**#Reading ';' delimited file into R**

Camera=read.table(file.choose(),sep=";",header=TRUE)

**#sorting**

Camera[1:10, ]

attach(Camera)

sort1.Camera <- Camera[order(Price) , ]

head(sort1.Camera)

sort2.Camera <- Camera[order(Release\_date,Price) , ]

head(sort2.Camera)

sort3.Camera <- Camera[order(Release\_date,-Price) , ]

head(sort3.Camera)

Camera$Price[2:5] <- NA

sort4.Camera <- Camera[order(Camera$Price, na.last=FALSE) , ]

head(sort4.Camera)

sort5.Camera <- Camera[order(Camera$Price, na.last=TRUE) , ]

tail(sort5.Camera)

detach(Camera)

**#Transpose**

cars <- mtcars[1:5,1:4]

cars

t(cars)

**#Aggregate Data in R**

****

data=read.xlsx(file.choose(),1,stringsAsFactors=FALSE)

**Method1**

agg\_mean <- aggregate(data[,5:6],by=list(data$Role),FUN=mean, na.rm=TRUE)

agg\_mean1 <- aggregate(data[,5:6],by=list(data$Role,data$Shift),FUN=mean, na.rm=TRUE)

agg\_mean2 <- aggregate(data[,5:6],by=list(data$Role,data$Shift,data$Gender),FUN=mean, na.rm=TRUE)

agg\_sum <- aggregate(data[,5:6],by=list(data$Role,data$Shift,data$Gender),FUN=sum, na.rm=TRUE)

**Method2**

# by() function in R with mean

 by(data$Salary,list(data$Role),mean)

 by(data$Salary,list(data$Role),sum)

# by() function in R with more than one list

 by(data$Salary,list(data$Role,data$Shift),mean)

 by(data$Salary,list(data$Role,data$Shift),sum)

**#Subset Function**

Syntax of Subset Function in R:

subset(x, condition,select)

# subset() function in R

newdata<-subset(mtcars,mpg>=30)

newdata

# subset() function in R with select specific columns

newdata<-subset(mtcars,mpg>=30, select=c(mpg,cyl,gear))

newdata

# subset() function in R

newdata<-subset(mtcars,mpg>=30, select=mpg:drat)

newdata

#taking hotel examples again

subset1<-subset(data, Role=="Chef")

subset1

subset2<-subset(data, Role=="Chef" & Shift=="Breakfast")

subset2

subset3<-subset(data, Role=="Chef" | Shift=="Breakfast",select=c(Salary,Age))

subset3

#Keeping Variables

keep <- data[, c(1, 3, 5)]

keep

select\_col=data[c("Role","Salary")]

select\_col

#Dropping Variables

df = subset(data, select = -c(Role,Shift) )

df

#Selecting observation

newdata <- data[1:2,]

newdata

newdata <- data[c(1,3,5),]

newdata

newdata <- data[1:2,2:3]

newdata

newdata <- data[c(1,3,5),3:5]

newdata

**Merging Data**

# Syntax merge two data frames by ID  
total <- merge(data frameA, data frameB, by="ID")

master=read.csv(file.choose())

slave=read.csv(file.choose())

Merge\_data <- merge(master, slave, by="Order\_ID")

head(Merge\_data)

#Join by more than 2 or more key variables

# merge two data frames by ID and Country

total <- merge(data frameA,data frameB,by=c("ID","Country"))

**Cbind Function**

df1 = data.frame(name = c("Rahul","joe","Adam","Brendon"), married\_year = c(2016,2015,2016,2008))

df2 = data.frame(Birth\_place = c("Delhi","Seattle","London","Moscow"), Birth\_year = c(1988,1990,1989,1984))

df1

df2

cbinded\_df<-cbind(df1,df2)

cbinded\_df

**Rbind Function**

df1 = data.frame(CustomerId = c(1:6), Product = c(rep("Oven", 3), rep("Television", 3)))

df2 = data.frame(CustomerId = c(4:7), Product = c(rep("Television", 2), rep("Air conditioner", 2)))

df1

df2

rbind(df1,df2)

**Substring Function**

## Extracting values with substring in R

substring("HumptyDumpty sat on a wall",5,9)

## Replacing values with substring in R

mystring<-"Humpty\_Dumpty sat on a wall"

substring(mystring,7,7)=" "

mystring

**Table Function in R – Frequency table in R & cross table in R**

## Frequency table with table() function in R

table(iris$Species)

## Frequency table with condition using table function in R

table(iris$Sepal.Length>5.0)

**#** **2 way cross table in R**

## cross tabulation gear \* carb

table(mtcars$gear,mtcars$carb)

**3 way cross table in R:**

## 3 way cross tabulation gear \* carb\* cyl with table function in R

table(mtcars$gear,mtcars$carb,mtcars$cyl)

**SET Operators:**

**union in R- union of data frames :**

# Create two data frames

df1 = data.frame(CustomerId = c(1:6), Product = c(rep("Oven", 3), rep("Television", 3)))

df2 = data.frame(CustomerId = c(4:7), Product = c(rep("Television", 2), rep("Air conditioner", 2)))

df1

df2

# union in R - union of data frames in R

df\_union1<-merge(df1,df2,all=TRUE)

df\_union1

# intersection of data frames in R

df\_intersect<-merge(df1,df2,all=FALSE)

df\_intersect

**SQL - Set Operations and Joins (sqldf) package**

**Set Operations**

**UNION**

doctors <- data.frame(first=c('A','B','C'), last=c('W','X','Y'), age=c(20,30,40))

teachers <- data.frame(first=c('A','B','D'), last=c('W','X','Z'), age=c(20,30,50))

doctors

teachers

**# Union - doctors or teachers**

sqldf("SELECT first, last FROM doctors UNION

SELECT first, last FROM teachers")

**# Intersect - doctors and teachers**

sqldf("SELECT first, last FROM doctors INTERSECT

SELECT first, last FROM teachers")

**# Except - doctors who are not teachers**

sqldf("SELECT first, last FROM doctors EXCEPT

SELECT first, last FROM teachers")

**Joins/Merge**

customers <- data.frame(cust\_id=c(1,2,3), name=c('A','B','C'), age=c(20,30,50))

orders <- data.frame(order\_id=c(1,2,3), cust\_id=c(1,2,4), order\_qty=c(4,7,8))

customers

orders

**# Inner join**

sqldf("SELECT \* FROM orders o INNER JOIN customers c ON o.cust\_id=c.cust\_id")

**# Left join**

sqldf("SELECT \* FROM orders o LEFT JOIN customers c ON o.cust\_id=c.cust\_id")

**# Right join (simulate, because SQLITE doesn't support)**

sqldf("SELECT \* FROM customers c LEFT JOIN orders o ON o.cust\_id=c.cust\_id")

**# Full join (simulate, because SQLITE doesn't support)**

sqldf(

"SELECT order\_id, order\_qty, o.cust\_id, name, age FROM orders o LEFT JOIN customers c ON o.cust\_id=c.cust\_id

UNION

SELECT order\_id, order\_qty, c.cust\_id, name, age FROM customers c LEFT JOIN orders o ON o.cust\_id=c.cust\_id"

)

**Basic Queries in SQL:**

a=sqldf("select \* from mtcars")

a

a=sqldf("select \* from mtcars limit 5")

a

a=sqldf("select mpg from mtcars")

a

a=sqldf("select mpg,hp from mtcars")

a

a=sqldf("select avg(mpg) as Miles\_per\_Gallon from mtcars")

a

a=sqldf("select sum(mpg) as Miles\_per\_Gallon from mtcars")

a

a=sqldf("select \* from data where Role='Chef'")

a

a=sqldf("select Role,avg(Salary) as Mean\_Salary, avg(Age) as Mean\_Age from data group by Role order by Role")

a

a=sqldf("select Role,avg(Salary) as Mean\_Salary, avg(Age) as Mean\_Age from data group by Role having avg(Age) > 50 order by Role")

a