#importing data

**data<- read.csv("Hours and Scores.csv")**

**View(data)**

|  |
| --- |
| Hours Scores  1 2.5 21  2 5.1 47  3 3.2 27  4 8.5 75  5 3.5 30  6 1.5 20  7 9.2 88  8 5.5 60  9 8.3 81  10 2.7 25  11 7.7 85  12 5.9 62  13 4.5 41  14 3.3 42  15 1.1 17  16 8.9 95  17 2.5 30  18 1.9 24  19 6.1 67  20 7.4 69  21 2.7 30  22 4.8 54  23 3.8 35  24 6.9 76  25 7.8 86 |
|  |
| |  | | --- | |  | |

**summary(data)**

|  |
| --- |
| Hours Scores  Min. :1.100 Min. :17.00  1st Qu.:2.700 1st Qu.:30.00  Median :4.800 Median :47.00  Mean :5.012 Mean :51.48  3rd Qu.:7.400 3rd Qu.:75.00  Max. :9.200 Max. :95.00 |
|  |
| |  | | --- | | #Splitting the dataset into traing and testing set  **library(caTools)**  **set.seed(123)**  **split<- sample.split(data$Scores, SplitRatio = 0.7)**  **training\_set<- subset(data, split==TRUE)**  **View(training\_set)** | |

Hours Scores

1 2.5 21

2 5.1 47

3 3.2 27

6 1.5 20

7 9.2 88

9 8.3 81

10 2.7 25

12 5.9 62

13 4.5 41

14 3.3 42

15 1.1 17

17 2.5 30

18 1.9 24

19 6.1 67

22 4.8 54

23 3.8 35

25 7.8 86

**test\_set<- subset(data, split==FALSE)**

**View(test\_set)**

Hours Scores

4 8.5 75

5 3.5 30

8 5.5 60

11 7.7 85

16 8.9 95

20 7.4 69

21 2.7 30

24 6.9 76

#Fitting the Simple Linear Regression to the training set

**regressor<- lm(formula = Scores~Hours, data = training\_set)**

**summary(regressor)**

Call:

lm(formula = Scores ~ Hours, data = training\_set)

Residuals:

Min 1Q Median 3Q Max

-6.871 -4.665 2.057 3.407 7.711

Coefficients:

Estimate Std. Error t value

(Intercept) 2.9717 2.6216 1.134

Hours 9.6561 0.5286 18.267

Pr(>|t|)

(Intercept) 0.275

Hours 1.17e-11 \*\*\*

---

Signif. codes:

0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.132 on 15 degrees of freedom

Multiple R-squared: 0.957, Adjusted R-squared: 0.9541

F-statistic: 333.7 on 1 and 15 DF, p-value: 1.168e-11

#Predicting for test set

**pred<- predict(regressor, newdata = test\_set)**

**pred**

4 5 8 11 16 20 21 24

85.04837 36.76798 56.08014 77.32351 88.91080 74.42668 29.04312 69.59865

#PREDICT SCORE IF A STUDENT STUDY FOR 9.25hrs A DAY

**pred\_score<- predict(regressor, data.frame(Hours=9.25))**

**pred\_score**

92.29043

#VISUALIZATION

**library(ggplot2)**

#visualising the dataset provided

**ggplot()+**

**geom\_point(aes(x=data$Hours, y=data$Scores),**

**colour="red", size=2.5)+**

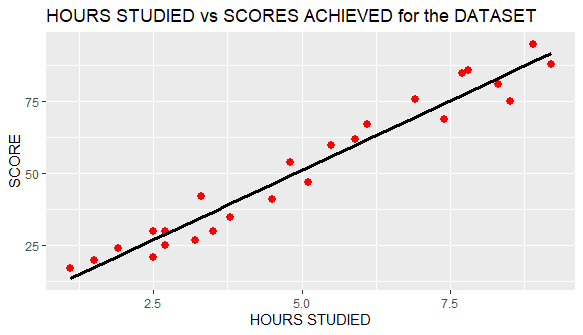
**geom\_line(aes(x=data$Hours, y=predict(regressor, newdata = data)),**

**colour="black", size=1.3)+**

**xlab("HOURS STUDIED")+**

**ylab("SCORE")+**

**ggtitle("HOURS STUDIED vs SCORES ACHIEVED for the DATASET")**

****

#visualising training set

**ggplot()+**

**geom\_point(aes(x=training\_set$Hours, y=training\_set$Scores),**

**colour="red", size=2.5)+**

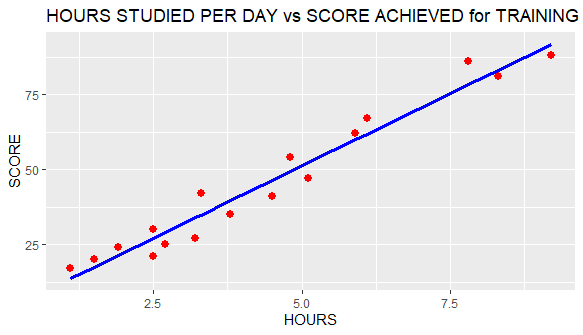
**geom\_line(aes(x=training\_set$Hours, y=predict(regressor, newdata = training\_set)),**

**color="blue", size=1.3)+**

**xlab("HOURS")+**

**ylab("SCORE")+**

**ggtitle("HOURS STUDIED PER DAY vs SCORE ACHIEVED for TRAINING SET")**

****

#visualising for test set

**ggplot()+**

**geom\_point(aes(x=test\_set$Hours, y=test\_set$Scores),**

**colour="red", size=2.5)+**

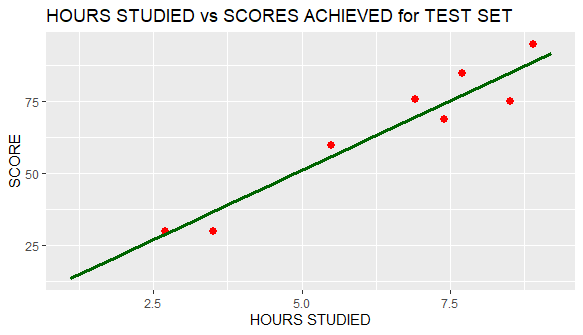
**geom\_line(aes(x=training\_set$Hours, y=predict(regressor, newdata = training\_set)),**

**colour="dark green", size=1.3)+**

**xlab("HOURS STUDIED")+**

**ylab("SCORE")+**

**ggtitle("HOURS STUDIED vs SCORES ACHIEVED for TEST SET")**

****