**Assume that name of our data set is ‘who’ whose breakup is as follows:**

*'data.frame': 194 obs. of 13 variables:*

*$ Country : Factor w/ 194 levels "Afghanistan",..: 1 2 3 4 5 6 7 8 9 10 ...*

*$ Region : Factor w/ 6 levels "Africa","Americas",..: 3 4 1 4 1 2 2 4 6 4 ...*

*$ Population : int 29825 3162 38482 78 20821 89 41087 2969 23050 8464 ...*

*$ Under15 : num 47.4 21.3 27.4 15.2 47.6 ...*

*$ Over60 : num 3.82 14.93 7.17 22.86 3.84 ...*

*$ FertilityRate : num 5.4 1.75 2.83 NA 6.1 2.12 2.2 1.74 1.89 1.44 ...*

*$ LifeExpectancy : int 60 74 73 82 51 75 76 71 82 81 ...*

*$ ChildMortality : num 98.5 16.7 20 3.2 163.5 ...*

*$ CellularSubscribers : num 54.3 96.4 99 75.5 48.4 ...*

*$ LiteracyRate : num NA NA NA NA 70.1 99 97.8 99.6 NA NA ...*

*$ GNI : num 1140 8820 8310 NA 5230 ...*

*$ PrimarySchoolEnrollmentMale : num NA NA 98.2 78.4 93.1 91.1 NA NA 96.9 NA ...*

*$ PrimarySchoolEnrollmentFemale: num NA NA 96.4 79.4 78.2 84.5 NA NA 97.5 NA ...*

1. Vector - A vector is a series of numbers or characters stored in as the same object. Can be created using c function.
2. getwd() - to check the directory
3. setwd() - to change the directory
4. read.table() - to read a table or data set
5. read.csv() - to upload a csv data set into the R console
6. ls() -list all the variables that have been created in the console
7. Ctrl+l - clear the console
8. C (lower case) - to create a vector manually, called combine.
9. Seq - to create a sequence; for instance seq(0,100,2) would generate a sequence of numbers with an interval of 2 starting from zero and lasting up to 100.
10. To add a new variable to the data frame just add the name of new variable that you want to create with the $ sign and equal it to new vector with the help of combine function. For example - CountryData is a data frame consisting 2 variables (Country name and Life expectancy) and 5 observations. Now suppose you want to add a new variable population to this data frame, following is what you are required to do:
    1. CountryData$Population = c(100000,150000,200000,50000,316000)
    2. Now check str(CountryData) and it should give you three variables and 5 observations.
11. rbind - to combine two separate data frames
12. summary() - to see a brief statistical summary of your data
13. subset() - to select a particular variable from the data set; x= subset(who, Region == “Europe”)
14. write.csv(“Data Frame”, “filename.csv”) - this function is very useful if you want to generate a csv file for some subset of a large amount of data.
15. rm() - to remove a data frame from your console to free up some of the memory. This is particularly helpful if working with the large amount of data.
16. mean() - to calculate the mean
17. sd() - to calculate the standard deviation
18. which.min() - to calculate the minimum value ; for example which.min(who $ Under15 ) will give the row number that contains minimum value of variable Under15 under the data set ‘who’. To see the exact minimum value one more operation needs to perform. The []. However, this function is useful to see what other variable corresponds to this minimum value.
19. Which.max () - likewise
20. Plot (who$GNI, who$FertilityRate) - this will plot a scatter plot between two variables GNI and fertility rate of dataset ‘who’.
21. nrow() - returns the number of rows in a data set.
22. ncol() - returns the number of colums in a data set. In most cases, nrow and ncol are used to observe a subset among a large dataset. For instance, how many many countries are there that have GNI.10,000 & FertilityRate >2.5.
23. hist() - to create a histogram between the variables of a dataset
24. boxplot(who$GNI ~ who$LifeExpectancy) - to create the box plot between the variables
25. table() - By using the table() function on two variables, we can tell how they are related.is a nice function for a variable with less values; mostly it is good for string variables of factors with fewer levels.
26. tapply() - is a very versawhichtile function for numerical variables that is used to do multiple types of data manipulation. For instance, what is the minimum literacy rate in each region? A tapply function for the same would be as follows and render results as you can see along with it.

> tapply(who$LiteracyRate, who$Region, min, na.rm = TRUE)

Africa Americas Eastern Mediterranean

31.1 75.2 63.9

Europe South-East Asia Western Pacific

95.2 56.8 60.6