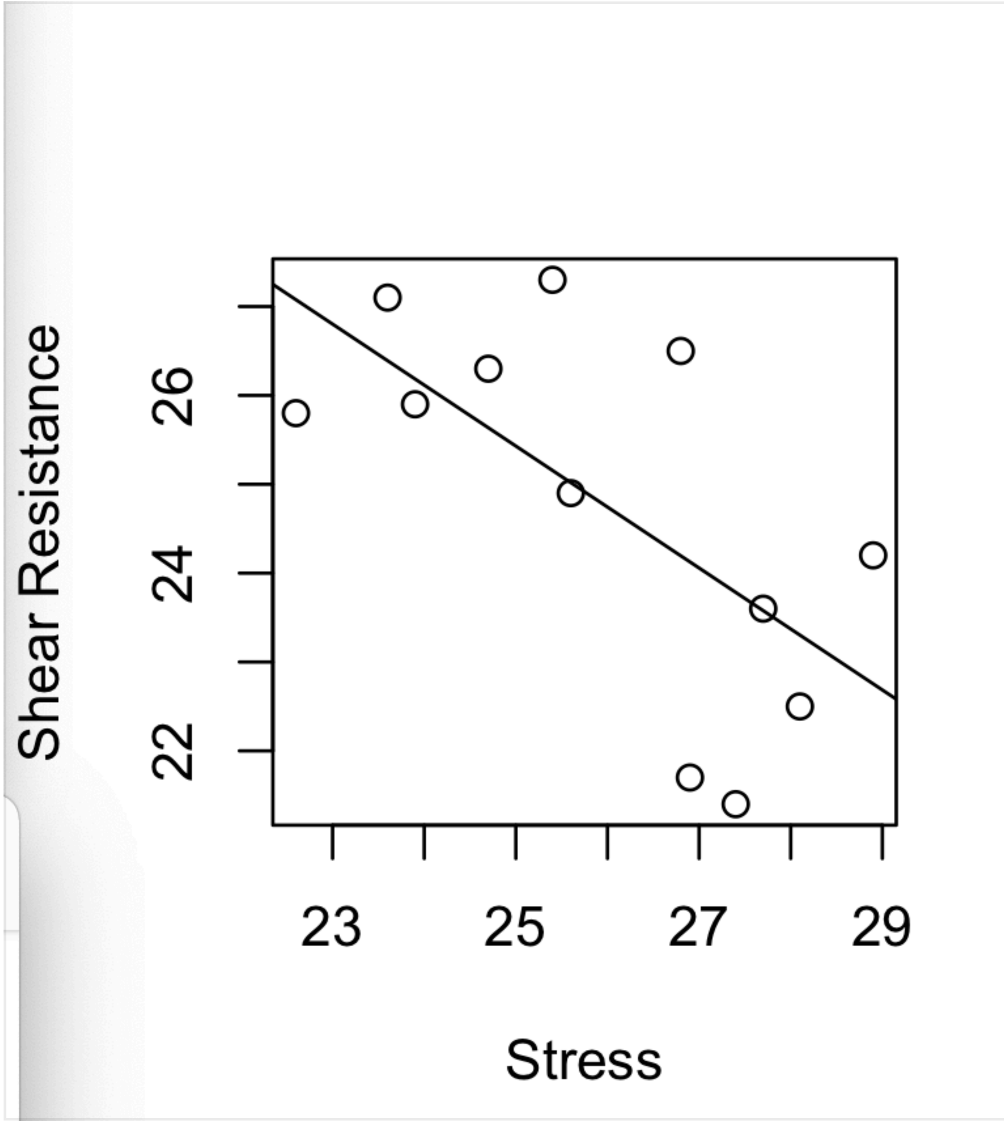
STATS HW

Question 1

Create a scatterplot of y vs. x.
> data <- read.table("~/Downloads/Ex11.06.txt", header = TRUE)
> plot(data[,1], data[,2], xlab = "Stress", ylab = "Shear Resistance")

Fit a simple linear regression model using y as the response and plot the regression line (with the data).
> data_lm <- lm(Shear_resistance ~ Stress, data = data)
> abline(data_lm)



Test whether x is a significant predictor.

```
> summary(data_lm)

Call:
lm(formula = Shear_resistance ~ Stress, data = data)

Residuals:
     Min       1Q   Median       3Q      Max
-2.42633 -0.92139 -0.04785  0.89367  2.30506

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  42.5818     6.5065   6.544 6.52e-05 ***
Stress       -0.6861     0.2499  -2.745   0.0206 *
---
```

```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.64 on 10 degrees of freedom
Multiple R-squared:  0.4298,  Adjusted R-squared:  0.3727
F-statistic: 7.537 on 1 and 10 DF,  p-value: 0.02064

> anova(data_lm)
Analysis of Variance Table

Response: Shear_resistance
          Df Sum Sq Mean Sq F value  Pr(>F)
Stress     1 20.262 20.2621  7.5367 0.02064 *
Residuals 10 26.884  2.6885
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

According to a Stack Overflow thread, "The p-value in the last column tells you the significance of the regression coefficient for a given parameter. If the p-value is small enough to claim statistical significance, that just means there is strong evidence that the coefficient is different from 0." Therefore, with the p-value of Stress being approx 0.021, it is small enough to claim statistical significance.

Create and interpret a 95% CI around the slope coefficient.

```
> confint(data_lm, level = 0.95)
                2.5 %     97.5 %
(Intercept) 28.084338 57.0792671
Stress      -1.242908 -0.1292458
```

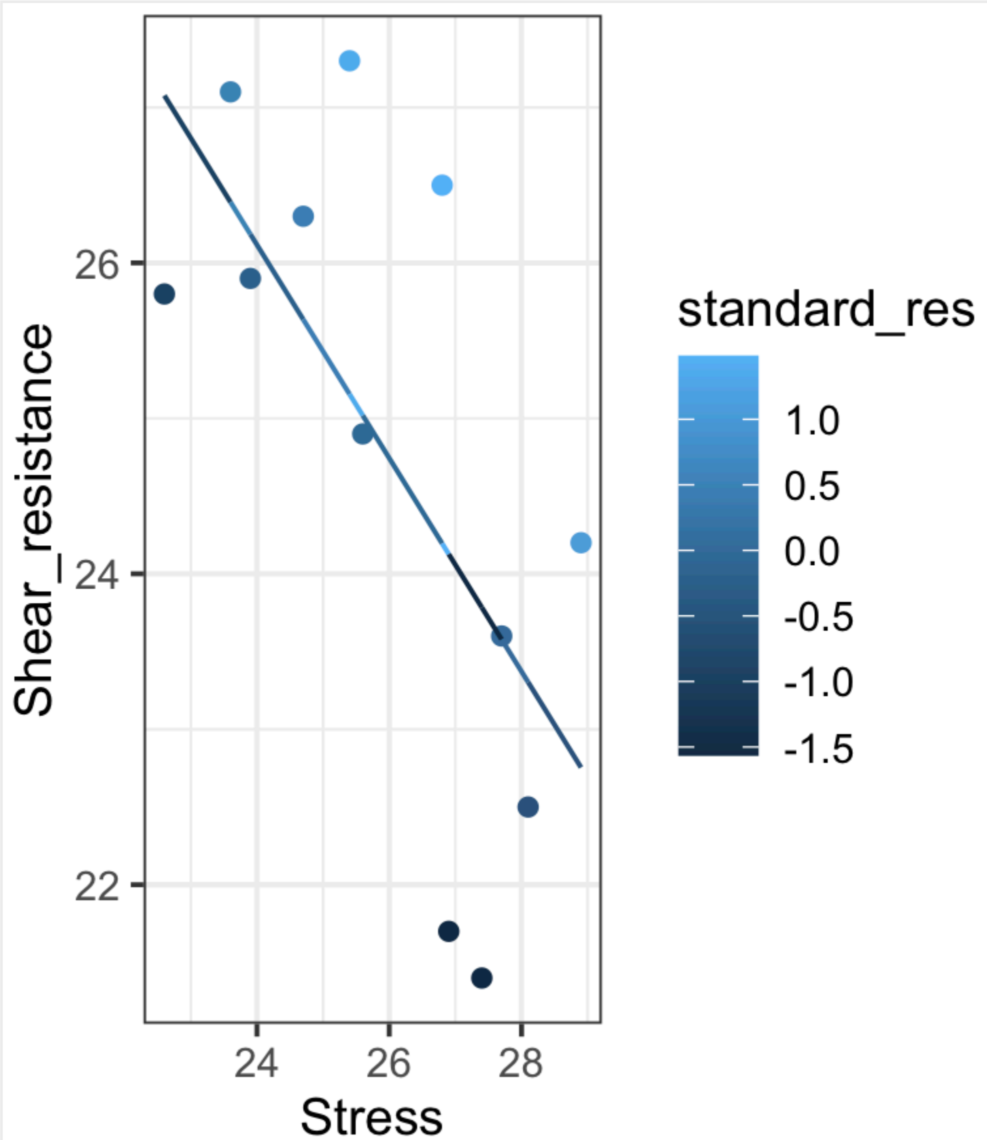
We can be 95% confident that the slope coefficient is between -1.24 and -0.13

Create a normal qq-plot of the standardized residuals. Does the assumption of normally distributed errors seem to be violated? Explain.

```
> standard_res <- rstandard(data_lm)
> final_data <- cbind(data, standard_res)
> final_data[order(-standard_res),]
   Stress Shear_resistance standard_res
1    26.8             26.5   1.48143209
2    25.4             27.3   1.37168658
3    28.9             24.2   1.04153035
4    23.6             27.1   0.48798909
7    24.7             26.3   0.43203879
5    27.7             23.6   0.01493387
12   25.6             24.9  -0.07544068
6    23.9             25.9  -0.19195180
```

```
8    28.1            22.5  -0.54386207
11   22.6            25.8  -0.96311785
10   27.4            21.4  -1.55930318
9    26.9            21.7  -1.56293208
> plot(final_data$Stress, standard_res, ylab='Standardized Residuals', xlab='Stress')
```

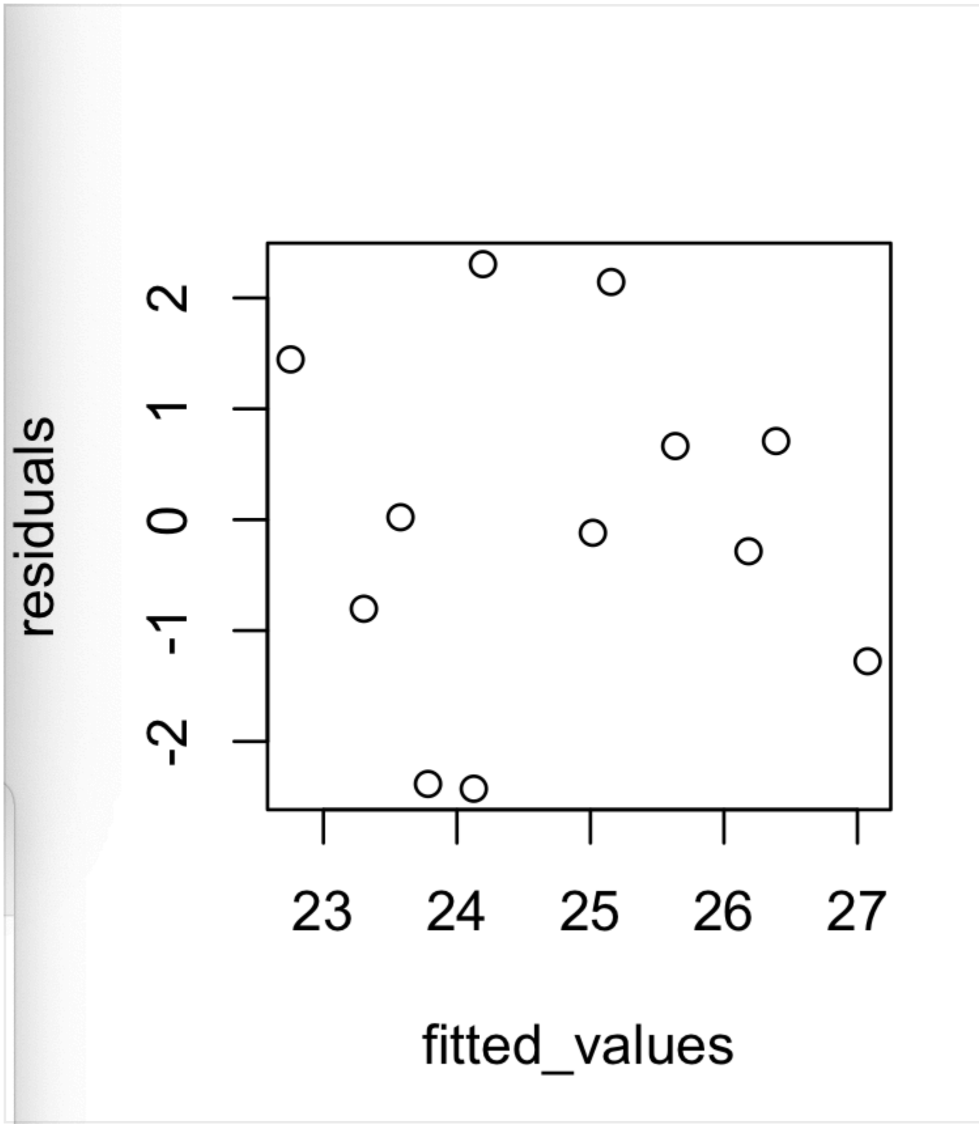

```
> library(ggplot2)
> ggplot(data = final_data, mapping = aes(x = Stress, y = Shear_resistance, color = standard_res)) + geom_point() + geom_line(aes(y = pred)) + theme_bw()
```



The assumption of normally distributed errors seems to be violated, since the points do not seem to form any sort of line. They are scattered around the plot.

Create a plot of the residuals vs. the fitted values. Does the assumption of homoscedasticity of the errors seem to be violated? Explain.

```
> b0 <- means[2] - b1*means[1]
> fitted_values <- b0 + b1*data[,1]
> residuals <- data[,2] - fitted_values
> plot(fitted_values, residuals)
```



The assumption of homoscedasticy of errors seems to hold, as the residuals to not converge to any point, they remain generally distributed.

Report and interpret the coefficient of determination.

```
> df <- data.frame(fitted_values, residuals)
> model <- lm(residuals ~ fitted_values, data = df)
> summary(model)
Call:
lm(formula = residuals ~ fitted_values, data = df)

Residuals:
    Min      1Q  Median      3Q     Max 
-37.993  -6.592   4.375  10.230  22.941 
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    1.084e-14  2.741e+01       0        1
fitted_values -1.780e-16  4.632e-01       0        1

Residual standard error: 16.17 on 18 degrees of freedom
Multiple R-squared:  2.6e-32, Adjusted R-squared:  -0.05556
F-statistic: 4.68e-31 on 1 and 18 DF,  p-value: 1
```

The coefficient of determination is very close to 0. The means that 0% of the variation of the residuals can be explained by the fitted values.

Estimate the shear resistance for a normal stress of x = 24.5.

```
> x=24.5
> print(b0+(b1*x))
Shear_resistance
       25.77291
```

Construct a 95% CI for the mean shear resistance at a normal stress of x = 24.5.

```
> predict(data_lm, newdata = data.frame(Stress = 24.5), interval = "confidence")
       fit      lwr      upr
1 25.77291 24.43903 27.10679
```

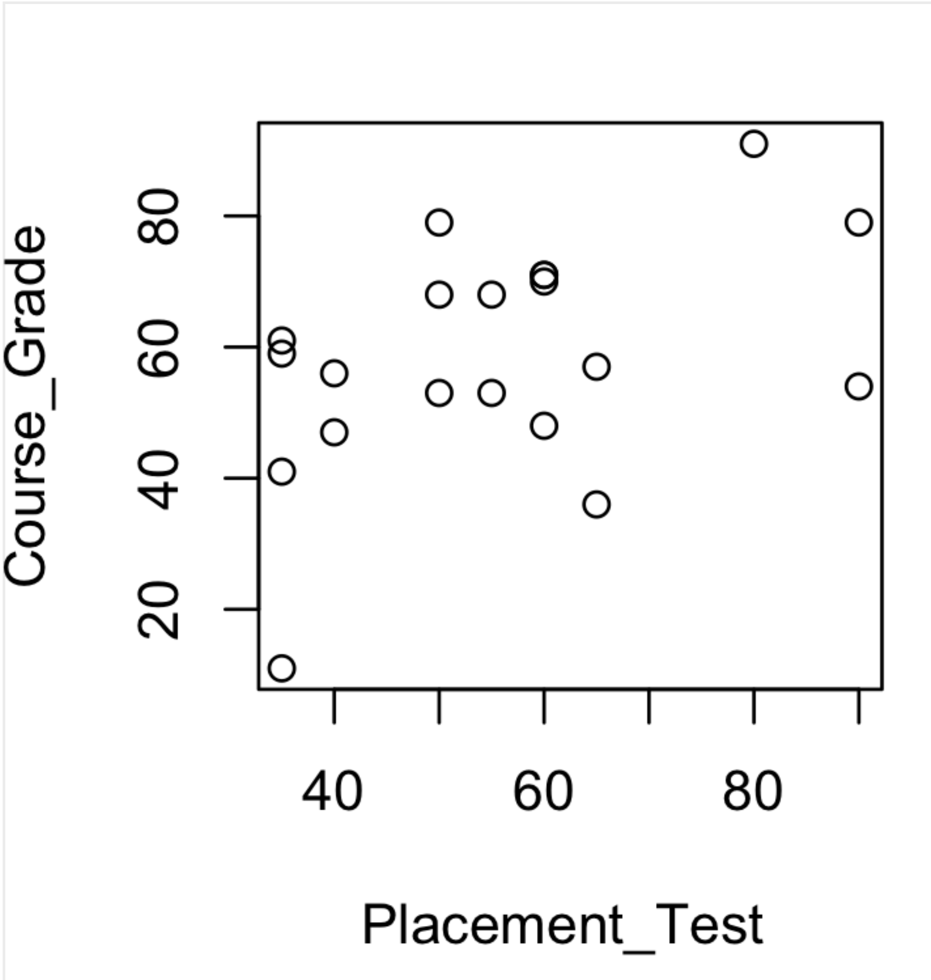
For x=40, can we use the model to estimate E(Y|X=40)? Explain why or why not.

For x = 40, we cannot use the model to estimate Y. This is because extrapolating outside the current data set is never guaranteed and can result in improper conclusions.

Question 2

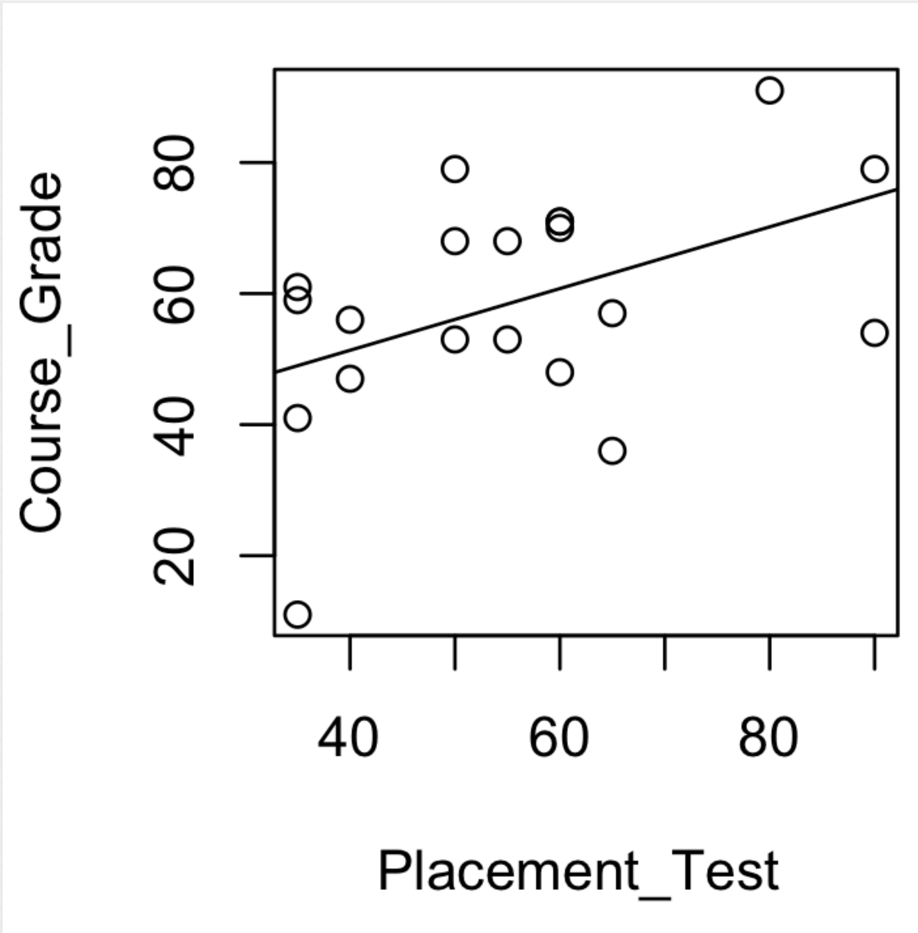
Create a scatterplot of y vs. x.

```
> data <- read.table("~/Downloads/Ex11.08.txt", header = TRUE)
> plot(data[,1], data[,2], xlab = "Placement_Test ", ylab = "Course_Grade")
```



Fit a simple linear regression model using y as the response and plot the regression line (with the data).

```
> data_lm <- lm(Course_Grade ~ Placement_Test, data = data)
> abline(data_lm)
```



Test whether x is a significant predictor.

```
> summary(data_lm)

Call:
lm(formula = Course_Grade ~ Placement_Test, data = data)

Residuals:
    Min      1Q  Median      3Q     Max 
-37.993  -6.592   4.375  10.230  22.941 

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     32.5059    12.6386   2.572   0.0192 *
Placement_Test   0.4711     0.2182   2.159   0.0446 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 16.17 on 18 degrees of freedom
Multiple R-squared:  0.2057,	Adjusted R-squared:  0.1615 
F-statistic: 4.661 on 1 and 18 DF,  p-value: 0.04461
```

```
> anova(data_lm)
Analysis of Variance Table

Response: Course_Grade
             Df Sum Sq Mean Sq F value  Pr(>F)
Placement_Test  1 1219.4 1219.35  4.6607 0.04461 *
Residuals      18 4709.2  261.62
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

With the p-value of Stress being approx 0.045, it is small enough to claim statistical significance.

Create and interpret a 95% CI around the slope coefficient.

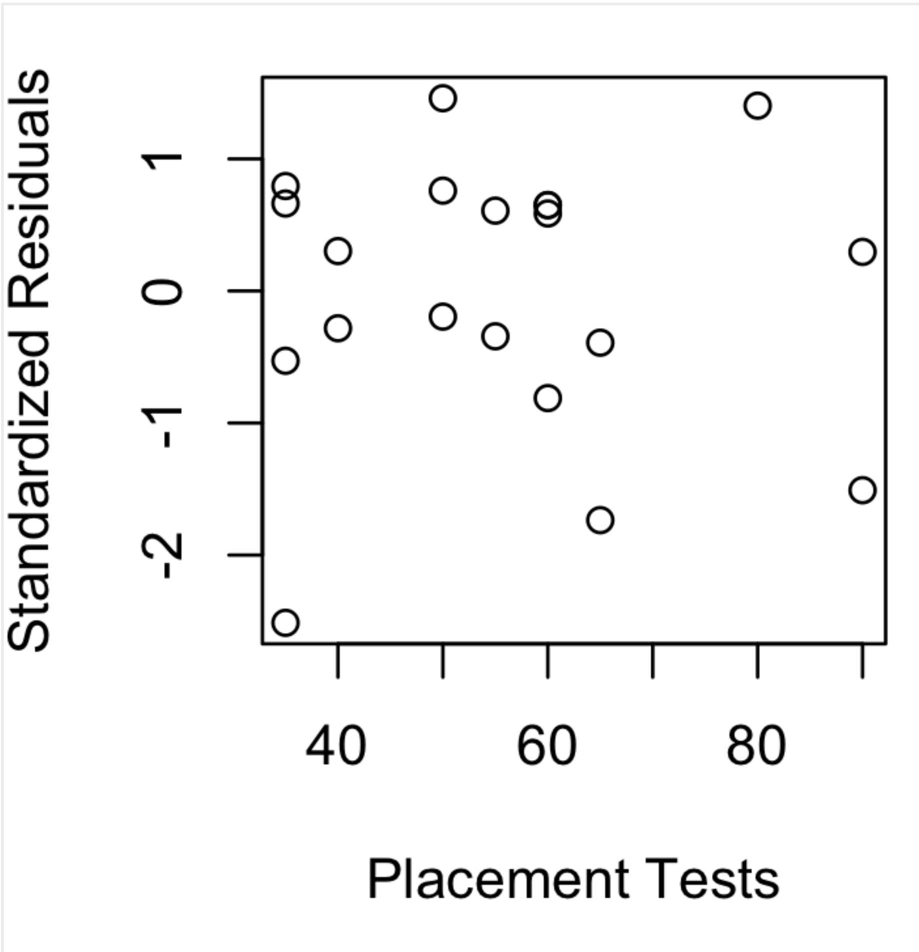
```
> confint(data_lm, level = 0.95)
                2.5 %     97.5 %
(Intercept)    5.9531568 59.0586722
Placement_Test 0.0126448  0.9294844
```

We are 95% confident that the slope coefficient is between 0.01 and 0.93

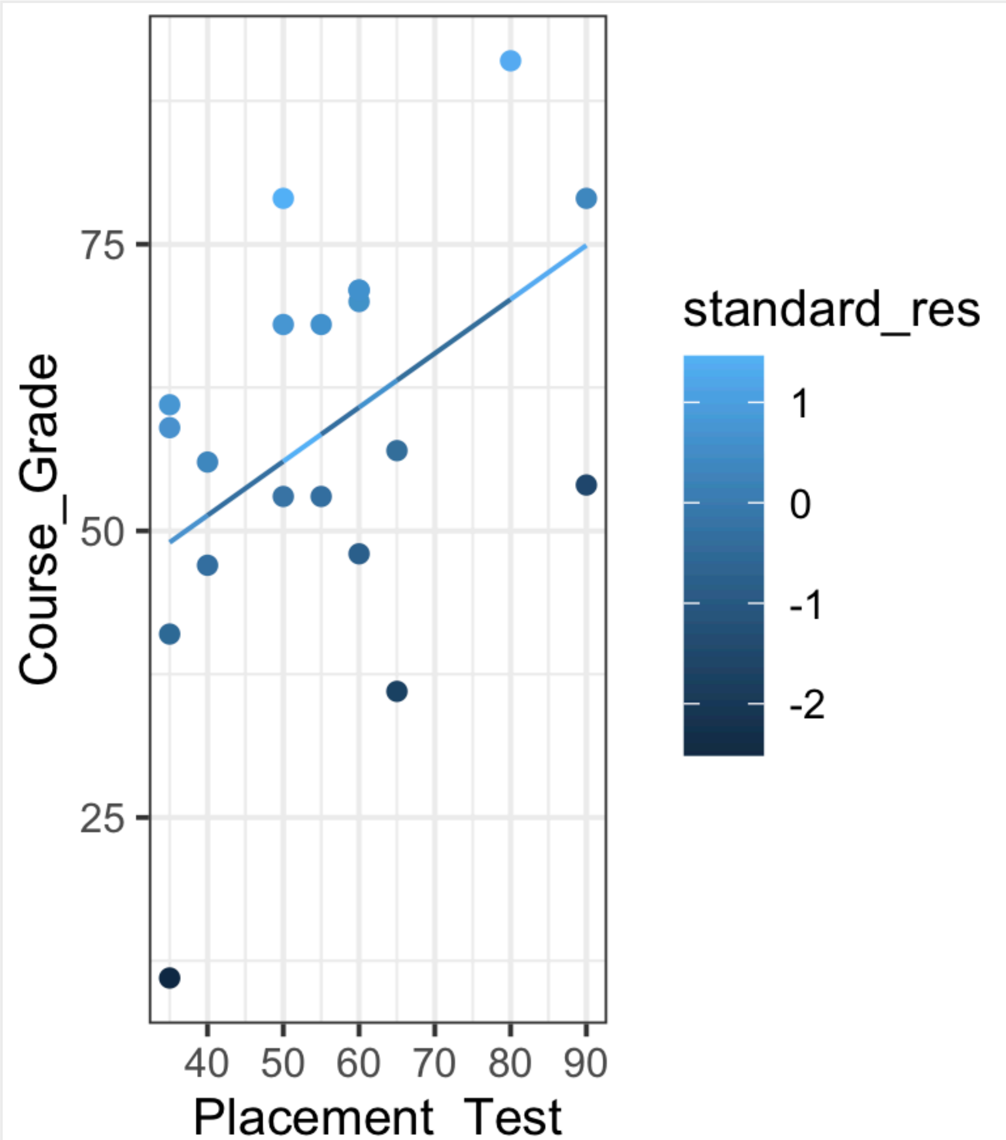
Create a normal qq-plot of the standardized residuals. Does the assumption of normally distributed errors seem to be violated? Explain.

```
> standard_res <- rstandard(data_lm)
> final_data <- cbind(data, standard_res)
> final_data[order(-standard_res),]
   Placement_Test Course_Grade standard_res
20             50           79    1.4593938
12             80           91    1.4030562
3              35           61    0.7942443
18             50           68    0.7596234
10             35           59    0.6619455
14             60           71    0.6501737
15             60           71    0.6501737
5              55           68    0.6080336
8              60           70    0.5866194
4              40           56    0.3020821
9              90           79    0.2958658
1              50           53   -0.1946090
16             40           47   -0.2824043
17             55           53   -0.3434528
19             65           57   -0.3919242
2              35           41   -0.5287438
```

```
13            60          48   -0.8115750
11            90          54   -1.5089549
6             65          36   -1.7356392
7             35          11   -2.5132260
> plot(final_data$Placement_Test, standard_res, ylab='Standardized Residuals', xlab='Placement Tests')
```



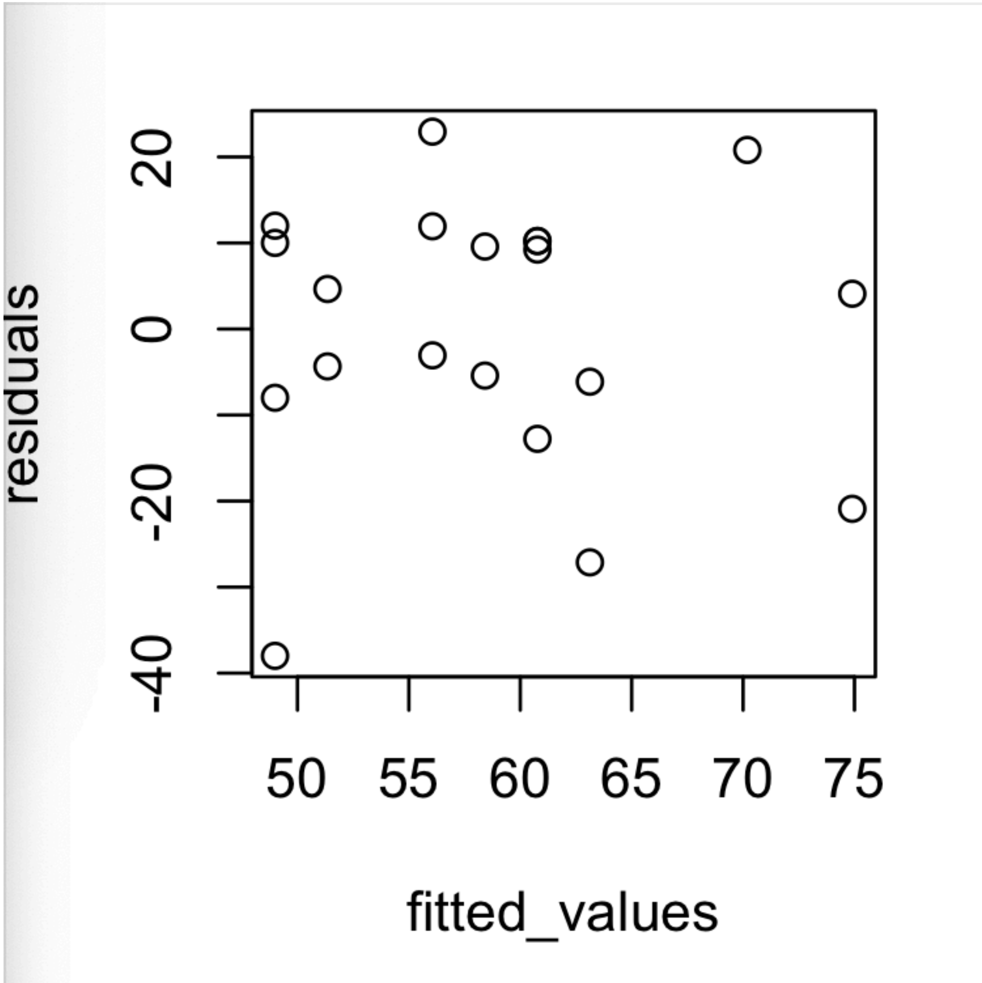

```
> library(ggplot2)
> final_data$pred <- predict(data_lm)
> ggplot(data = final_data, mapping = aes(x = Placement_Test, y = Course_Grade, color = standard_res)) + geom_point() + geom_line(aes(y = pred)) + theme_bw()
```



The assumption of normally distributed errors seems to be violated, since the points do not seem to form any sort of line. They are scattered around the plot.

Create a plot of the residuals vs. the fitted values. Does the assumption of Homoscedasticity of the errors seem to be violated? Explain.

```
> b0 <- means[2] - b1*means[1]
> fitted_values <- b0 + b1*data[,1]
> residuals <- data[,2] - fitted_values
> plot(fitted_values, residuals)
```



Assumption of homoscedasticity seems to hold. The values are evenly distributed about the 0 line.

Report and interpret the coefficient of determination.

```
> df <- data.frame(fitted_values, residuals)
> model <- lm(residuals ~ fitted_values, data = df)
> summary(model)

Call:
lm(formula = residuals ~ fitted_values, data = df)

Residuals:
    Min      1Q  Median      3Q     Max
-37.993  -6.592   4.375  10.230  22.941

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.084e-14  2.741e+01       0        1
```

```
fitted_values -1.780e-16  4.632e-01       0        1

Residual standard error: 16.17 on 18 degrees of freedom
Multiple R-squared:  2.6e-32, Adjusted R-squared:  -0.05556
F-statistic: 4.68e-31 on 1 and 18 DF,  p-value: 1
```

Essentially 0% of the variation of the residuals can be explained by the fitted values.

Estimate the course grade for a placement test score of x = 70.

```
> x=70
> print(b0+(b1*x))
Course_Grade
    65.48044
```

Construct a 95% PI (prediction interval) for a new observation of x = 70.

```
> predict(data_lm, newdata = data.frame(Placement_Test = 70), interval =
"prediction")
       fit      lwr      upr
1 65.48044 30.03062 100.9303
```