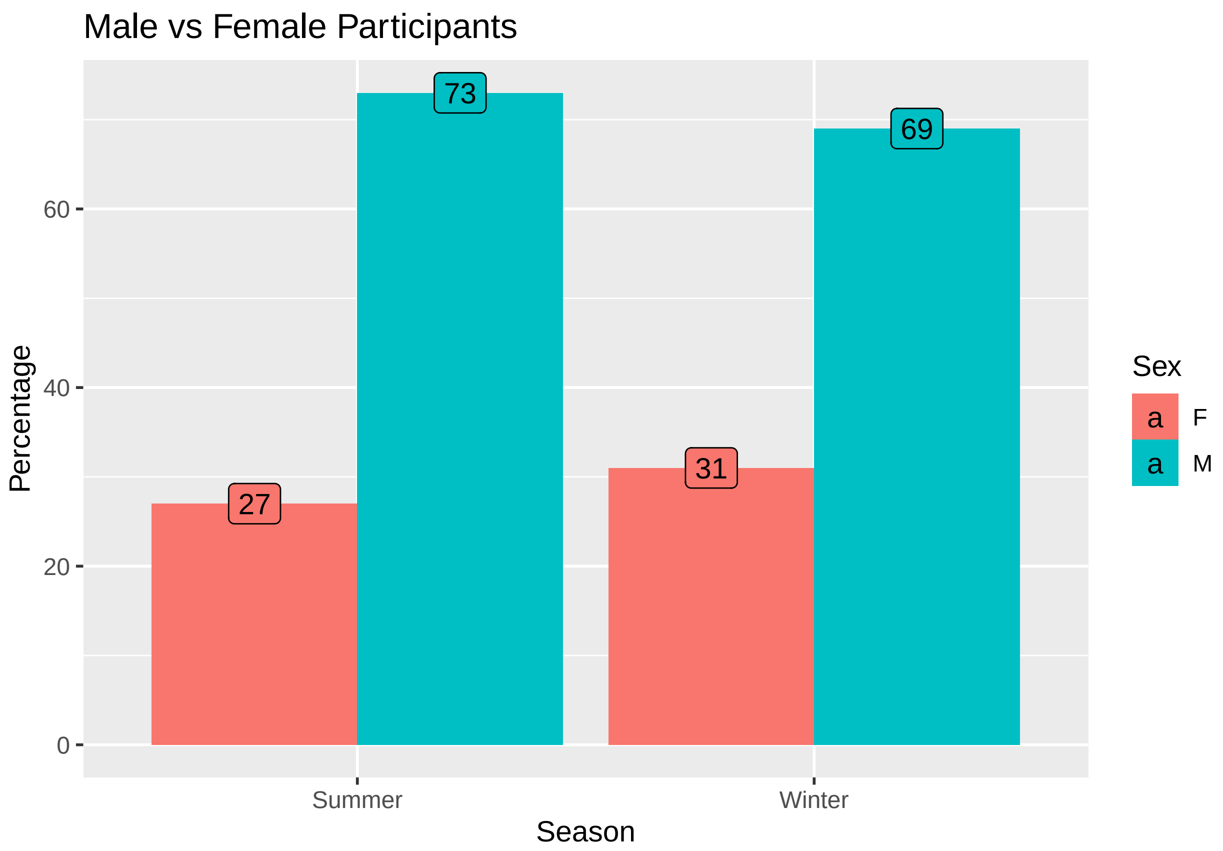
```
##      NOC               region              notes
##  Length:230         Length:230         Length:230
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
```

Sex

```
df <- atheletes %>%
  group_by(Season, Sex) %>%
  summarise(Count = n()) %>%
  mutate(Percentage = round(Count*100 / sum(Count)))
```

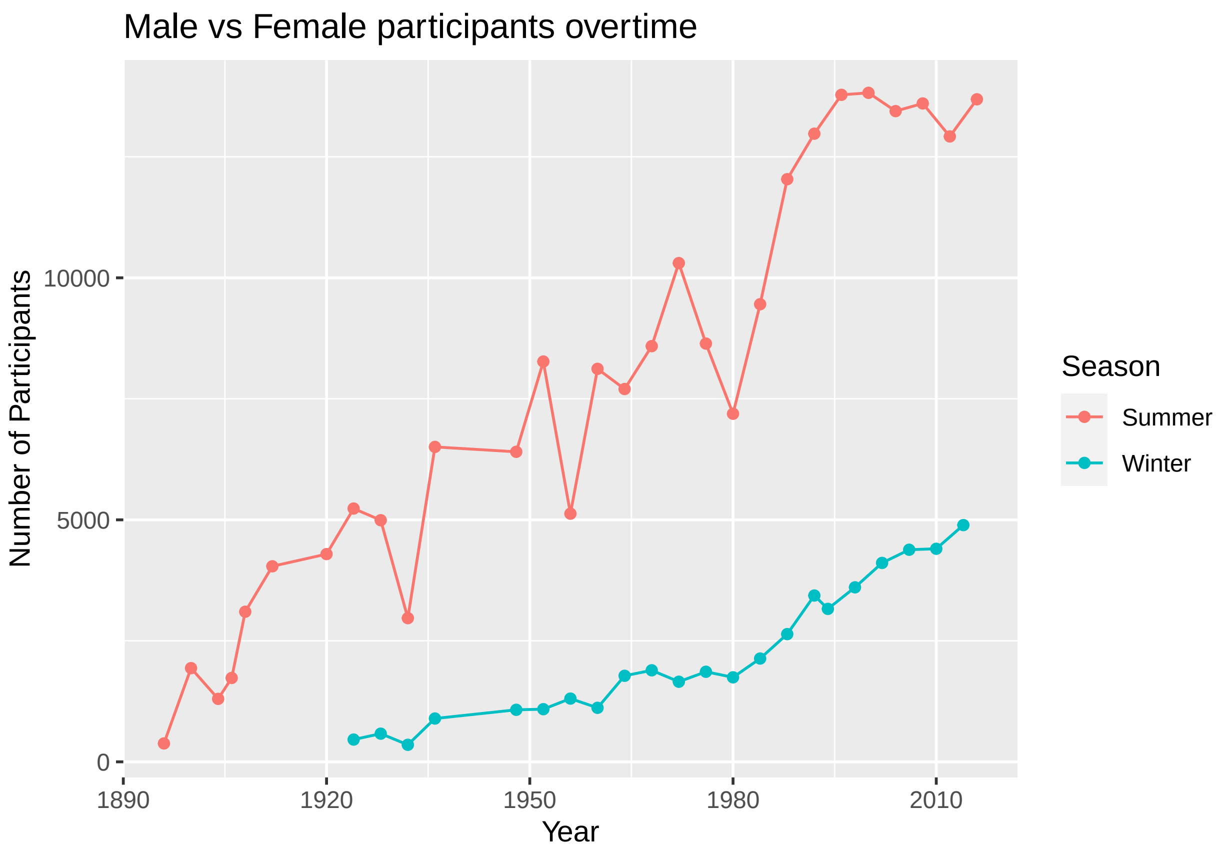
```
## `summarise()` has grouped output by 'Season'. You can override using the `.groups` argument.
```

```
df %>%
ggplot(aes(x=Season, y=Percentage, fill = Sex)) + geom_bar(stat='identity',position=position_dodge()) +
        ggtitle("Male vs Female Participants") +
        geom_label(label=df$Percentage, position = position_dodge(0.9))
```



```
atheletes %>%
  group_by(Year, Season) %>%
  summarise(NumberOfParticipants = n()) %>%
  ggplot(aes(x = Year, y = NumberOfParticipants, group = Season)) +
  geom_line(aes(color = Season)) +
  geom_point(aes(color = Season)) +
  labs(x = "Year", y = "Number of Participants", title = "Male vs Female participants overtime")
```

```
## `summarise()` has grouped output by 'Year'. You can override using the `.groups` argument.
```



```
groupMale <- atheletes %>%
              filter(Sex == "M") %>%
              group_by(Year, Season) %>%
              summarise(Number_Of_Men = n())
```

```
## ‘summarise()‘ has grouped output by ’Year’. You can override using the ‘.groups‘ argument.
```

```
groupFemale <- atheletes %>%
              filter(Sex == "F") %>%
              group_by(Year, Season) %>%
              summarise(Number_Of_Women = n())
```

```
## ‘summarise()‘ has grouped output by ’Year’. You can override using the ‘.groups‘ argument.
```

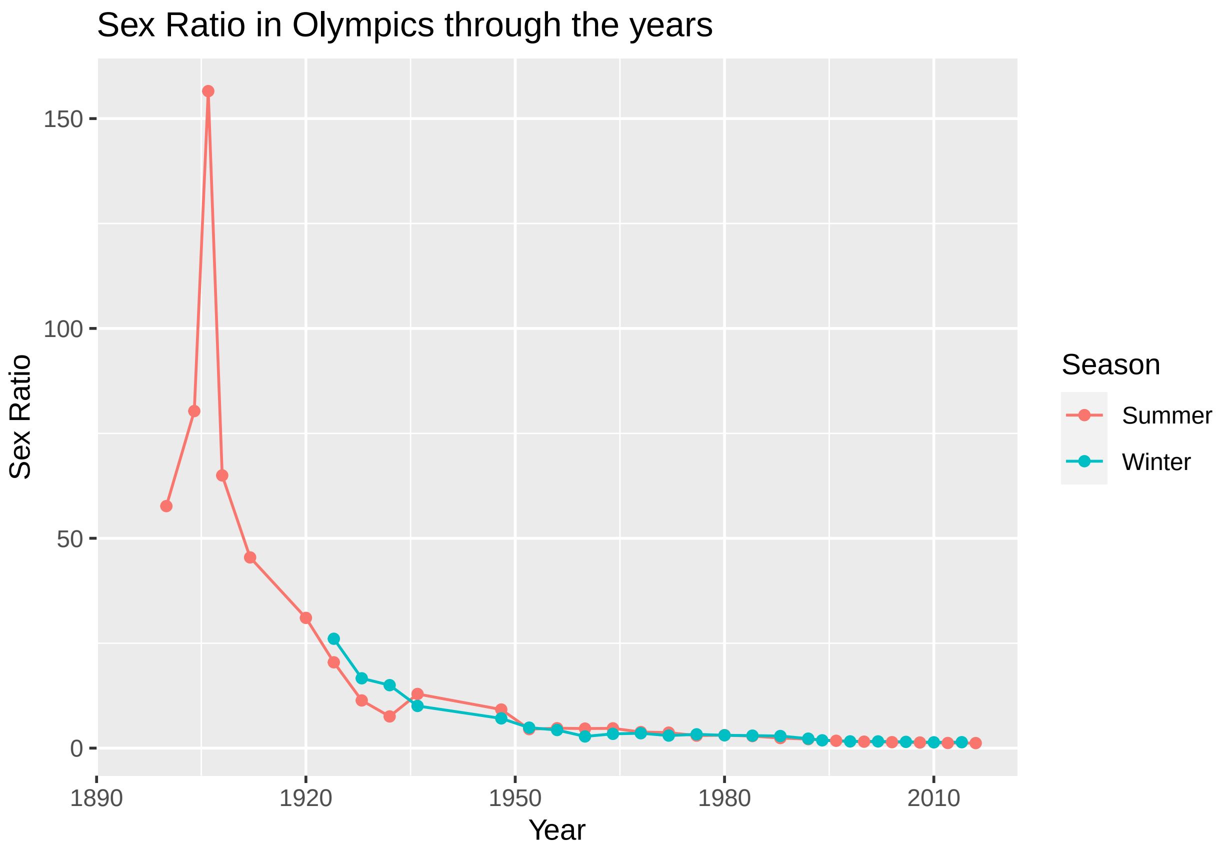
```
group <- groupMale %>%
          left_join(groupFemale) %>%
          mutate(Sex_Ratio = Number_Of_Men/Number_Of_Women)
```

```
## Joining, by = c("Year", "Season")
```

```
group %>%
    ggplot(aes(x = Year, y= Sex_Ratio, group = Season)) +
    geom_line(aes(color = Season)) +
    geom_point(aes(color = Season)) +
    labs(x = "Year", y = "Sex Ratio", title = "Sex Ratio in Olympics through the years")
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



Age

```
atheletes$Age[is.na(atheletes$Age)] <- median(atheletes$Age, na.rm = T)
cat("The median age of the athletes in the modern olympics is", median(atheletes$Age))
```

```
## The median age of the athletes in the modern olympics is 24
```

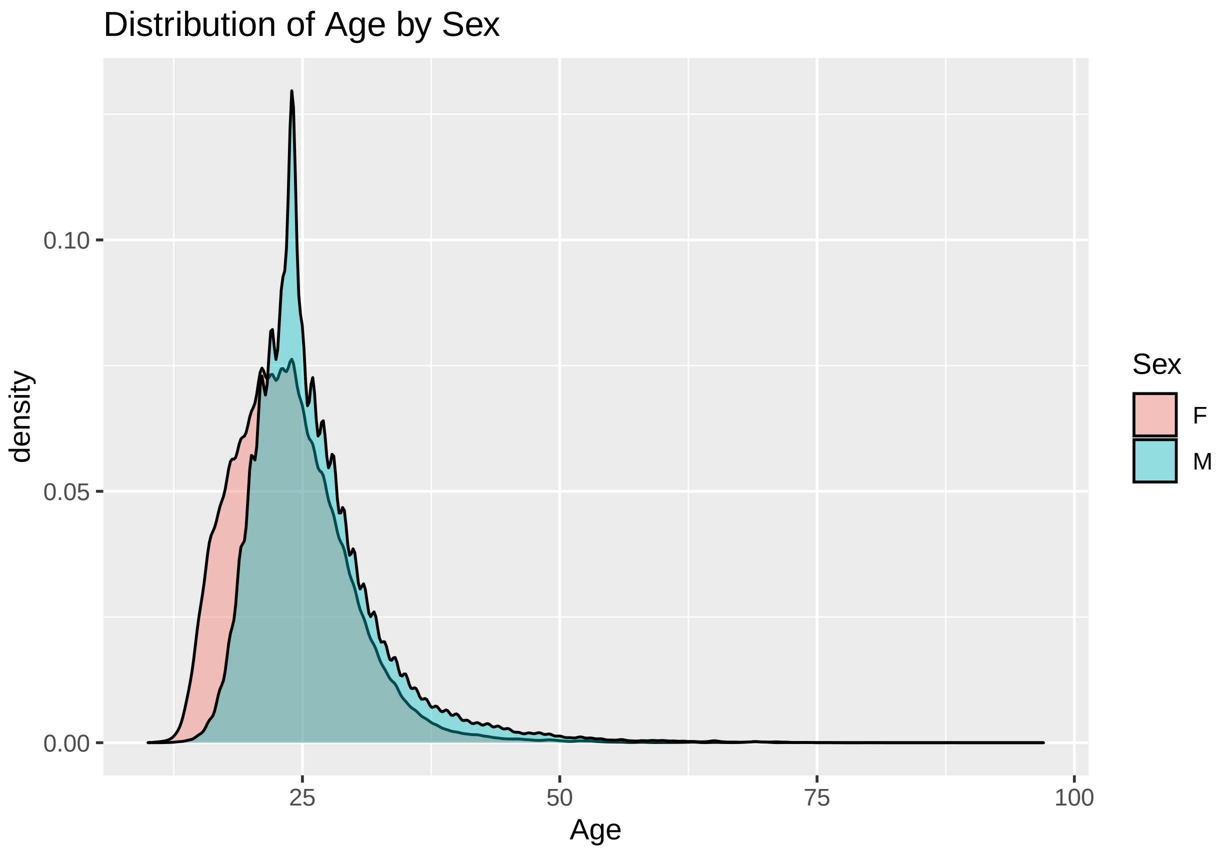
```
cat("\nThe median age of the male athletes in the modern olympics is", median(atheletes$Age[atheletes$S
```

```
##
## The median age of the male athletes in the modern olympics is 25
```

```
cat("\nThe median age of the female athletes in the modern olympics is", median(atheletes$Age[atheletes
```

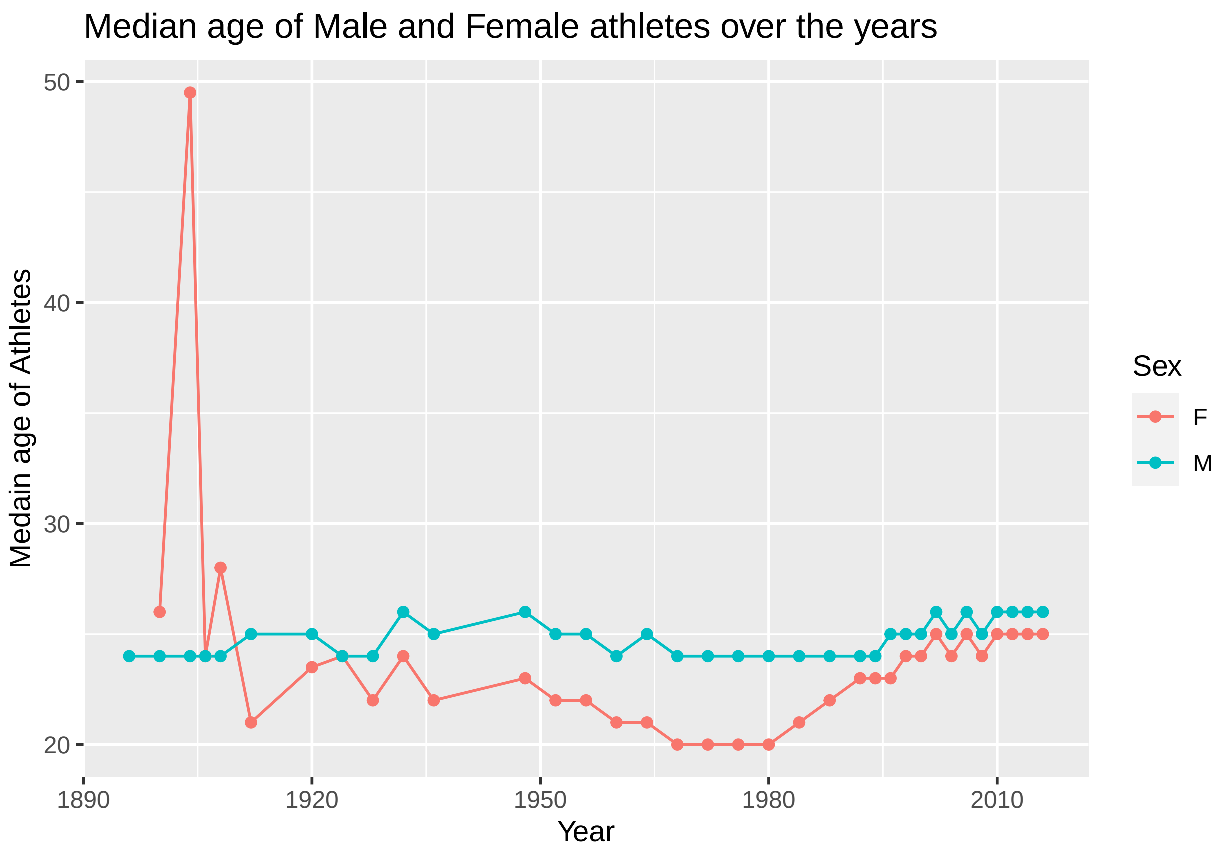
```
##
## The median age of the female athletes in the modern olympics is 23
```

```
atheletes %>%
      ggplot(aes(x=Age, fill=Sex)) +
      geom_density(alpha=0.4) +
      labs(x = "Age", title = "Distribution of Age by Sex")
```



```
atheletes %>%
  group_by(Year, Sex) %>%
  summarise(Median_Age = median(Age)) %>%
  ggplot(aes(x = Year, y = Median_Age, Group = Sex)) +
  geom_line(aes(color = Sex)) +
  geom_point(aes(color = Sex)) +
  labs( x = "Year", y = "Medain age of Athletes", title = "Median age of Male and Female athletes over t
```

```
## `summarise()` has grouped output by 'Year'. You can override using the `.groups` argument.
```



Team

```
cat("The total number of teams that have paricipated in the olympics are", length(unique(atheletes$Team
```

```
## The total number of teams that have paricipated in the olympics are 1184
```

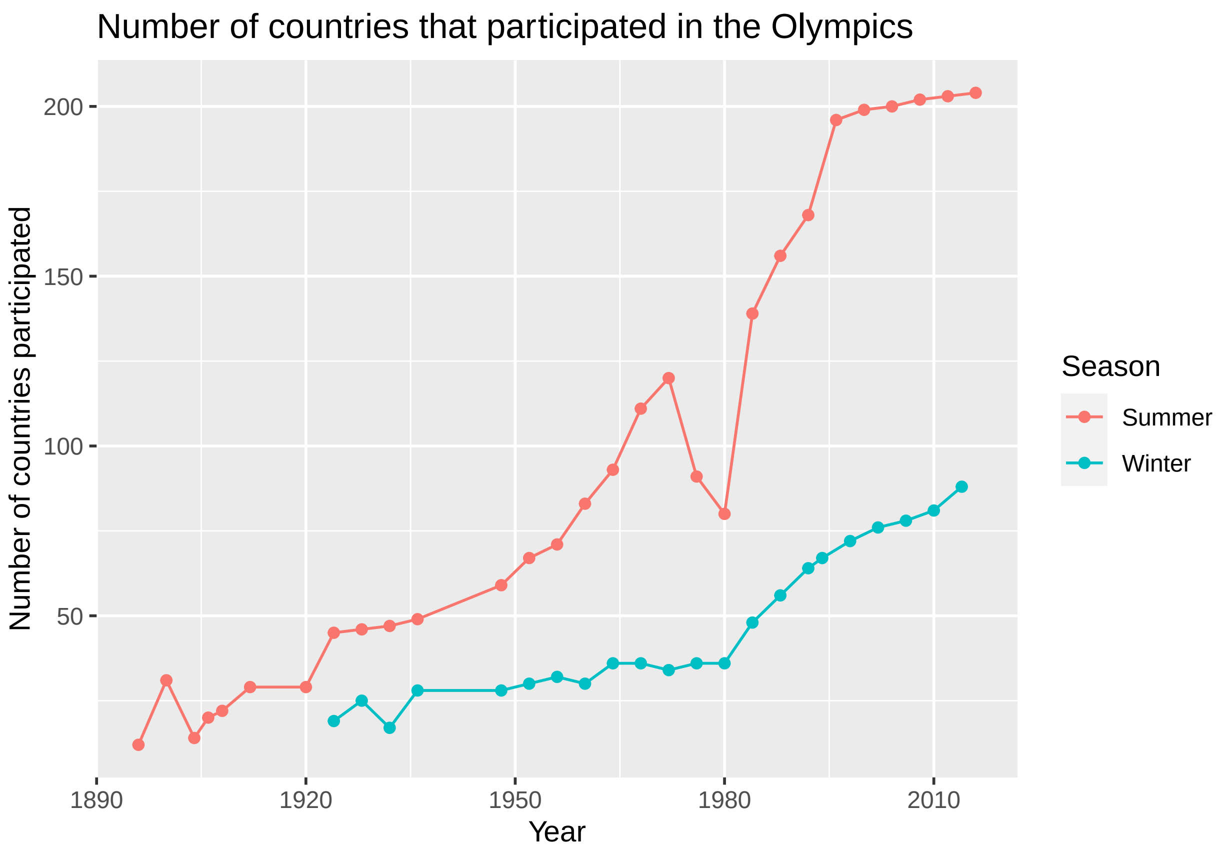
```
atheletes <- atheletes %>%
              left_join(regions, by = "NOC")
```

```
cat("The total number of National Olympics Committees that have paricipated in the olympics are", lengt
```

```
## The total number of National Olympics Committees that have paricipated in the olympics are 206
```

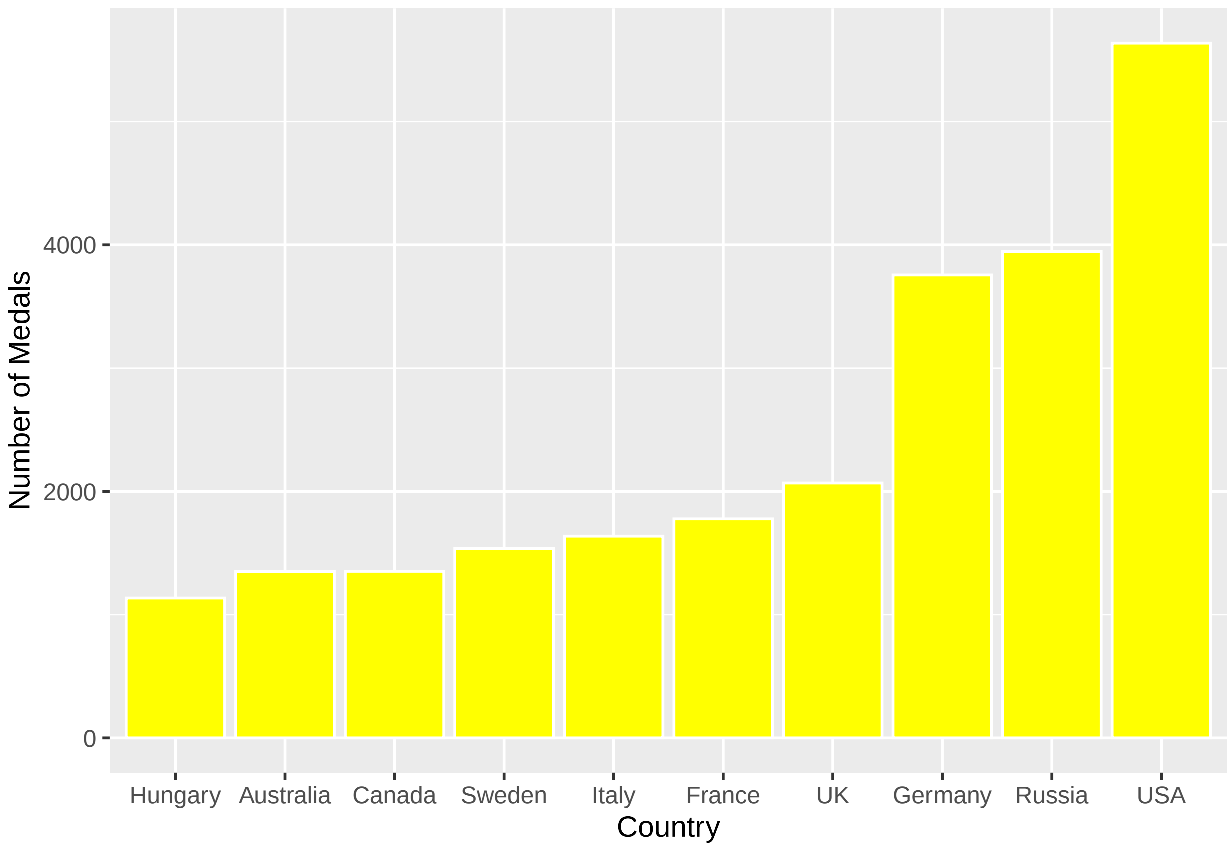
```
atheletes %>%
  group_by(Year, Season) %>%
  summarise(NoOfCountries = length(unique(region))) %>%
  ggplot(aes(x = Year, y = NoOfCountries, group = Season)) +
  geom_line(aes(color = Season)) +
  geom_point(aes(color = Season)) +
  labs(x = "Year", y = "Number of countries participated", title = "Number of countries that participat
```

```
## `summarise()` has grouped output by 'Year'. You can override using the `.groups` argument.
```



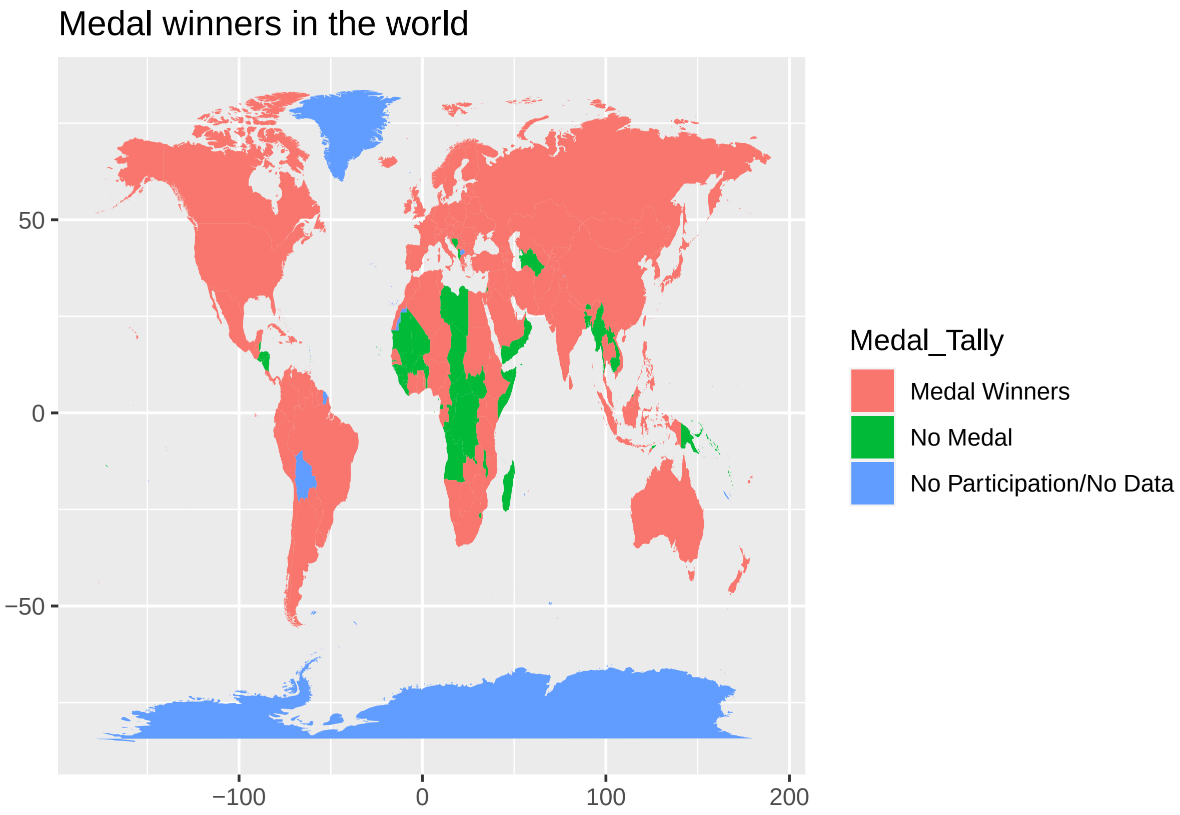
```
atheletes %>%
  filter(Medal != "<NA>") %>%
  group_by(region) %>%
  summarise(Medal_Tally = length(Medal))%>%
  arrange(desc(Medal_Tally)) %>%
  ungroup() %>%
  mutate(region = reorder(region,Medal_Tally)) %>%
  top_n(10) %>%
  ggplot(aes(x = region,y = Medal_Tally)) +
    geom_bar(stat='identity',colour="white", fill = "yellow") +
    labs(x = 'Country', y = 'Number of Medals')
```

```
## Selecting by Medal_Tally
```



```
Gold_Winners <- atheletes %>%
                     filter(Medal == "Gold") %>%
                     group_by(region) %>%
                     summarise(Medal_Tally = length(Medal)) %>%
                     arrange(desc(Medal_Tally)) %>%
                     mutate(region = str_trim(region),Medal_Tally = str_trim(Medal_Tally))

Silver_Winners <- atheletes %>%
                     filter(Medal == "Silver") %>%
                     group_by(region) %>%
                     summarise(Medal_Tally = length(Medal)) %>%
                     arrange(desc(Medal_Tally)) %>%
                     mutate(region = str_trim(region),Medal_Tally = str_trim(Medal_Tally))

Bronze_Winners <- atheletes %>%
                     filter(Medal == "Bronze") %>%
                     group_by(region) %>%
                     summarise(Medal_Tally = length(Medal)) %>%
                     arrange(desc(Medal_Tally)) %>%
                     mutate(region = str_trim(region),Medal_Tally = str_trim(Medal_Tally))

AllMedals <- atheletes %>%
                     filter(Medal != "<NA>") %>%
                     group_by(region) %>%
                     summarise(Medal_Tally = length(Medal)) %>%
                     arrange(desc(Medal_Tally)) %>%
```

```r
                     mutate(region = str_trim(region),Medal_Tally = str_trim(Medal_Tally))

All <- atheletes %>%
                      group_by(region) %>%
                     summarise(Medal_Tally = length(Medal)) %>%
                     arrange(desc(Medal_Tally)) %>%
                     mutate(region = str_trim(region),Medal_Tally = str_trim(Medal_Tally)) %>%
                     filter(!region %in% AllMedals$region) %>%
                     mutate(Medal_Tally = "No Medal")

AllMedals$Medal_Tally <- "Medal Winners"

Medal_Tally <- rbind(AllMedals, All)

map.world <- map_data("world")

map.world_joined <- left_join(map.world, Medal_Tally, by ='region')

map.world_joined$Medal_Tally[is.na(map.world_joined$Medal_Tally)] <- "No Participation/No Data"

ggplot() +
  geom_polygon(data = map.world_joined, aes(x = long, y = lat, group = group, fill = Medal_Tally)) +
  labs(x = " ", y = " ", title = 'Medal winners in the world')
```


Medal winners in the world
50
0
−50
−100
0
100
200
Medal_Tally
Medal Winners
No Medal
No Participation/No Data