```
correlation <- cor(x, y)

# Print the correlation coefficient
cat("The correlation coefficient is", correlation, "\n")
```

The output of the above code will give you the correlation coefficient between X and Y.
To test the significance of the correlation coefficient, you can perform a hypothesis test to see if the correlation coefficient is significantly different from zero. This test is known as the Pearson's correlation significance test. In R, you can use the cor.test function to perform this test.
Here's how you can perform the significance test in R:

```
# Perform the significance test using the cor.test function
result <- cor.test(x, y)

# Extract the p-value from the result
p_value <- result$p.value

# Check if the p-value is less than the significance level (0.05)
if (p_value < 0.05) {
  print("The correlation is significant.")
}
else {
  print("The correlation is not significant.")
}
```

The cor.test function takes the two variables as arguments and returns the test statistics, including the correlation coefficient, the p-value, and the degrees of freedom. If the p-value is less than the significance level (0.05), you can reject the null hypothesis that the correlation coefficient is equal to zero and conclude that the correlation is significant.

15) Consider the inbuilt data set cars.
Find Correlation between possible variables and pairwise correlation
Find regression line between appropriate variables
Display the summary statistics and comment on the results

The cars data set in R contains the speed and stopping distances of 50 cars. Here's how you can perform the steps you mentioned in your question:
Find the correlation between possible variables and pairwise correlation:

```
# Load the cars data set
data(cars)

# View the data
head(cars)

# Calculate the pairwise correlation matrix
cor_matrix <- cor(cars)

# View the correlation matrix
cor_matrix
```

Find the regression line between appropriate variables:

```
# Fit a linear regression model to predict stopping distance from speed
model <- lm(dist ~ speed, data = cars)

# Summarize the model
```