1) For Practice session refer to [1] and [2]

   # Load the package.
   library(tseries)
   data(AirPassengers)

   # How the data looks like
   AirPassengers

   # To find the class type of the data
   class(AirPassengers)
   [1] "ts"

   summary(AirPassengers)

   # Lets plot this data
   plot(AirPassengers)

   AP<-AirPassengers

   # This is the start of the time series
   start(AirPassengers)

   # This is the end of the time series
   end(AirPassengers)

   # Check the frequency of the time series
   frequency(AP)
   # The cycle of this time series is 12 months in a year

   # This will fit in a line
   abline(reg=lm(AirPassengers~time(AirPassengers)))

   # How to check trend: This will aggregate the cycles and display a year on year trend
   plot(aggregate(AirPassengers,FUN=mean))

   # Additive time series
```r
plot(decompose(AP))

# Multiplicative time series
ap.decom=decompose(AP,type="multi")
plot(ap.decom)

# Box plot across months will give us a sense on seasonal effect
boxplot(AirPassengers~cycle(AirPassengers))

# Dickey-Fueller test to check the stationarity of the series
adf.test(diff(log(AirPassengers)), alternative="stationary", k=0)
# ?diff, will tell you that you can put a lag value. Here by default the lag is 1 and thus difference is 1

# 2 things need to be done
# 1) We need to remove unequal variance. We do this using log of the series.
plot(log(AirPassengers))

# 2) We need to fix mean (trend). We do this by differentiating
plot(diff(log(AirPassengers)))

# AR(Autoregressive) I (Integration) MA (Moving Average) : p
# Autoregression: You can use the past values for predicting future : d
# Moving average: take Different intervals and do an average: q (coefficient of MA)

# acf (auto correlation graph)
# NOTE: The blue line shows significantly different values than zero.
acf(AirPassengers)

# Let's see how ACF and PACF curve come out after regressing on the difference.
acf(diff(log(AirPassengers))) # Determines the value of q
# The line number starts from 0
# The line number preceding the line which gets inverted
pacf(diff(log(AirPassengers))) # Determines the value of p
# The line number preceding the line which gets inverted

# d is the number of times you have done differentiation for making mean constant
# In this case, we have done only one time thus, value of d is 1.
fit <- arima(log(AirPassengers), c(0, 1, 1),seasonal = list(order = c(0, 1, 1), period = 12))

# Predicting the values for next 10 years
pred <- predict(fit, n.ahead = 10*12)

# Convert the log values into decimal format
# 2.718 is basically e, thus, we are changing the values back to decimal from logarithmic
# pred is a list with two items: pred and se. ( prediction and standard error).
```

pred1<-2.718^pred$pred

#check the predictions
pred1

#Lets plot it
ts.plot(AirPassengers,2.718^pred$pred, log = "y", lty = c(1,3))

#Testing our model now
datawide <- ts(AirPassengers,frequency=12,start=c(1949,1),end=(1959,12))

#datawide check
ndatawide

fit <- arima(log(datawide), c(0, 1, 1),seasonal = list(order = c(0, 1, 1), period = 12))

#Lets us take the top row only, i.e, 1960 only
data1 <- head(pred1,12)

#Predicted values
pred_1960 <- round(data1,digits=0)
pred_1960

#Original values
orig_1960 <- tail(AirPassengers,12)
orig_1960

#To see the predictions, use this command: print(pred$pred)
ts.plot(AirPassengers,2.718^pred$pred, log = "y", lty = c(1,3))

2) Very good tutorial about stationarity [3]

References
[1] https://www.youtube.com/watch?v=wNB8AgZPFLU&t=115s
[3] https://www.youtube.com/watch?v=K3eUIDu4NWw

Some other links.
http://rstudio-pubs-static.s3.amazonaws.com/311446_08b00d63cc794e158b1f4763eb70d43a.html
http://rstudio-pubs-static.s3.amazonaws.com/147569_0edc74a8769d4bb693ecb0e69faeca77.html
https://rpubs.com/emb90/137525