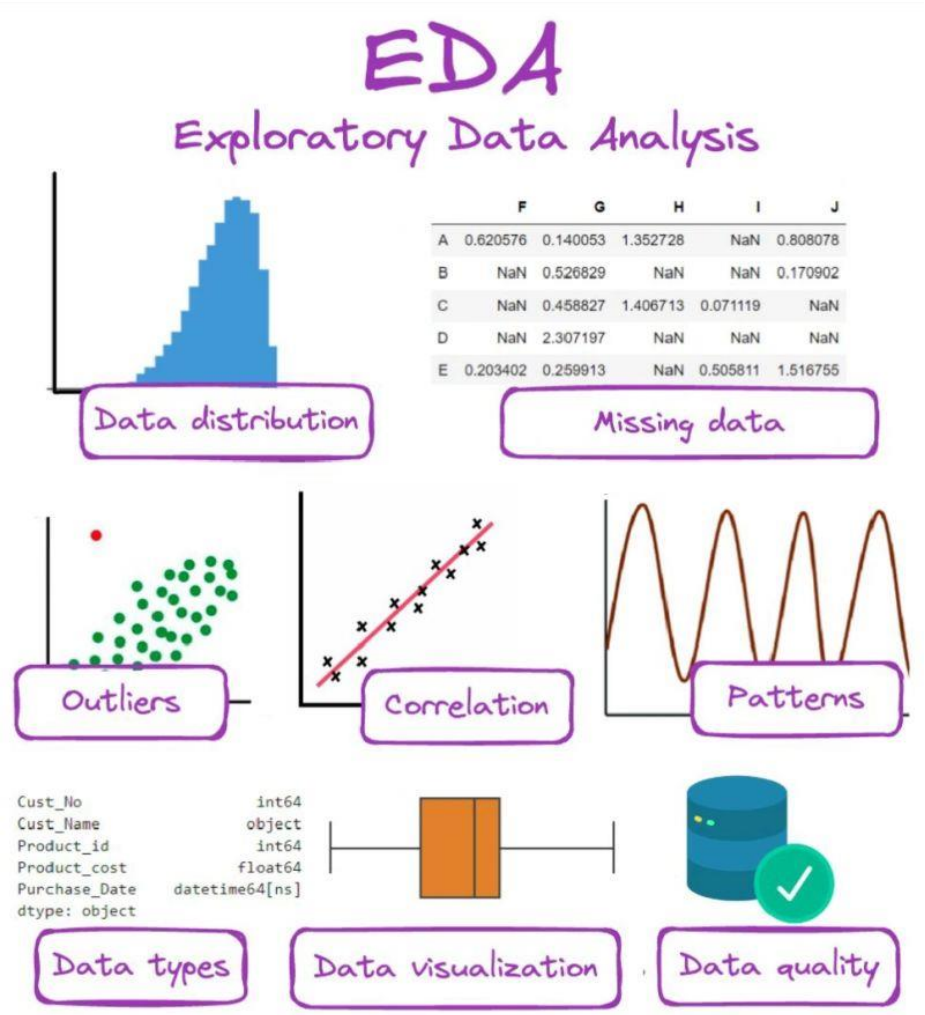
EXPERIMENT – 7.1



	F	G	H	I	J
A	0.620576	0.140053	1.352728	NaN	0.808078
B	NaN	0.526829	NaN	NaN	0.170902
C	NaN	0.458827	1.406713	0.071119	NaN
D	NaN	2.307197	NaN	NaN	NaN
E	0.203402	0.259913	NaN	0.505811	1.516755

```
Cust_No                int64
Cust_Name             object
Product_id             int64
Product_cost         float64
Purchase_Date    datetime64[ns]
dtype: object
```

NAME : JVN GANESH

Roll No : 21BDS0085

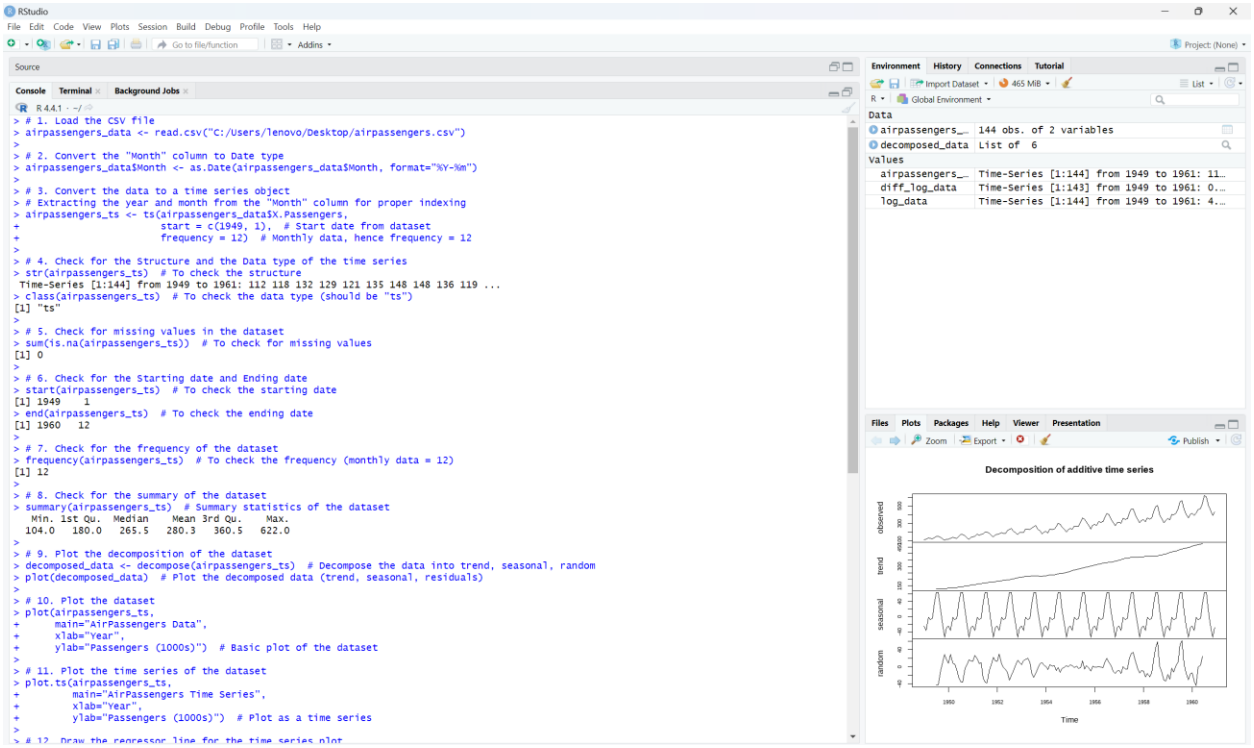
Code:

```
# 1. Load the CSV file
airpassengers_data <- read.csv("C:/Users/lenovo/Desktop/airpassengers.csv")

# 2. Convert the "Month" column to Date type
airpassengers_data$Month <- as.Date(airpassengers_data$Month, format="%Y-%m")

# 3. Convert the data to a time series object
# Extracting the year and month from the "Month" column for proper indexing
airpassengers_ts <- ts(airpassengers_data$X.Passengers,
                       start = c(1949, 1),  # Start date from dataset
                       frequency = 12)  # Monthly data, hence frequency = 12

# 4. Check for the Structure and the Data type of the time series
str(airpassengers_ts)  # To check the structure
class(airpassengers_ts)  # To check the data type (should be "ts")

# 5. Check for missing values in the dataset
sum(is.na(airpassengers_ts))  # To check for missing values

# 6. Check for the Starting date and Ending date
start(airpassengers_ts)  # To check the starting date
end(airpassengers_ts)  # To check the ending date

# 7. Check for the frequency of the dataset
frequency(airpassengers_ts)  # To check the frequency (monthly data = 12)
```

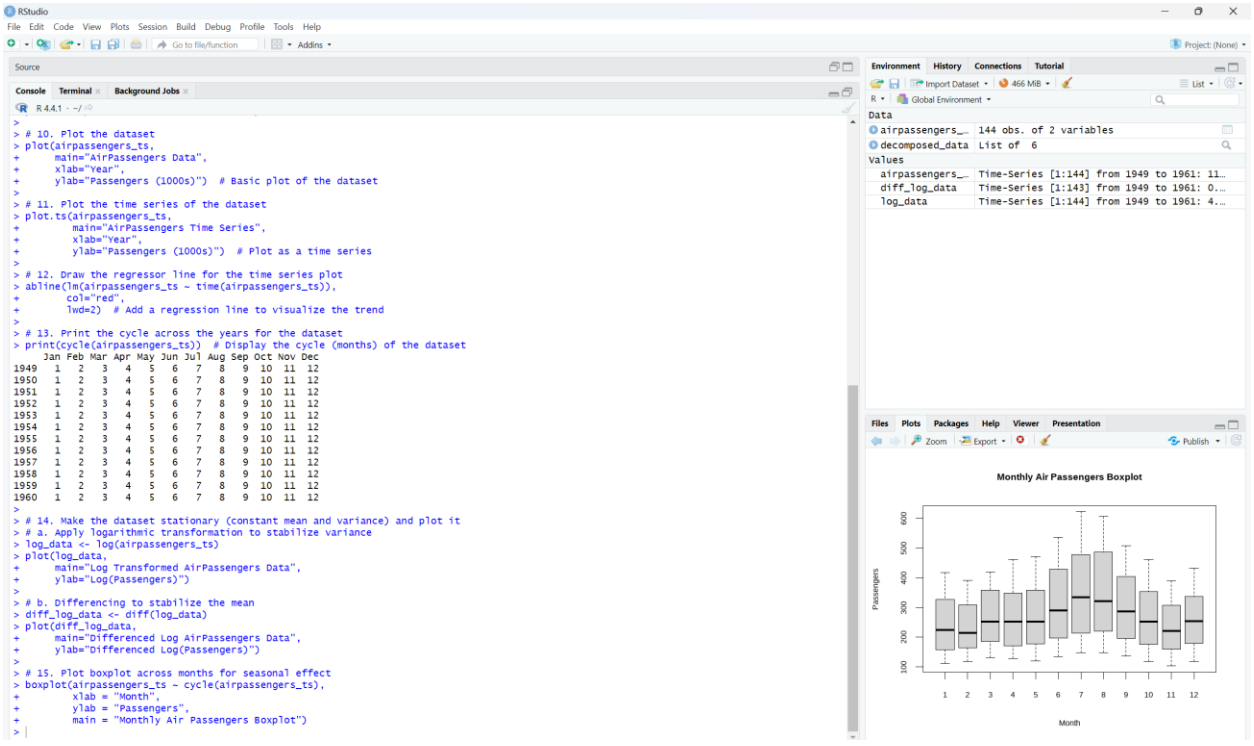
```
# 8. Check for the summary of the dataset
summary(airpassengers_ts)  # Summary statistics of the dataset

# 9. Plot the decomposition of the dataset
decomposed_data <- decompose(airpassengers_ts)  # Decompose the data into trend, seasonal, random
plot(decomposed_data)  # Plot the decomposed data (trend, seasonal, residuals)

# 10. Plot the dataset
plot(airpassengers_ts,
    main="AirPassengers Data",
    xlab="Year",
    ylab="Passengers (1000s)")  # Basic plot of the dataset

# 11. Plot the time series of the dataset
plot.ts(airpassengers_ts,
      main="AirPassengers Time Series",
      xlab="Year",
      ylab="Passengers (1000s)")  # Plot as a time series

# 12. Draw the regressor line for the time series plot
abline(lm(airpassengers_ts ~ time(airpassengers_ts)),
     col="red",
     lwd=2)  # Add a regression line to visualize the trend
```

```r
# 13. Print the cycle across the years for the dataset
print(cycle(airpassengers_ts))  # Display the cycle (months) of the dataset

# 14. Make the dataset stationary (constant mean and variance) and plot it
# a. Apply logarithmic transformation to stabilize variance
log_data <- log(airpassengers_ts)
plot(log_data,
    main="Log Transformed AirPassengers Data",
    ylab="Log(Passengers)")

# b. Differencing to stabilize the mean
diff_log_data <- diff(log_data)
plot(diff_log_data,
    main="Differenced Log AirPassengers Data",
    ylab="Differenced Log(Passengers)")

# 15. Plot boxplot across months for seasonal effect
boxplot(airpassengers_ts ~ cycle(airpassengers_ts),
      xlab = "Month",
      ylab = "Passengers",
      main = "Monthly Air Passengers Boxplot")
```

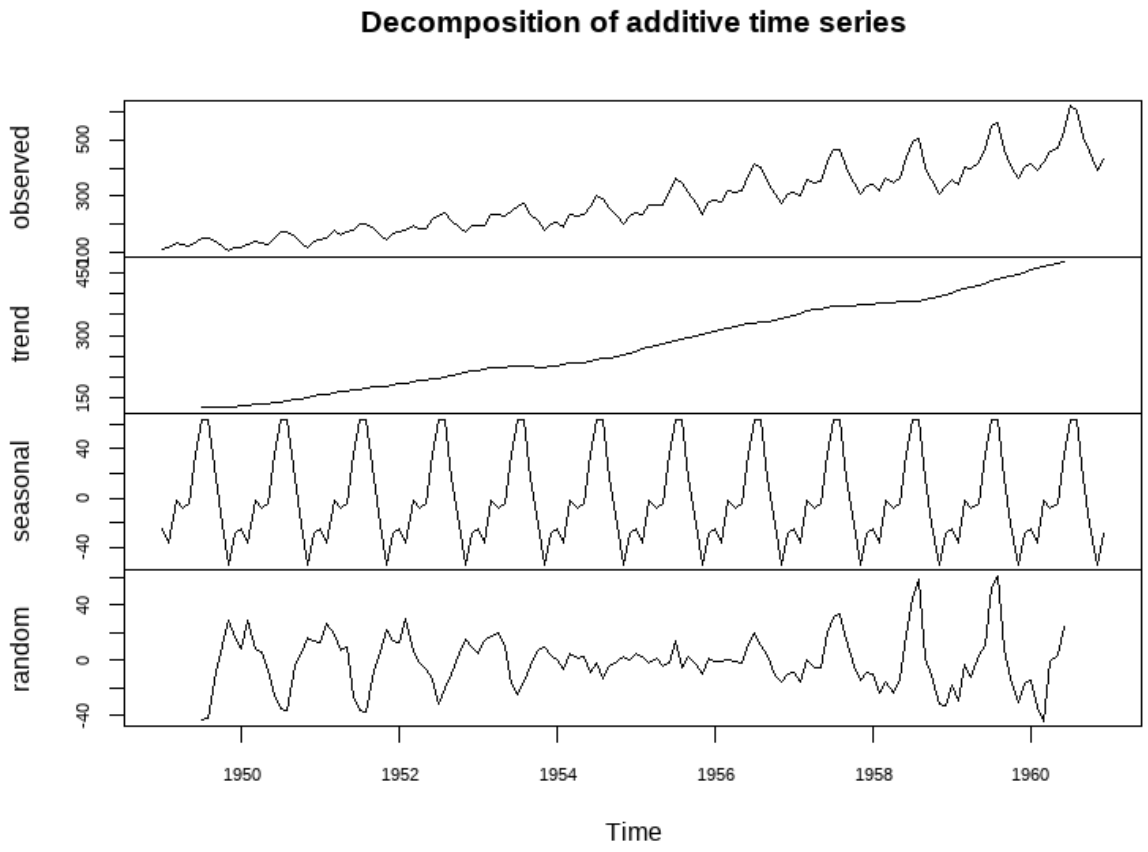
Output Screenshots

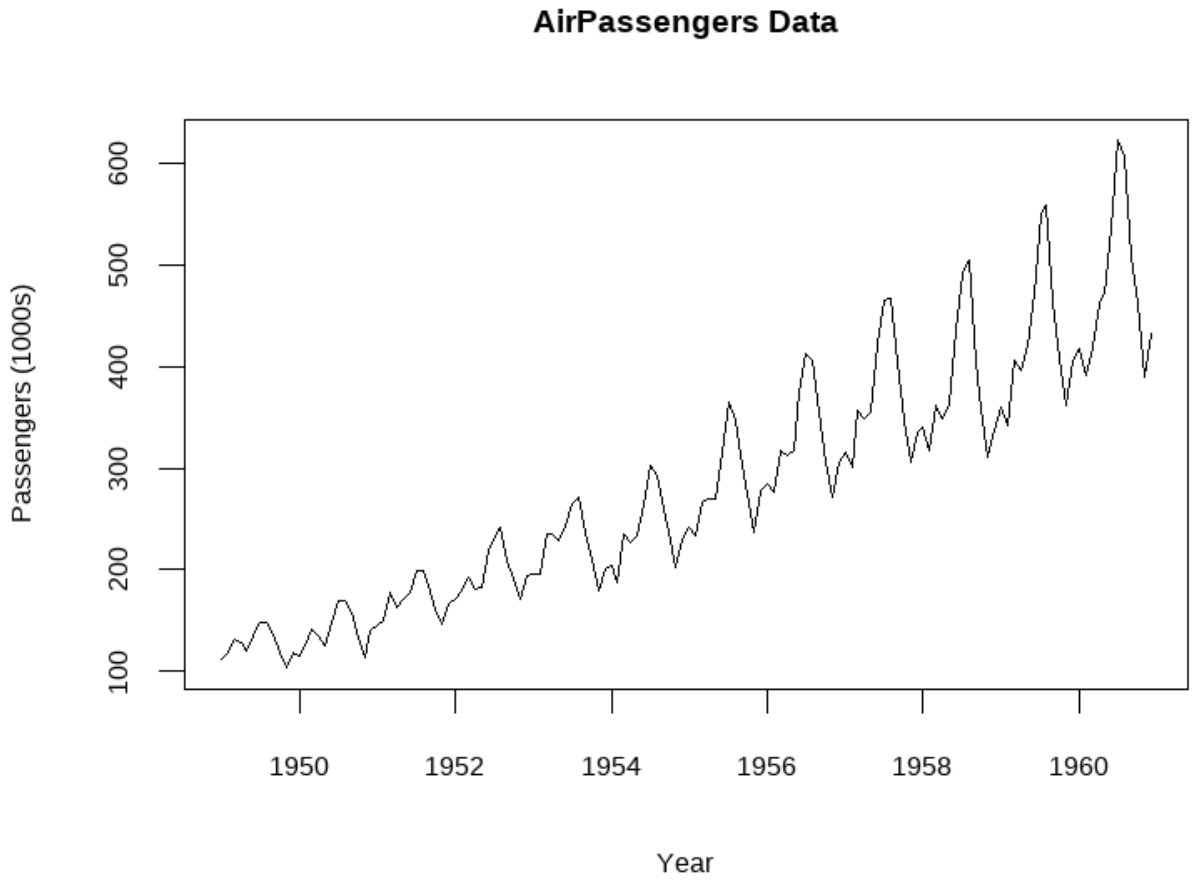


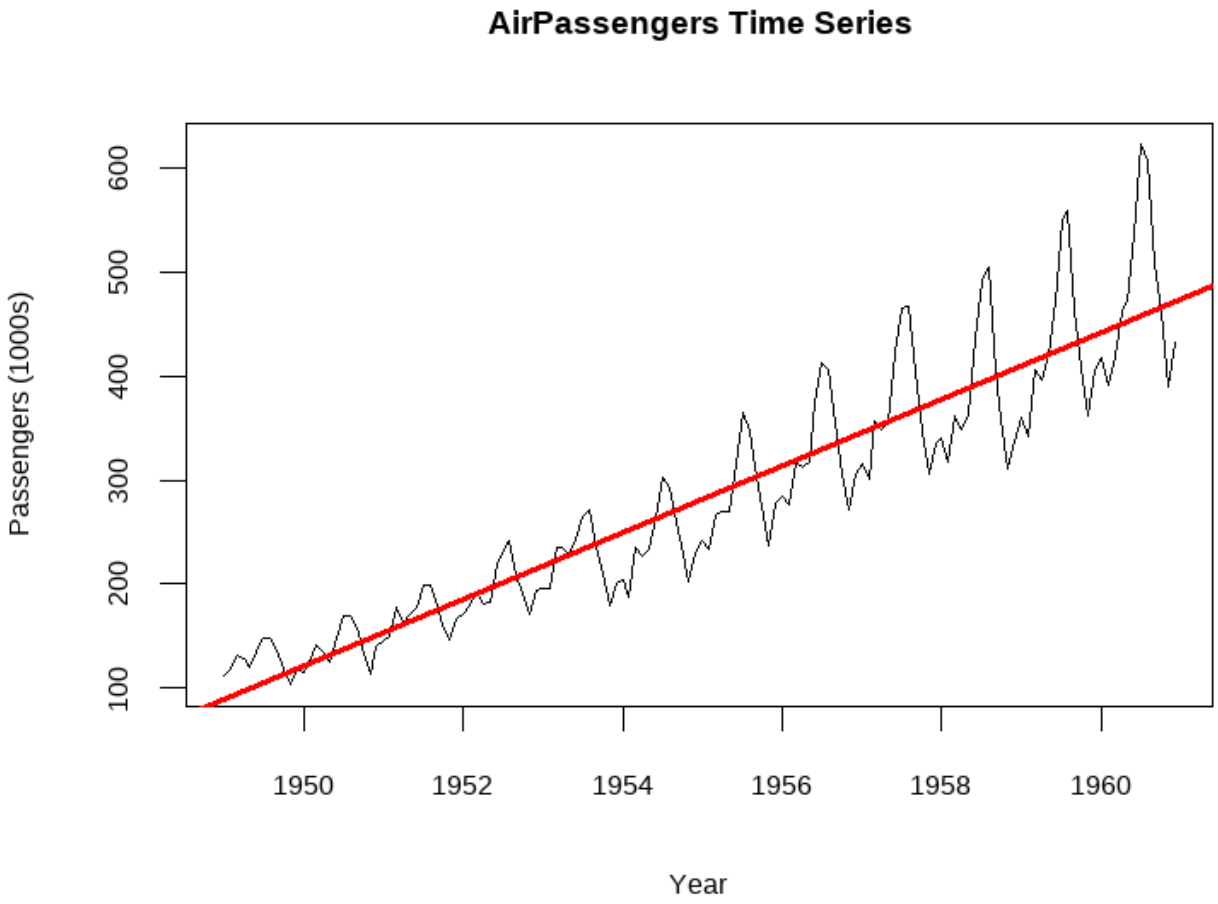
```
> # 1. Load the CSV file
> airpassengers_data <- read.csv("C:/Users/lenovo/Desktop/airpassengers.csv")
>
> # 2. Convert the "Month" column to Date type
> airpassengers_data$Month <- as.Date(airpassengers_data$Month, format="%Y-%m")
>
> # 3. Convert the data to a time series object
> # Extracting the year and month from the "Month" column for proper indexing
> airpassengers_ts <- ts(airpassengers_data$X.Passengers,
+                        start = c(1949, 1),  # Start date from dataset
+                        frequency = 12)  # Monthly data, hence frequency = 12
>
> # 4. Check for the Structure and the Data type of the time series
> str(airpassengers_ts)  # To check the structure
 Time-Series [1:144] from 1949 to 1961: 112 118 132 129 121 135 148 148 136 119 ...
> class(airpassengers_ts)  # To check the data type (should be "ts")
[1] "ts"
>
> # 5. Check for missing values in the dataset
> sum(is.na(airpassengers_ts))  # To check for missing values
[1] 0
>
> # 6. Check for the Starting date and Ending date
> start(airpassengers_ts)  # To check the starting date
[1] 1949    1
> end(airpassengers_ts)  # To check the ending date
[1] 1960   12
>
> # 7. Check for the frequency of the dataset
> frequency(airpassengers_ts)  # To check the frequency (monthly data = 12)
[1] 12
>
> # 8. Check for the summary of the dataset
> summary(airpassengers_ts)  # Summary statistics of the dataset
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  104.0   180.0   265.5   280.3   360.5   622.0
>
> # 9. Plot the decomposition of the dataset
> decomposed_data <- decompose(airpassengers_ts)  # Decompose the data into trend, seasonal, random
> plot(decomposed_data)  # Plot the decomposed data (trend, seasonal, residuals)
>
> # 10. Plot the dataset
> plot(airpassengers_ts,
+      main="AirPassengers Data",
+      xlab="Year",
+      ylab="Passengers (1000s)")  # Basic plot of the dataset
>
> # 11. Plot the time series of the dataset
> plot.ts(airpassengers_ts,
+         main="AirPassengers Time Series",
+         xlab="Year",
+         ylab="Passengers (1000s)")  # Plot as a time series
>
> # 12. Draw the regressor line for the time series plot
```

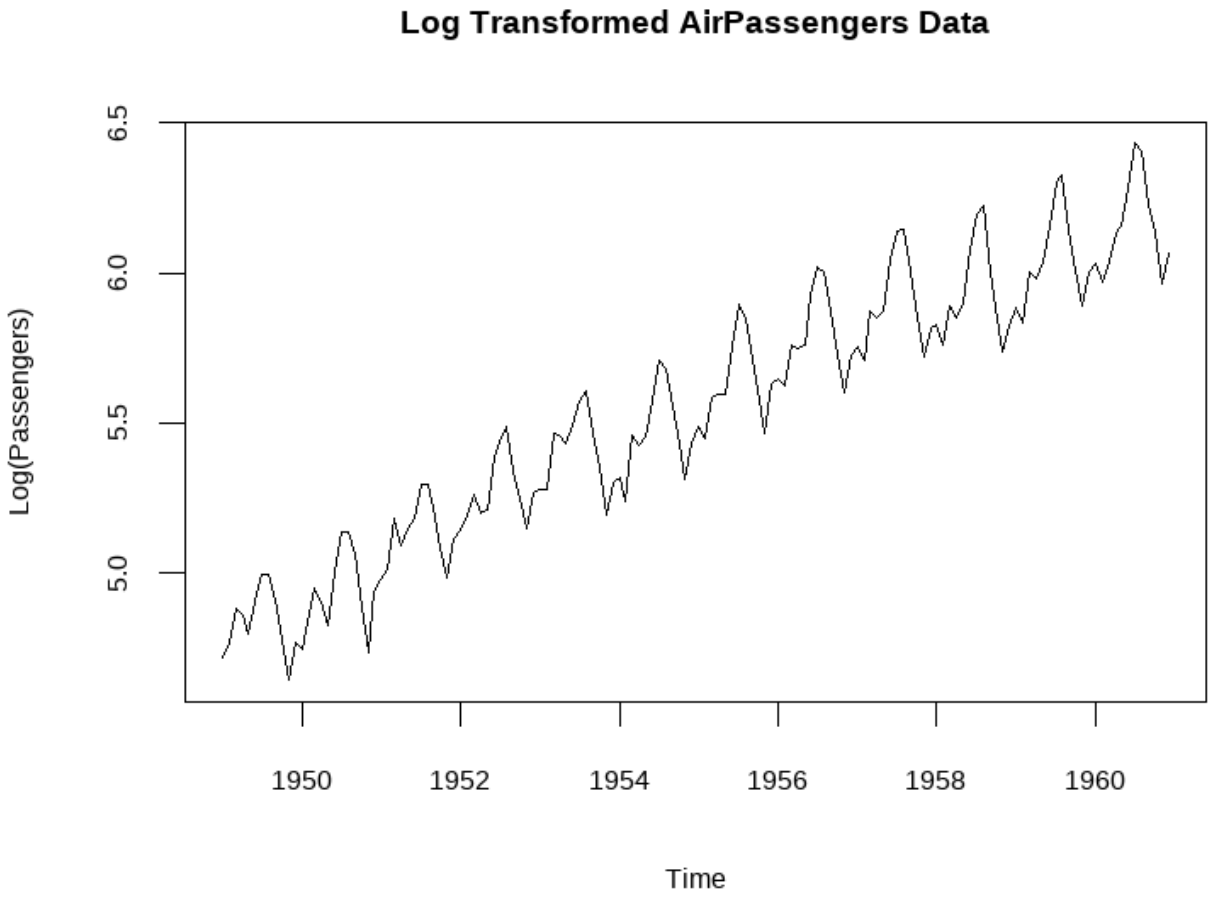


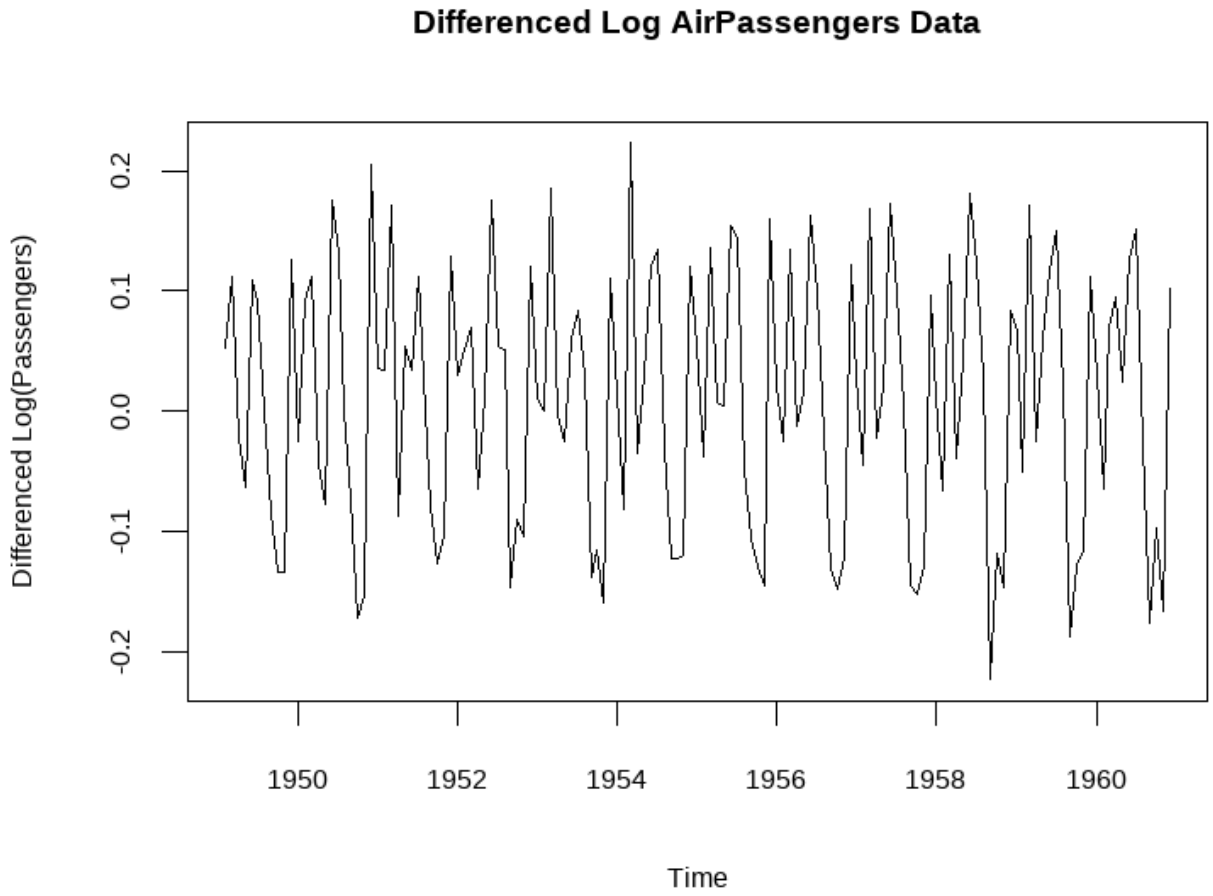
```
>
> # 10. Plot the dataset
> plot(airpassengers_ts,
+      main="AirPassengers Data",
+      xlab="Year",
+      ylab="Passengers (1000s)")  # Basic plot of the dataset
>
> # 11. Plot the time series of the dataset
> plot.ts(airpassengers_ts,
+         main="AirPassengers Time Series",
+         xlab="Year",
+         ylab="Passengers (1000s)")  # Plot as a time series
>
> # 12. Draw the regressor line for the time series plot
> abline(lm(airpassengers_ts ~ time(airpassengers_ts)),
+        col="red",
+        lwd=2)  # Add a regression line to visualize the trend
>
> # 13. Print the cycle across the years for the dataset
> print(cycle(airpassengers_ts))  # Display the cycle (months) of the dataset
     Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
1949   1   2   3   4   5   6   7   8   9  10  11  12
1950   1   2   3   4   5   6   7   8   9  10  11  12
1951   1   2   3   4   5   6   7   8   9  10  11  12
1952   1   2   3   4   5   6   7   8   9  10  11  12
1953   1   2   3   4   5   6   7   8   9  10  11  12
1954   1   2   3   4   5   6   7   8   9  10  11  12
1955   1   2   3   4   5   6   7   8   9  10  11  12
1956   1   2   3   4   5   6   7   8   9  10  11  12
1957   1   2   3   4   5   6   7   8   9  10  11  12
1958   1   2   3   4   5   6   7   8   9  10  11  12
1959   1   2   3   4   5   6   7   8   9  10  11  12
1960   1   2   3   4   5   6   7   8   9  10  11  12
>
> # 14. Make the dataset stationary (constant mean and variance) and plot it
> # a. Apply logarithmic transformation to stabilize variance
> log_data <- log(airpassengers_ts)
> plot(log_data,
+      main="Log Transformed AirPassengers Data",
+      ylab="Log(Passengers)")
>
> # b. Differencing to stabilize the mean
> diff_log_data <- diff(log_data)
> plot(diff_log_data,
+      main="Differenced Log AirPassengers Data",
+      ylab="Differenced Log(Passengers)")
>
> # 15. Plot boxplot across months for seasonal effect
> boxplot(airpassengers_ts ~ cycle(airpassengers_ts),
+         xlab = "Month",
+         ylab = "Passengers",
+         main = "Monthly Air Passengers Boxplot")
>
```

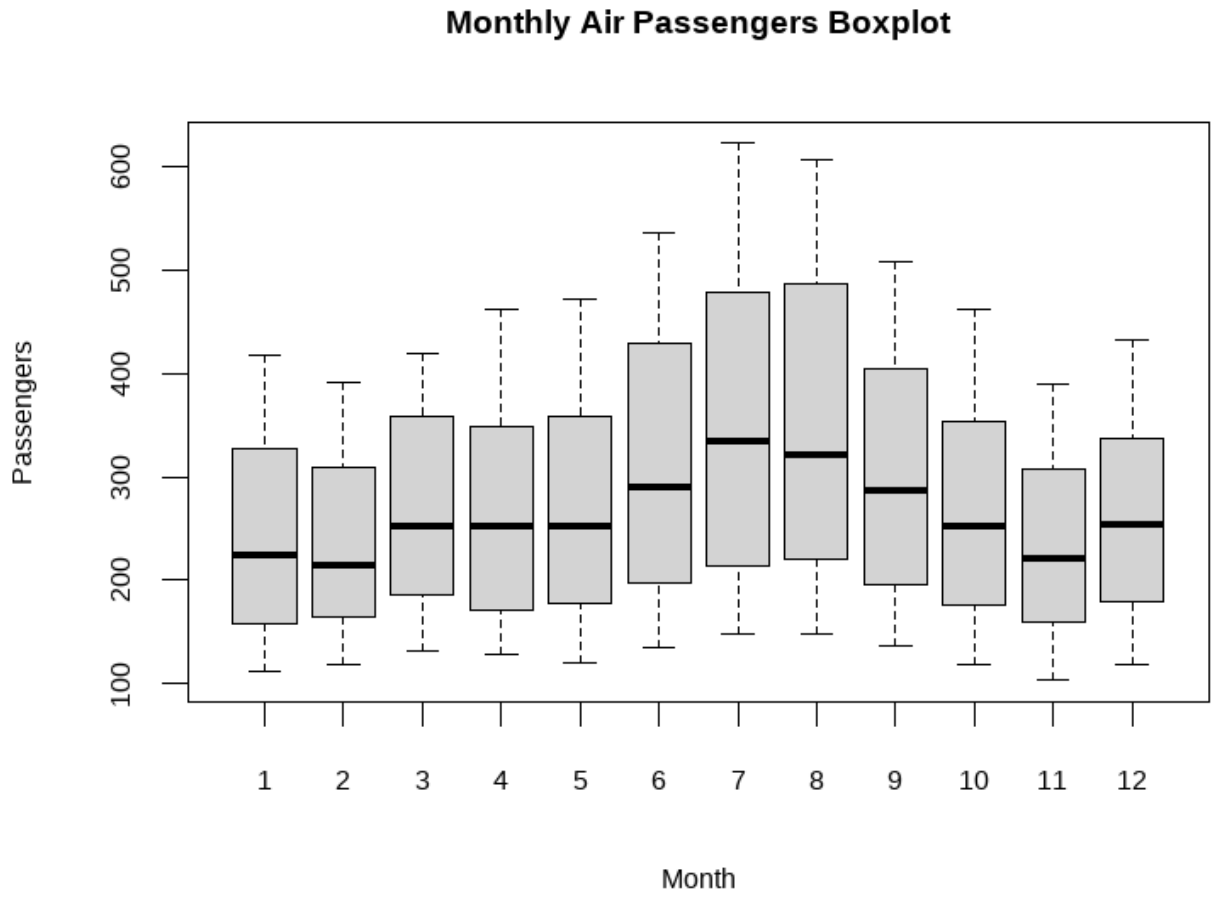

Decomposition of additive time series
observed
trend
seasonal
random
500
300
100
450
300
150
40
0
-40
40
0
-40
1950
1952
1954
1956
1958
1960
Time


AirPassengers Data
Passengers (1000s)
100
200
300
400
500
600
1950
1952
1954
1956
1958
1960
Year


AirPassengers Time Series
Passengers (1000s)
100
200
300
400
500
600
1950
1952
1954
1956
1958
1960
Year


Log Transformed AirPassengers Data
Log(Passengers)
6.5
6.0
5.5
5.0
1950
1952
1954
1956
1958
1960
Time


Differenced Log AirPassengers Data
Differenced Log(Passengers)
0.2
0.1
0.0
-0.1
-0.2
1950
1952
1954
1956
1958
1960
Time



OUTPUT

```
> # 1. Load the CSV file
> airpassengers_data <- read.csv("C:/Users/lenovo/Desktop/airpassengers.csv")
>
> # 2. Convert the "Month" column to Date type
> airpassengers_data$Month <- as.Date(airpassengers_data$Month, format="%Y-%m")
>
> # 3. Convert the data to a time series object
> # Extracting the year and month from the "Month" column for proper indexing
> airpassengers_ts <- ts(airpassengers_data$X.Passengers,
```

```
+                 start = c(1949, 1),  # Start date from dataset
+                 frequency = 12)  # Monthly data, hence frequency = 12
>
> # 4. Check for the Structure and the Data type of the time series
> str(airpassengers_ts)  # To check the structure
 Time-Series [1:144] from 1949 to 1961: 112 118 132 129 121 135 148 148 136 119 ...
> class(airpassengers_ts)  # To check the data type (should be "ts")
[1] "ts"
>
> # 5. Check for missing values in the dataset
> sum(is.na(airpassengers_ts))  # To check for missing values
[1] 0
>
> # 6. Check for the Starting date and Ending date
> start(airpassengers_ts)  # To check the starting date
[1] 1949    1
> end(airpassengers_ts)  # To check the ending date
[1] 1960   12
>
> # 7. Check for the frequency of the dataset
> frequency(airpassengers_ts)  # To check the frequency (monthly data = 12)
[1] 12
>
> # 8. Check for the summary of the dataset
> summary(airpassengers_ts)  # Summary statistics of the dataset
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
```

```
 104.0  180.0  265.5  280.3  360.5  622.0
>
> # 9. Plot the decomposition of the dataset
> decomposed_data <- decompose(airpassengers_ts)  # Decompose the data into trend, seasonal, random
> plot(decomposed_data)  # Plot the decomposed data (trend, seasonal, residuals)
>
> # 10. Plot the dataset
> plot(airpassengers_ts,
+     main="AirPassengers Data",
+     xlab="Year",
+     ylab="Passengers (1000s)")  # Basic plot of the dataset
>
> # 11. Plot the time series of the dataset
> plot.ts(airpassengers_ts,
+       main="AirPassengers Time Series",
+       xlab="Year",
+       ylab="Passengers (1000s)")  # Plot as a time series
>
> # 12. Draw the regressor line for the time series plot
> abline(lm(airpassengers_ts ~ time(airpassengers_ts)),
+      col="red",
+      lwd=2)  # Add a regression line to visualize the trend
>
> # 13. Print the cycle across the years for the dataset
> print(cycle(airpassengers_ts))  # Display the cycle (months) of the dataset
```

```
   Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
1949   1   2   3   4   5   6   7   8   9  10  11  12
1950   1   2   3   4   5   6   7   8   9  10  11  12
1951   1   2   3   4   5   6   7   8   9  10  11  12
1952   1   2   3   4   5   6   7   8   9  10  11  12
1953   1   2   3   4   5   6   7   8   9  10  11  12
1954   1   2   3   4   5   6   7   8   9  10  11  12
1955   1   2   3   4   5   6   7   8   9  10  11  12
1956   1   2   3   4   5   6   7   8   9  10  11  12
1957   1   2   3   4   5   6   7   8   9  10  11  12
1958   1   2   3   4   5   6   7   8   9  10  11  12
1959   1   2   3   4   5   6   7   8   9  10  11  12
1960   1   2   3   4   5   6   7   8   9  10  11  12
>
> # 14. Make the dataset stationary (constant mean and variance) and plot it
> # a. Apply logarithmic transformation to stabilize variance
> log_data <- log(airpassengers_ts)
> plot(log_data,
+     main="Log Transformed AirPassengers Data",
+     ylab="Log(Passengers)")
>
> # b. Differencing to stabilize the mean
> diff_log_data <- diff(log_data)
> plot(diff_log_data,
+     main="Differenced Log AirPassengers Data",
+     ylab="Differenced Log(Passengers)")
```

```
>
> # 15. Plot boxplot across months for seasonal effect
> boxplot(airpassengers_ts ~ cycle(airpassengers_ts),
+       xlab = "Month",
+       ylab = "Passengers",
+       main = "Monthly Air Passengers Boxplot")
>
```