21BDS0340

Abhinav Dinesh Srivatsa

Exploratory Data Analysis Lab

Practice Problem Set – 2

**Experiment 3**

Code:

library(dplyr)

library(tidyr)

library(magrittr)

setwd("/Users/abhi/College Work/Year 4 Semester 1 (Sem 7)/Exploratory Data Analysis Lab/Assignment 2")

data = read.csv("DS2\_Match.csv")

Output:

> setwd("/Users/abhi/College Work/Year 4 Semester 1 (Sem 7)/Exploratory Data Analysis Lab/Assignment 2")

> data = read.csv("DS2\_Match.csv")

> library(dplyr)

> library(tidyr)

> library(magrittr)

> setwd("/Users/abhi/College Work/Year 4 Semester 1 (Sem 7)/Exploratory Data Analysis Lab/Assignment 2")

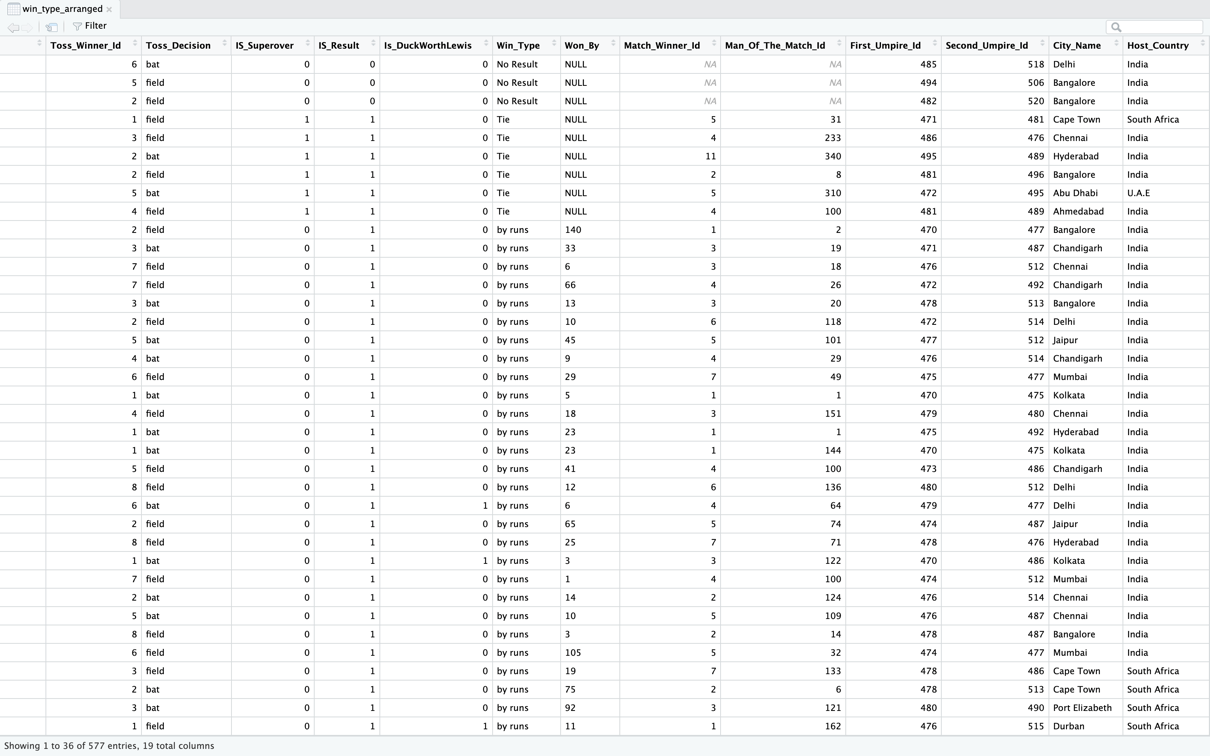
> data = read.csv("DS2\_Match.csv")

Code:

win\_type\_arranged = data %>% arrange(Win\_Type)

View(win\_type\_arranged)

Output:

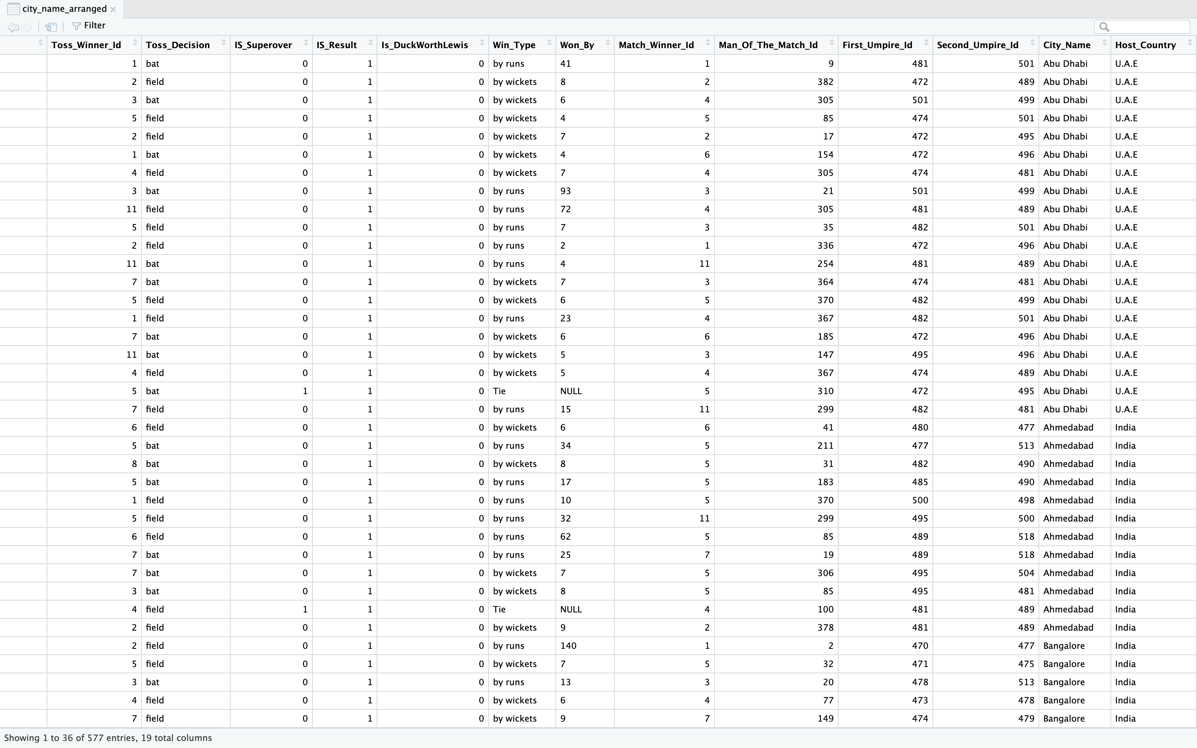


Code:

city\_name\_arranged = data %>% arrange(City\_Name)

View(city\_name\_arranged)

Output:

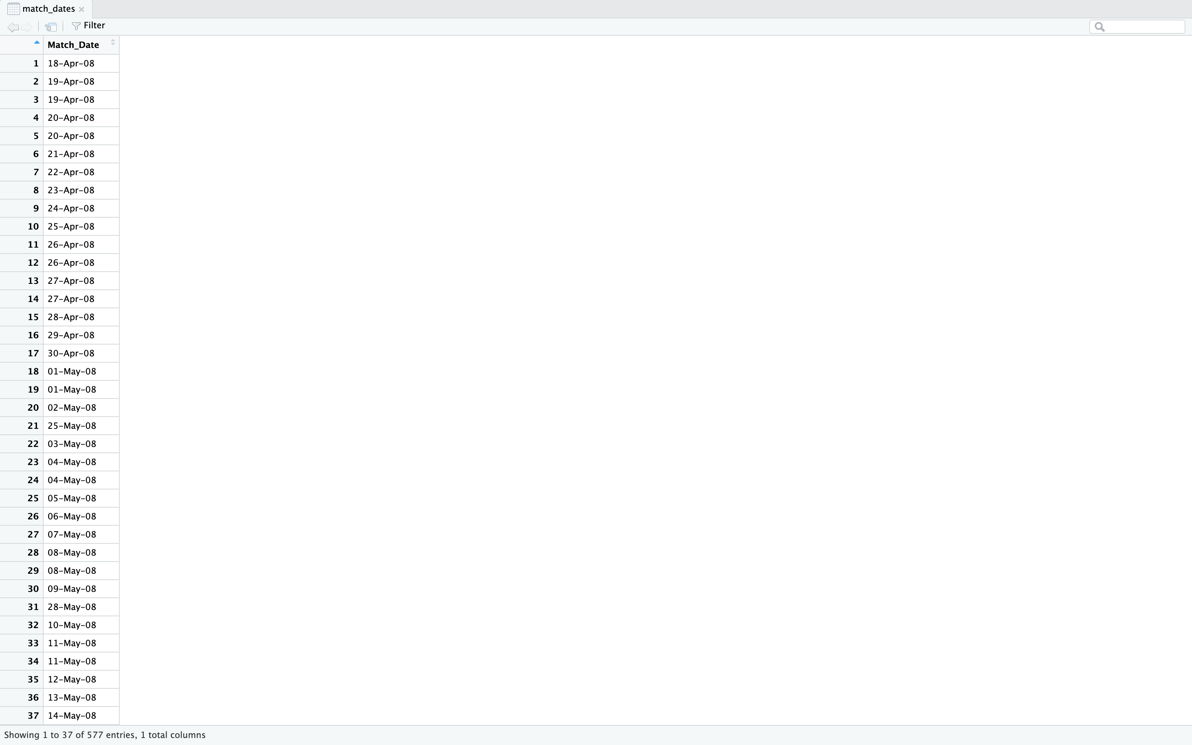


Code:

match\_dates = data %>% select(Match\_Date)

View(match\_dates)

Output:

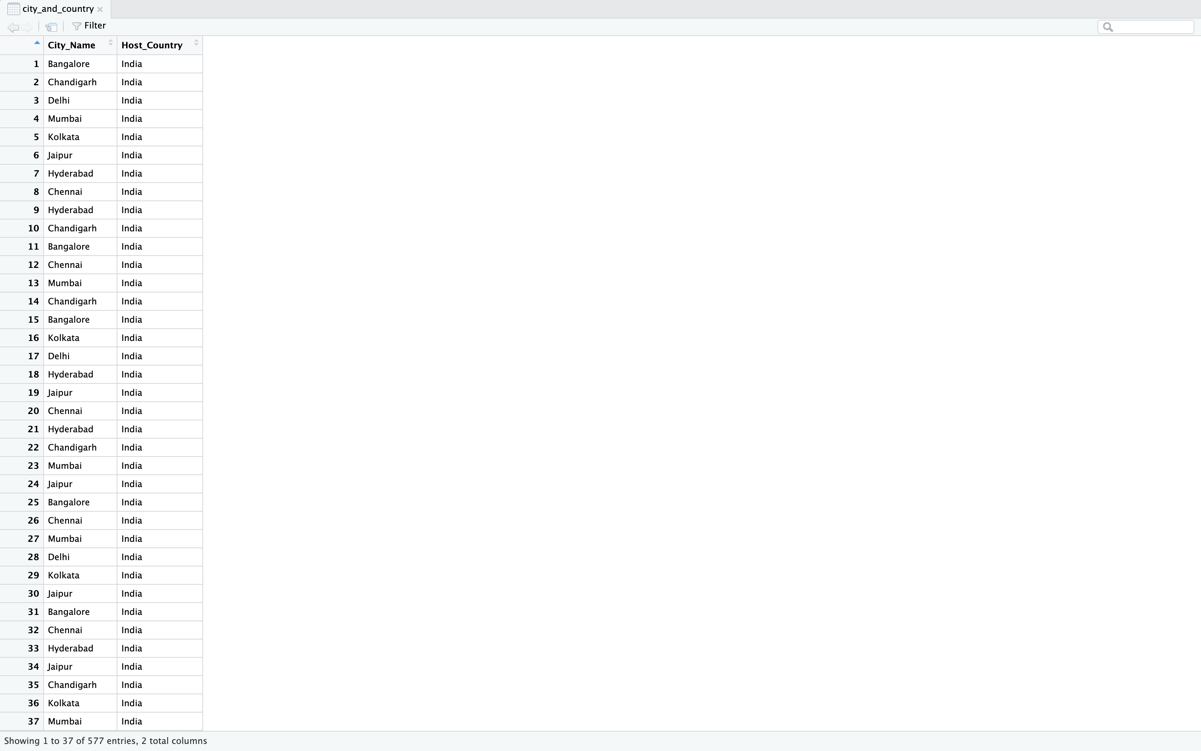


Code:

city\_and\_country = data %>% select(City\_Name, Host\_Country)

View(city\_and\_country)

Output:

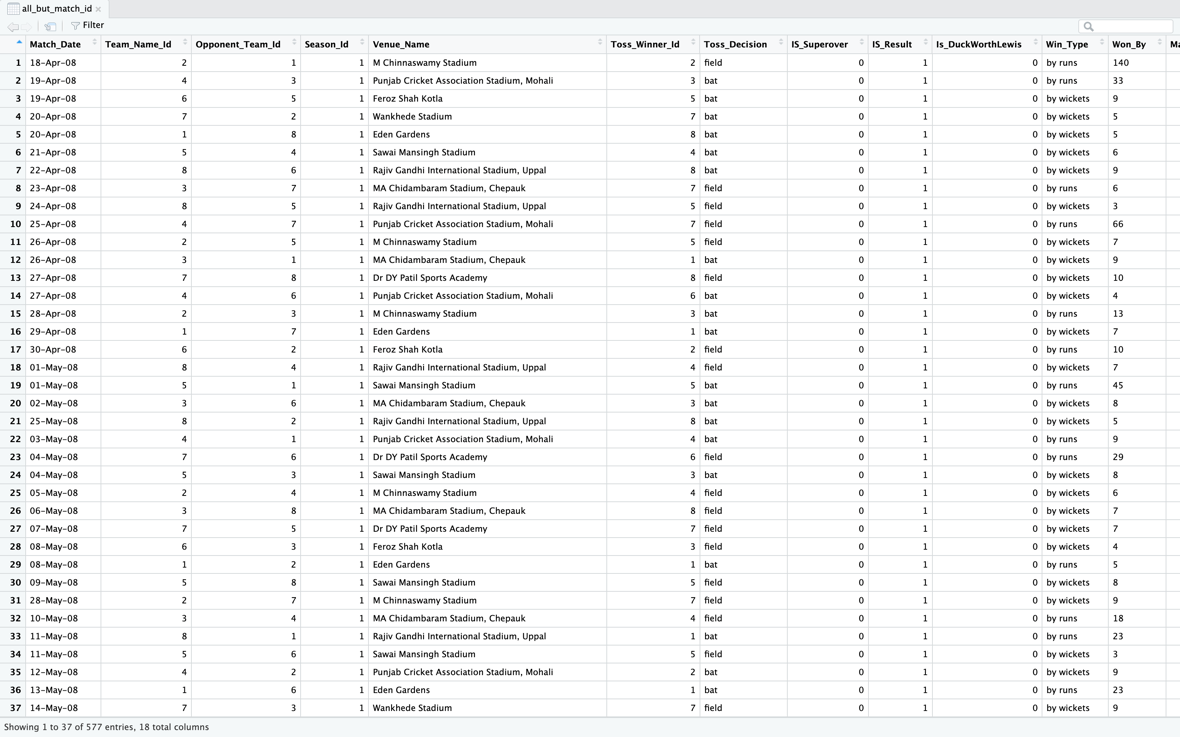


Code:

all\_but\_match\_id = data %>% select(-Match\_Id)

View(all\_but\_match\_id)

Output:

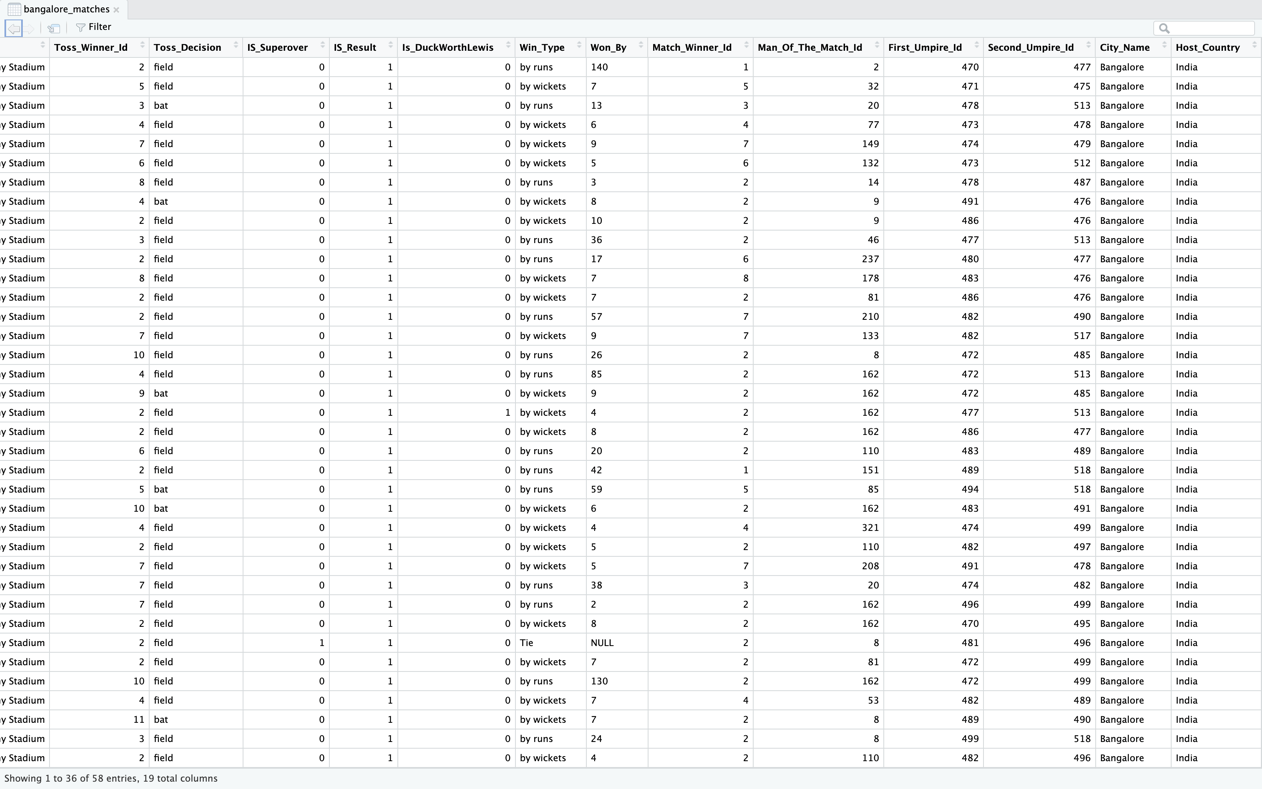


Code:

bangalore\_matches = data %>% filter(City\_Name == "Bangalore")

View(bangalore\_matches)

Output:

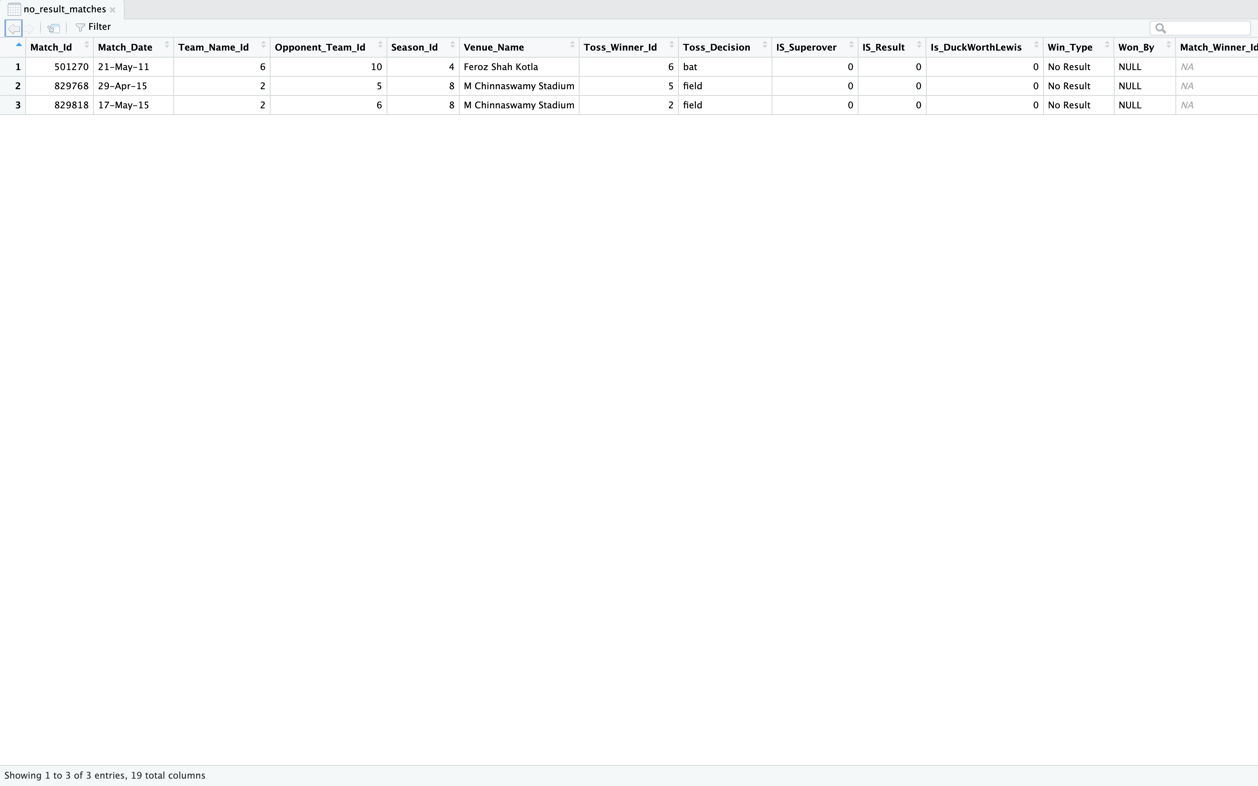


Code:

no\_result\_matches = data %>% filter(Win\_Type == "No Result")

View(no\_result\_matches)

Output:

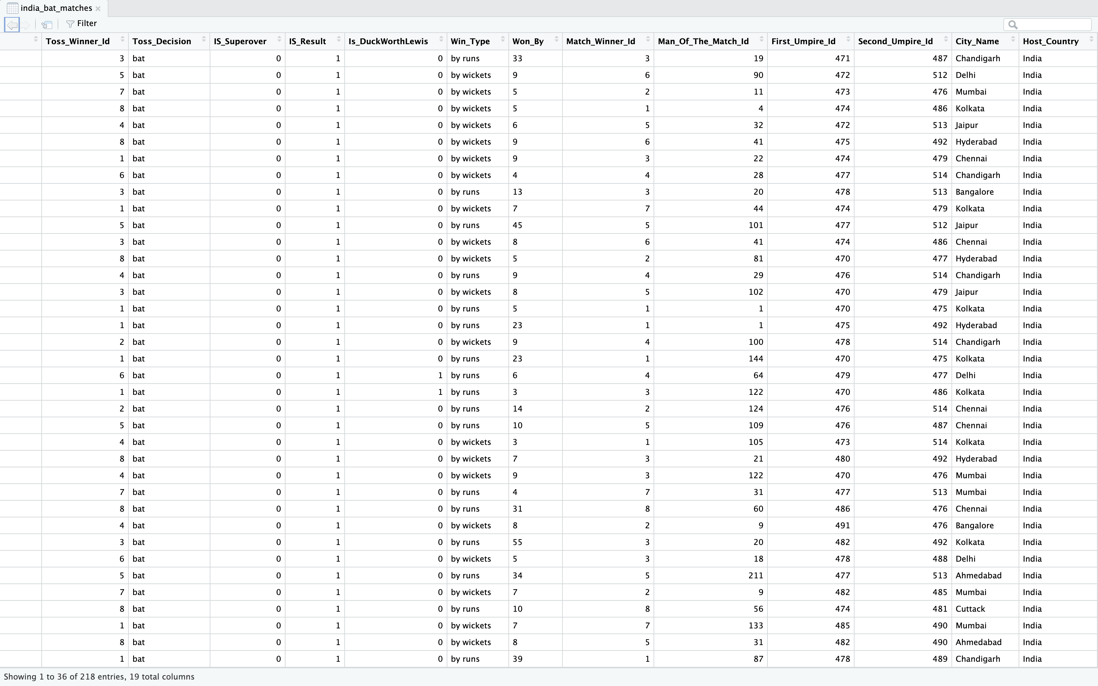


Code:

india\_bat\_matches = data %>% filter(Host\_Country == "India", Toss\_Decision == "bat")

View(india\_bat\_matches)

Output:



Code:

custom\_data = data.frame(

Name = c("Abhinav", "Tanush", "Sumathi"),

English = c(45, 67, 87),

Math = c(53, 65, 23)

)

gathered = custom\_data %>% gather("Course", "Marks", 2:3)

View(gathered)

Output:

A white rectangular object with a white border

Description automatically generated

Code:

spread = gathered %>% spread("Course", "Marks")

View(spread)

Output:

A white rectangular object with a white border

Description automatically generated

Code:

grouped = data %>% group\_by(Host\_Country) %>%

summarise(count = n())

View(grouped)

Output:

A white rectangular object with a white border

Description automatically generated

Code:

mutated = data %>% mutate(year=format(as.Date(data$Match\_Date, format="%d-%b-%y"), "%Y"))

View(mutated)

Output:

A screenshot of a computer

Description automatically generated

**Experiment 4**

Code:

library(dplyr)

library(ggplot2)

library(lubridate)

setwd("/Users/abhi/College Work/Year 4 Semester 1 (Sem 7)/Exploratory Data Analysis Lab/Assignment 2")

data = read.csv("./DS2\_Match.csv")

Output:

> library(dplyr)

> library(ggplot2)

> library(lubridate)

> setwd("/Users/abhi/College Work/Year 4 Semester 1 (Sem 7)/Exploratory Data Analysis Lab/Assignment 2")

> data = read.csv("./DS2\_Match.csv")

Code:

# viewing data

View(data)

Output:

A screenshot of a computer

Description automatically generated

Code:

# dimentions and names of columns

dim(data)

names(data)

Output:

> dim(data)

[1] 577 19

> names(data)

[1] "Match\_Id" "Match\_Date" "Team\_Name\_Id" "Opponent\_Team\_Id"

[5] "Season\_Id" "Venue\_Name" "Toss\_Winner\_Id" "Toss\_Decision"

[9] "IS\_Superover" "IS\_Result" "Is\_DuckWorthLewis" "Win\_Type"

[13] "Won\_By" "Match\_Winner\_Id" "Man\_Of\_The\_Match\_Id" "First\_Umpire\_Id"

[17] "Second\_Umpire\_Id" "City\_Name" "Host\_Country"

Code:

# sorting data by win type

head(sort(data$Win\_Type))

Output:

> head(sort(data$Win\_Type))

[1] "by runs" "by runs" "by runs" "by runs" "by runs" "by runs"

Code:

# summary of data

summary(data)

Output:

> summary(data)

Match\_Id Match\_Date Team\_Name\_Id Opponent\_Team\_Id Season\_Id

Min. :335987 Length:577 Min. : 1.000 Min. : 1.000 Min. :1.000

1st Qu.:419140 Class :character 1st Qu.: 3.000 1st Qu.: 3.000 1st Qu.:3.000

Median :548353 Mode :character Median : 5.000 Median : 5.000 Median :5.000

Mean :591636 Mean : 5.102 Mean : 5.211 Mean :5.029

3rd Qu.:734004 3rd Qu.: 7.000 3rd Qu.: 7.000 3rd Qu.:7.000

Max. :981024 Max. :13.000 Max. :13.000 Max. :9.000

Venue\_Name Toss\_Winner\_Id Toss\_Decision IS\_Superover IS\_Result

Length:577 Min. : 1.000 Length:577 Min. :0.0000 Min. :0.0000

Class :character 1st Qu.: 3.000 Class :character 1st Qu.:0.0000 1st Qu.:1.0000

Mode :character Median : 5.000 Mode :character Median :0.0000 Median :1.0000

Mean : 5.192 Mean :0.0104 Mean :0.9948

3rd Qu.: 7.000 3rd Qu.:0.0000 3rd Qu.:1.0000

Max. :13.000 Max. :1.0000 Max. :1.0000

Is\_DuckWorthLewis Win\_Type Won\_By Match\_Winner\_Id

Min. :0.000 Length:577 Length:577 Min. : 1.000

1st Qu.:0.000 Class :character Class :character 1st Qu.: 3.000

Median :0.000 Mode :character Mode :character Median : 5.000

Mean :0.026 Mean : 4.991

3rd Qu.:0.000 3rd Qu.: 7.000

Max. :1.000 Max. :13.000

NA's :3

Man\_Of\_The\_Match\_Id First\_Umpire\_Id Second\_Umpire\_Id City\_Name Host\_Country

Min. : 1.0 Min. :470.0 Min. :471.0 Length:577 Length:577

1st Qu.: 40.0 1st Qu.:475.0 1st Qu.:488.0 Class :character Class :character

Median :105.5 Median :482.0 Median :493.0 Mode :character Mode :character

Mean :139.8 Mean :484.1 Mean :495.2

3rd Qu.:209.5 3rd Qu.:493.0 3rd Qu.:500.0

Max. :460.0 Max. :511.0 Max. :521.0

NA's :3

Code:

# finding min and max of first umpire id

min(data$First\_Umpire\_Id)

max(data$First\_Umpire\_Id)

Output:

> min(data$First\_Umpire\_Id)

[1] 470

> max(data$First\_Umpire\_Id)

[1] 511

Code:

# finding mean and median of won by amount

data$Won\_By = sapply(data$Won\_By, function(x) {

if (x == "NULL") {

return(0)

}

x

})

data$Won\_By = as.numeric(data$Won\_By)

mean(data$Won\_By)

median(data$Won\_By)

Output:

> mean(data$Won\_By)

[1] 17.07972

> median(data$Won\_By)

[1] 8

Code:

# finding quantiles of won by

quantile(data$Won\_By)

Output:

> quantile(data$Won\_By)

0% 25% 50% 75% 100%

0 6 8 20 144

Code:

# checking NaN values (if cleanup is required)

sum(apply(data, 2, is.nan))

Output:

[1] 0

Code:

# check different host countries

levels(factor(data$Host\_Country))

Output:

[1] "India" "South Africa" "U.A.E"

Code:

# plotting by host country

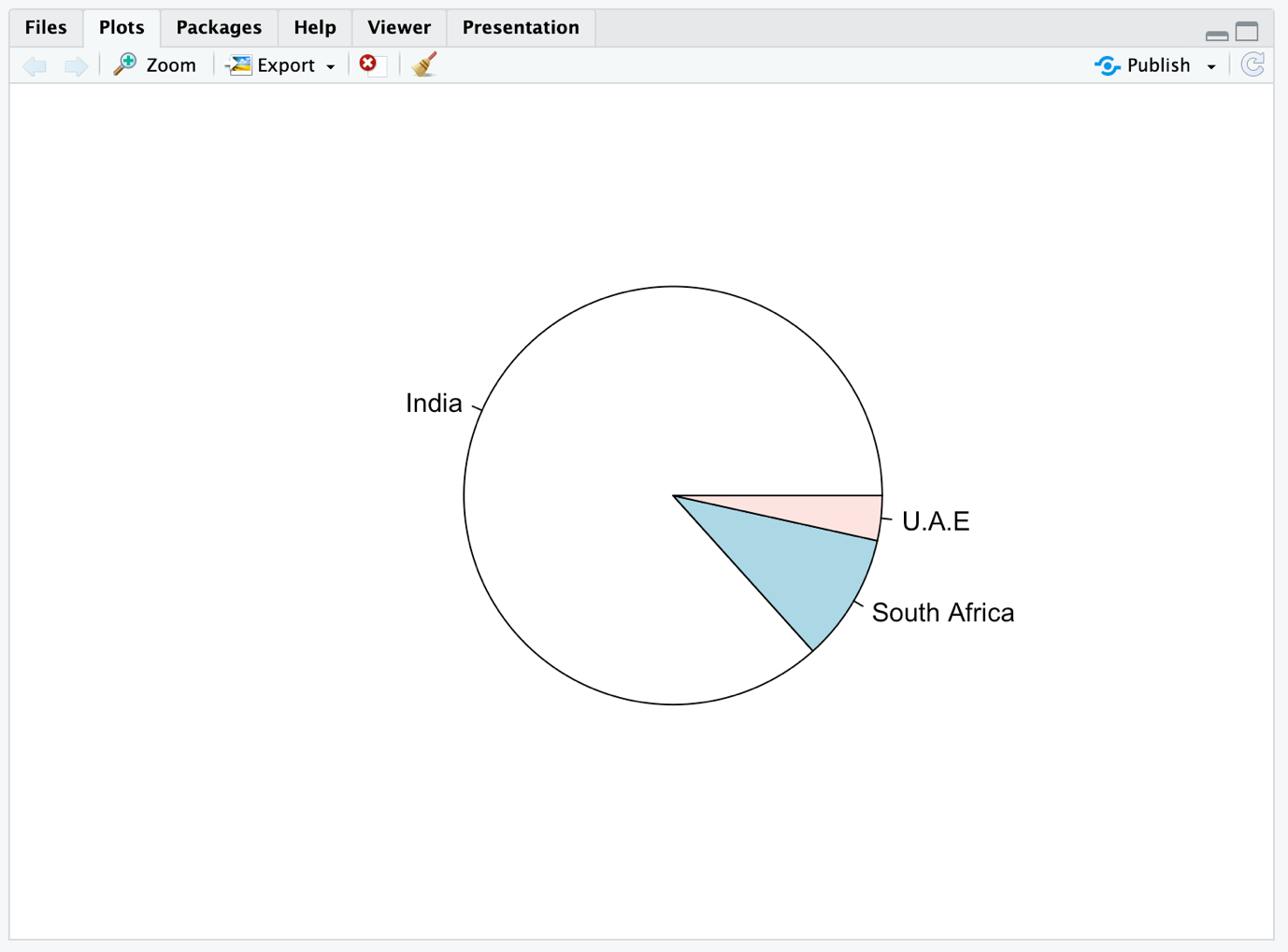
country\_counts = data %>%

group\_by(Host\_Country) %>%

summarise(count = length(Host\_Country))

pie(country\_counts$count, labels=country\_counts$Host\_Country)

Output:



Code:

# check different win conditions

levels(factor(data$Win\_Type))

Output:

[1] "by runs" "by wickets" "No Result" "Tie"

Code:

# plotting by win condition

win\_cond\_count = data %>%

group\_by(Win\_Type) %>%

summarise(count = length(Win\_Type))

barplot(win\_cond\_count$count, xlab="Win Condition", ylab="Count", names.arg=win\_cond\_count$Win\_Type, col="cyan")

Output:

A screenshot of a computer

Description automatically generated

Code:

# check different city names

levels(factor(data$City\_Name))

Output:

[1] "Abu Dhabi" "Ahmedabad" "Bangalore" "Bloemfontein" "Cape Town"

[6] "Centurion" "Chandigarh" "Chennai" "Cuttack" "Delhi"

[11] "Dharamsala" "Durban" "East London" "Hyderabad" "Indore"

[16] "Jaipur" "Johannesburg" "Kanpur" "Kimberley" "Kochi"

[21] "Kolkata" "Mumbai" "Nagpur" "Port Elizabeth" "Pune"

[26] "Raipur" "Rajkot" "Ranchi" "Visakhapatnam"

Code:

# plotting by city name, color by host country

city = data %>%

group\_by(City\_Name, Host\_Country) %>%

summarise(count = length(City\_Name))

ggplot(city, aes(x=City\_Name, y=count, fill=Host\_Country)) +

geom\_bar(stat="identity") +

geom\_text(aes(label=City\_Name), vjust=0.5, angle=90, hjust=0) + scale\_x\_discrete(labels=NULL) +

ylim(0, 100) +

labs(x="City", y="Matches")

Output:

A screenshot of a graph

Description automatically generated

Code:

# casting match date column to date type

match\_dates = data.frame(date=as.Date(data$Match\_Date, format="%d-%b-%y"))

head(match\_dates, 2)

Output:

date

1 2008-04-18

2 2008-04-19

Code:

# finding the matches played per month

match\_dates = match\_dates %>%

mutate(month=month(date)) %>%

group\_by(month) %>%

summarise(count=length(month))

head(match\_dates)

Output:

# A tibble: 4 × 2

month count

<dbl> <int>

1 3 29

2 4 261

3 5 285

4 6 2

Code:

# plotting matches played by month

ggplot(match\_dates, aes(x=month, y=count, fill=month)) +

geom\_bar(stat="identity") +

geom\_text(aes(label=month.name[month]), vjust=-0.5) +

labs(x="Month", y="Matches Played")

Output:

A graph on a screen

Description automatically generated