Employee Mental Health & Job Satisfaction

Select the following packages

```r
library(readxl)
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v lubridate 1.9.4     v tibble    3.2.1
## v purrr     1.0.4     v tidyr     1.3.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

Part 1

##. A. Mean, median, standard deviation, quantiles, min/max of numerical variables

```r
data <- read_excel("Group9_EmployeeMentalHealth_JobSatisfaction (1).xlsx")
selected_columns <- data[, c("Age", "Work_Experience", "Weekly_Work_Hours", "Stress_Level", "Work_Life_

# Compute summary statistics
summary_stats <- data.frame(
  Mean = sapply(selected_columns, mean),
  Median = sapply(selected_columns, median),
  Std_Dev = sapply(selected_columns, sd),
  Min = sapply(selected_columns, min),
  Q1 = sapply(selected_columns, function(x) quantile(x, 0.25)),
  Q3 = sapply(selected_columns, function(x) quantile(x, 0.75)),
  Max = sapply(selected_columns, max)
)

summary_stats
```

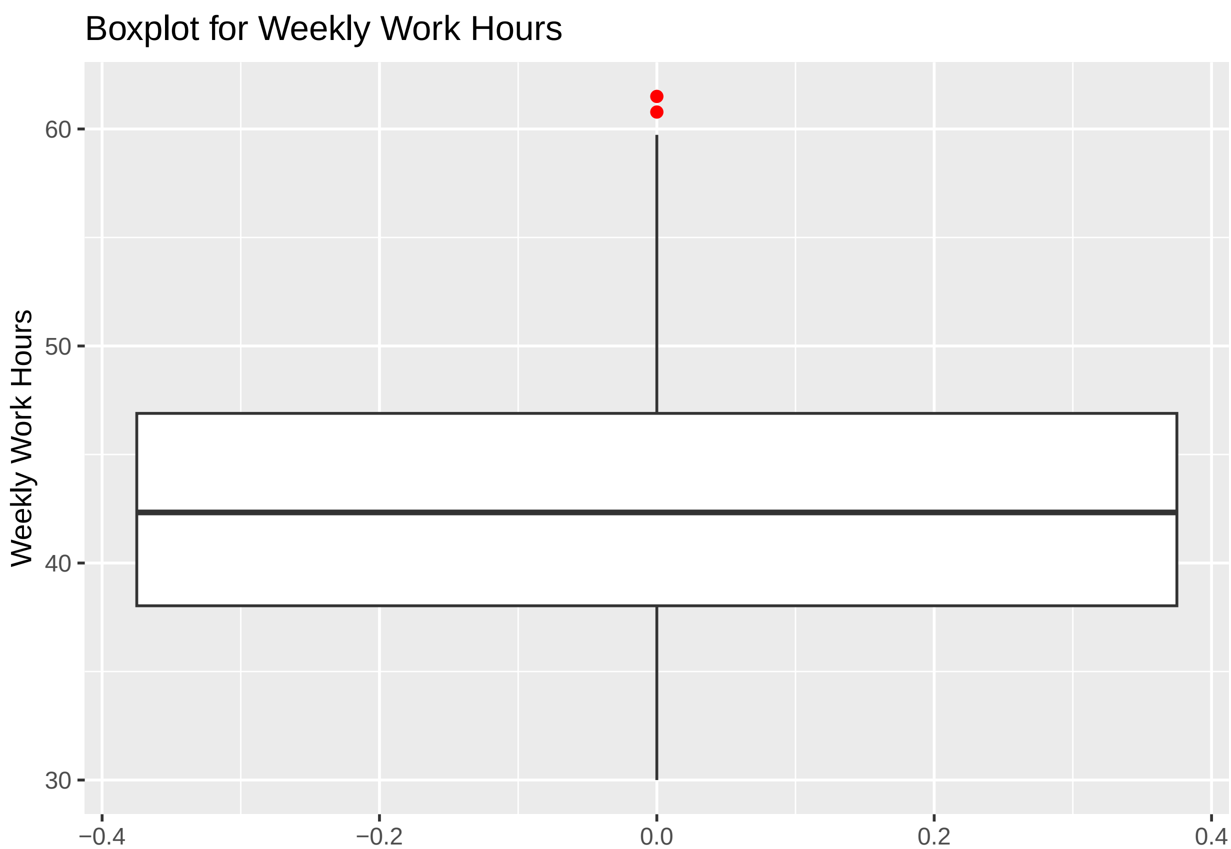
```
##                          Mean Median    Std_Dev   Min      Q1      Q3   Max
## Age                  44.11600 45.000 12.7332169 22.00 33.0000 54.0000 65.00
## Work_Experience      22.89400 24.000 12.4120458  1.00 12.0000 33.2500 40.00
## Weekly_Work_Hours    42.70618 42.330  6.4251079 30.00 38.0300 46.8950 61.50
## Stress_Level          2.19306  1.495  1.5302024  1.00  1.0000  3.1075  9.39
## Work_Life_Balance     7.78796  8.170  1.7412376  1.00  6.8475  9.0925 10.00
## Company_Culture_Score 6.95306  6.895  1.7344846  4.00  5.4725  8.4675  9.97
## Job_Satisfaction      9.98376 10.000  0.1666247  7.13 10.0000 10.0000 10.00
```

B. Create boxplots for each numerical variable.

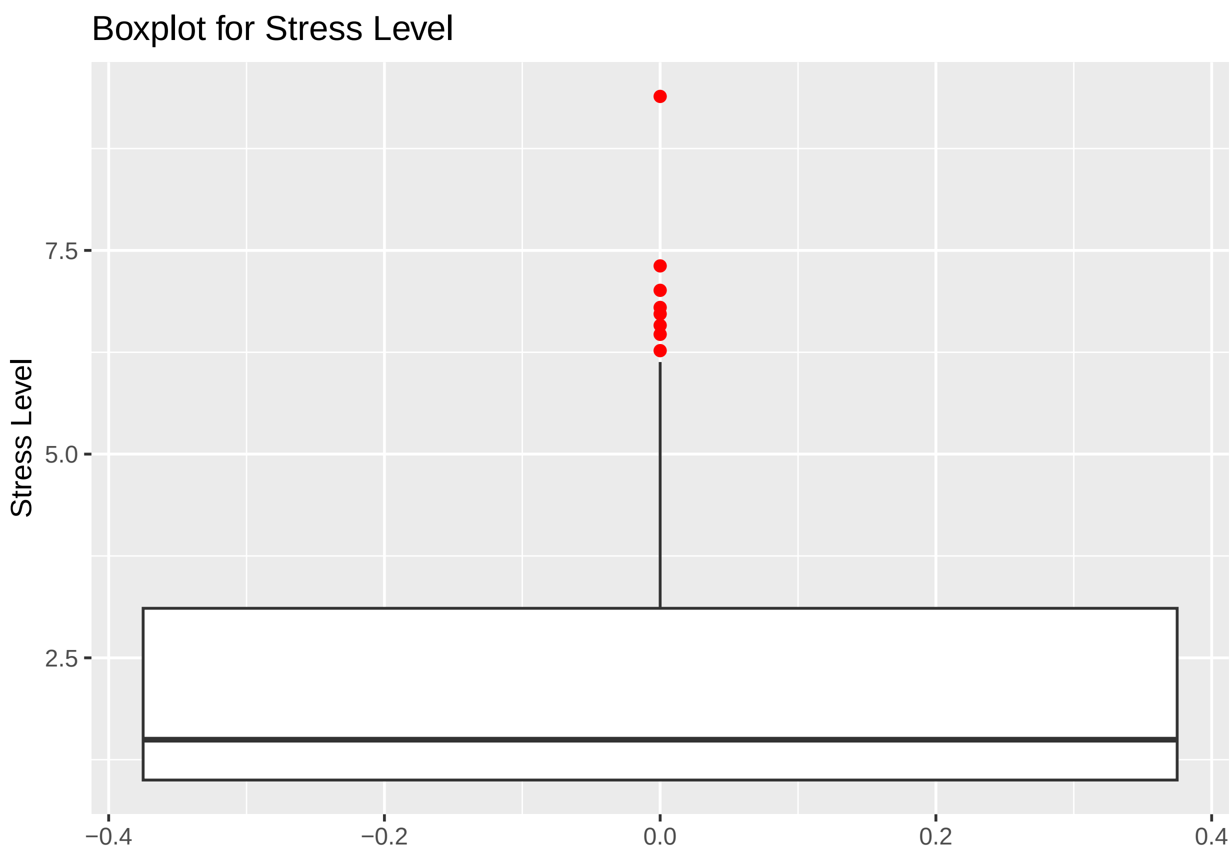
```r
ggplot(data, aes(y = Age)) +
  geom_boxplot(outlier.colour = "red", outlier.shape = 16, outlier.size = 2) +
  labs(title = "Boxplot for Age", y = "Ages")
```


```r
ggplot(data, aes(y = Work_Experience)) +
  geom_boxplot(outlier.colour = "red", outlier.shape = 16, outlier.size = 2) +
  labs(title = "Boxplot for Work Experience", y = "Work Experience (years)")
```

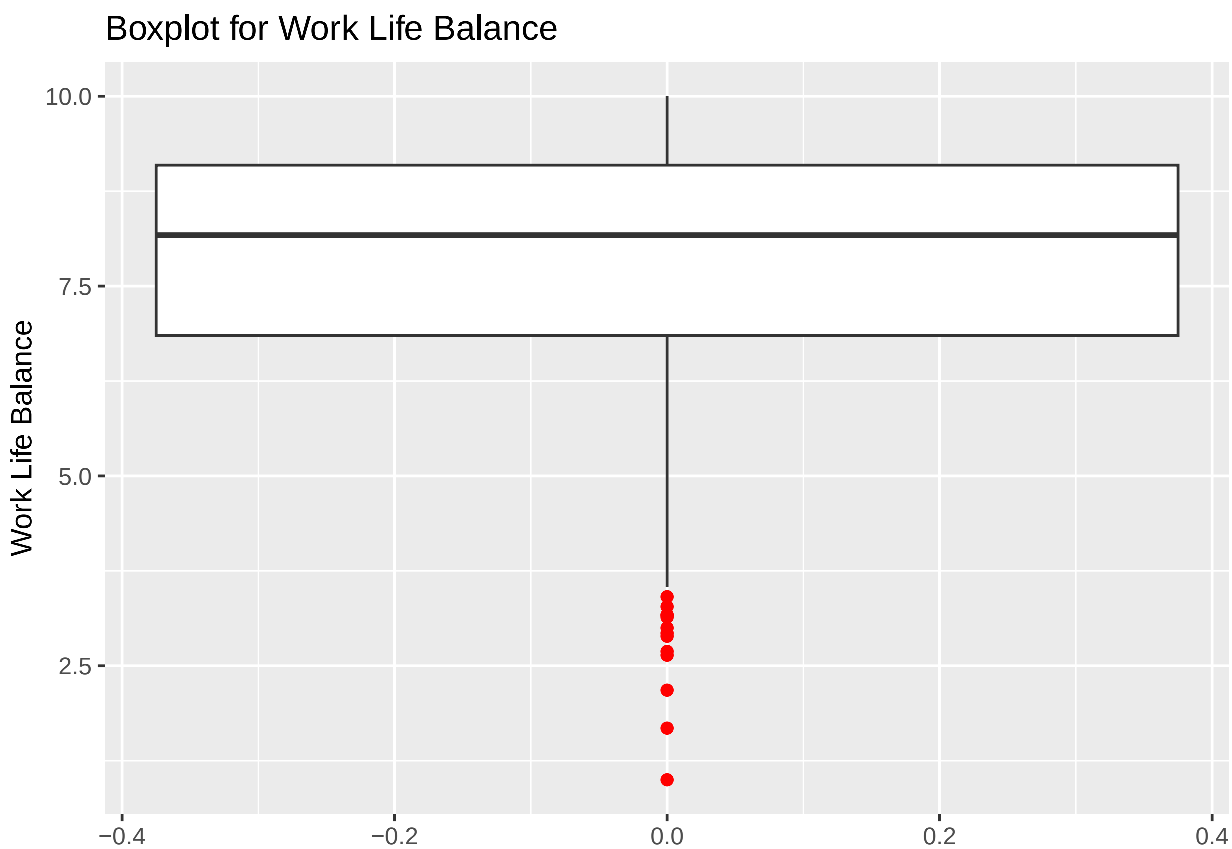

```r
ggplot(data, aes(y = Weekly_Work_Hours)) +
  geom_boxplot(outlier.colour = "red", outlier.shape = 16, outlier.size = 2) +
  labs(title = "Boxplot for Weekly Work Hours", y = "Weekly Work Hours")
```



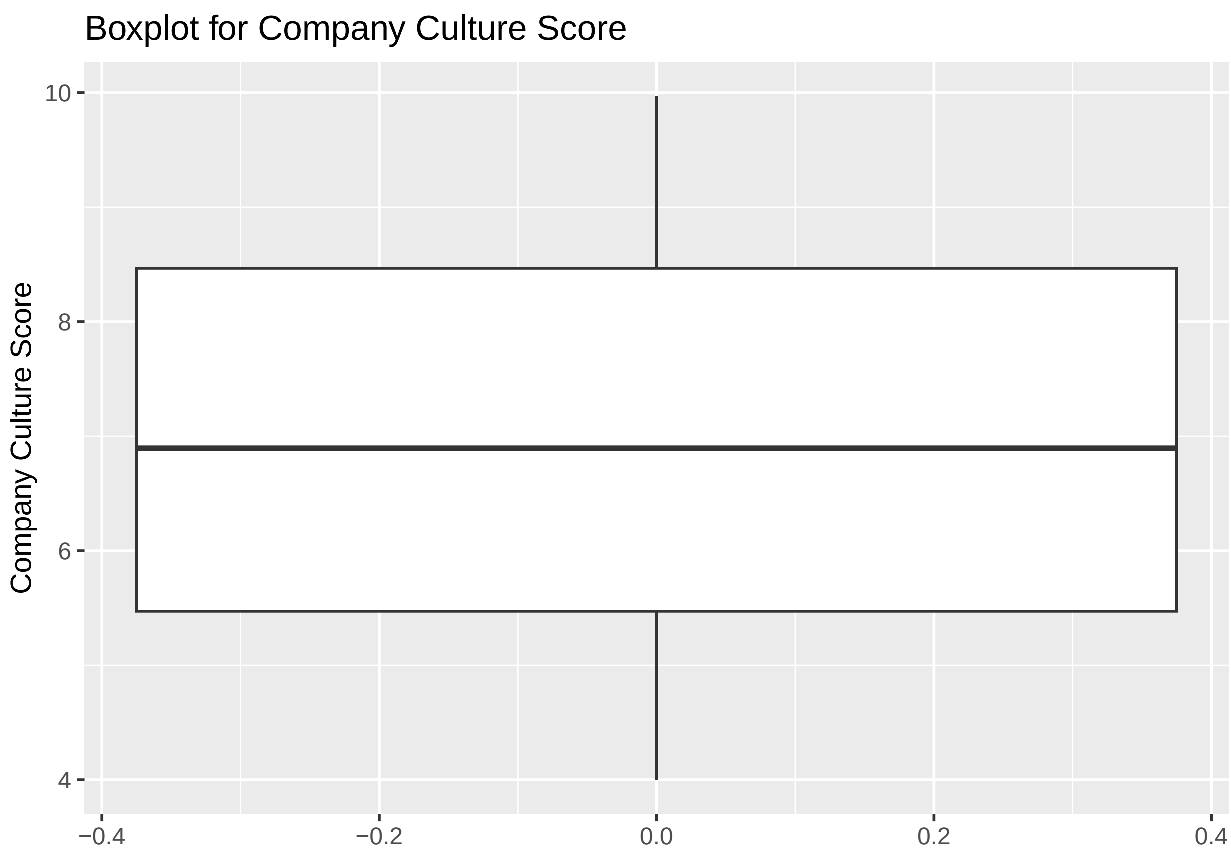

```r
ggplot(data, aes(y = Stress_Level)) +
  geom_boxplot(outlier.colour = "red", outlier.shape = 16, outlier.size = 2) +
  labs(title = "Boxplot for Stress Level", y = "Stress Level")
```



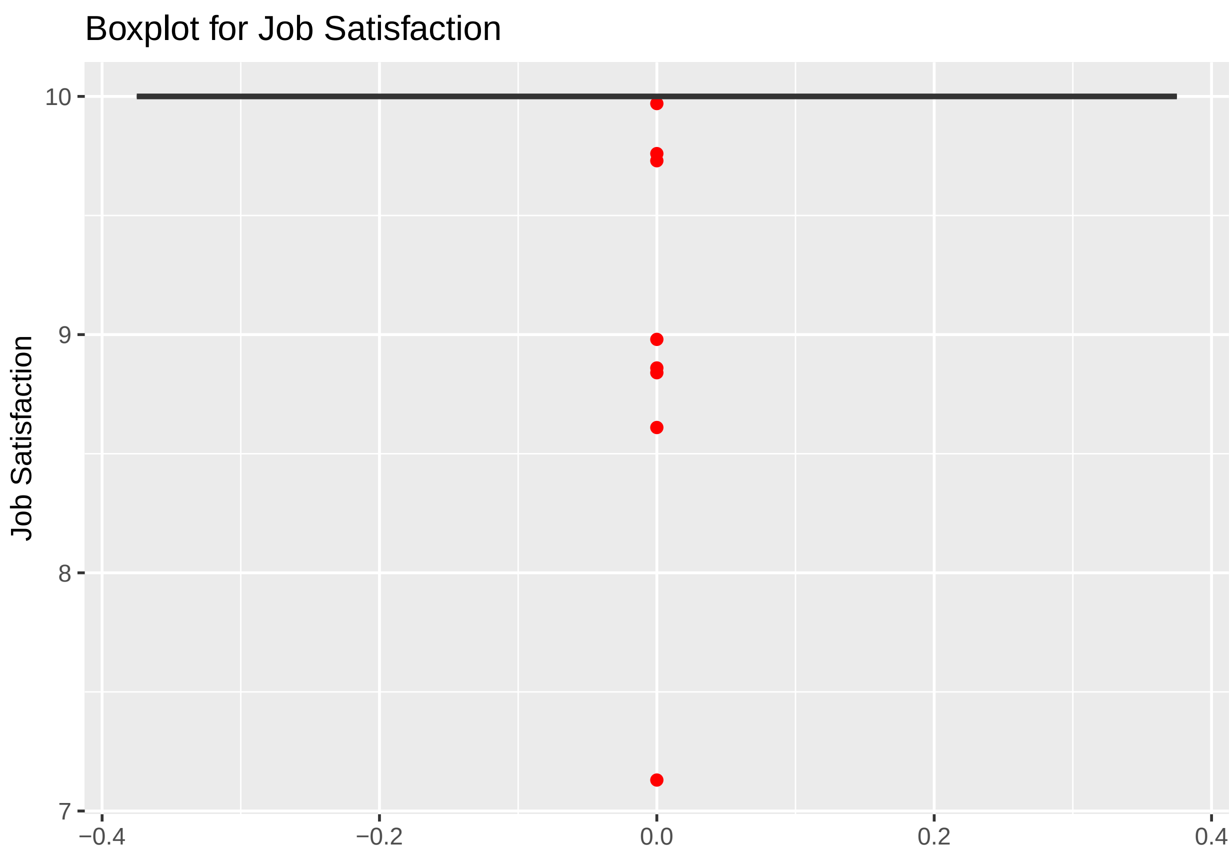

```r
ggplot(data, aes(y = Work_Life_Balance)) +
  geom_boxplot(outlier.colour = "red", outlier.shape = 16, outlier.size = 2) +
  labs(title = "Boxplot for Work Life Balance", y = "Work Life Balance")
```




```r
ggplot(data, aes(y = Company_Culture_Score)) +
  geom_boxplot(outlier.colour = "red", outlier.shape = 16, outlier.size = 2) +
  labs(title = "Boxplot for Company Culture Score", y = "Company Culture Score")
```




```r
ggplot(data, aes(y = Job_Satisfaction)) +
  geom_boxplot(outlier.colour = "red", outlier.shape = 16, outlier.size = 2) +
  labs(title = "Boxplot for Job Satisfaction", y = "Job Satisfaction")
```



Outliers are colored in red.

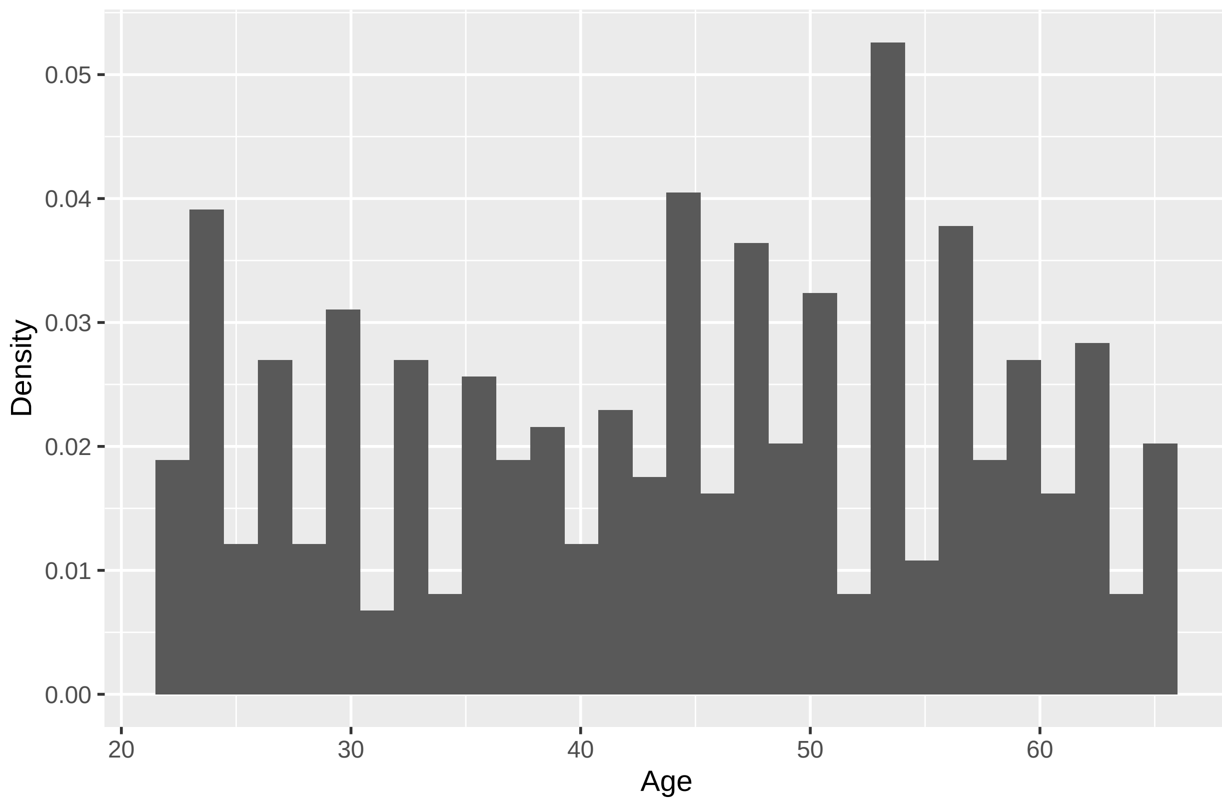
C. Normality Testing

First, let's inspect visually

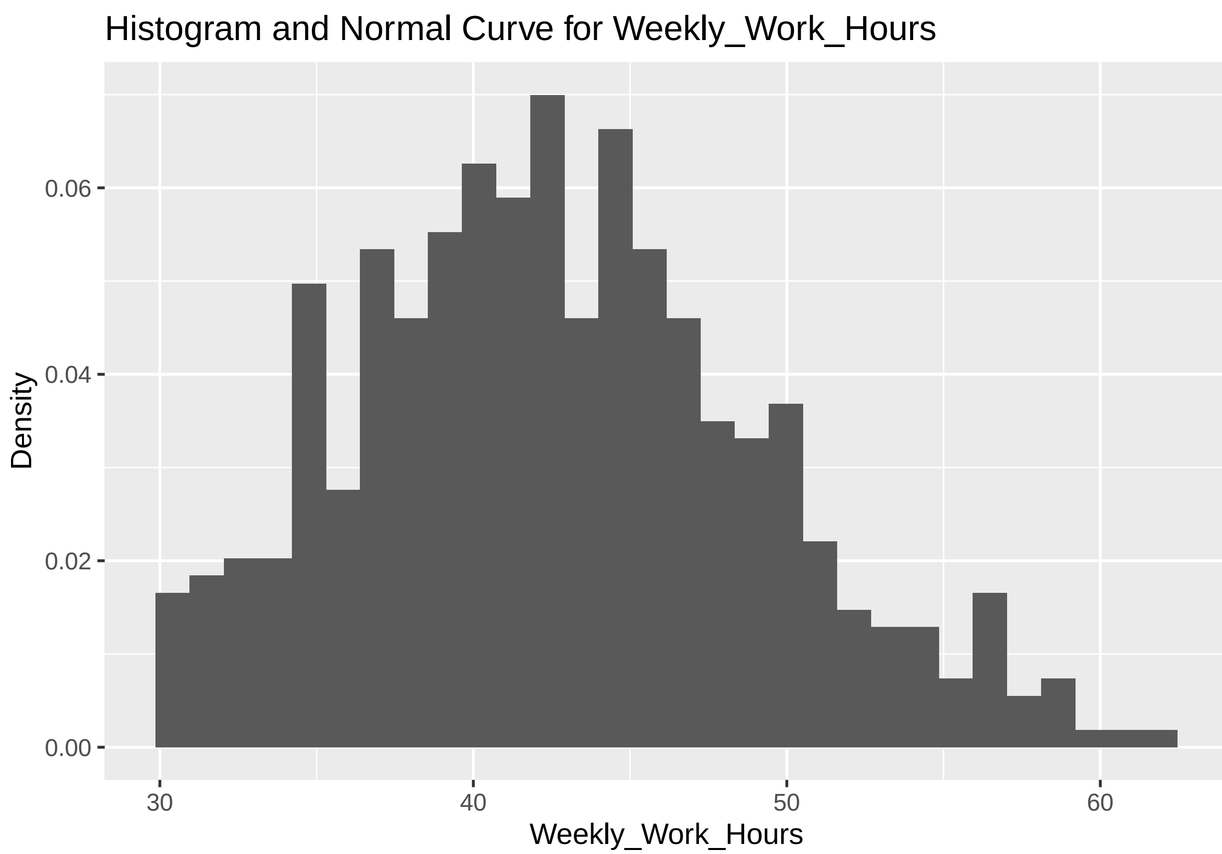
```
plot_normal_curve <- function(data, col_name) {
  ggplot(data, aes_string(x = col_name)) +
    geom_histogram(aes(y = after_stat(density)), bins = 30) +
    labs(title = paste("Histogram and Normal Curve for", col_name),
         x = col_name, y = "Density")
}

for (col in names(selected_columns)){print(plot_normal_curve(data, col))}
```

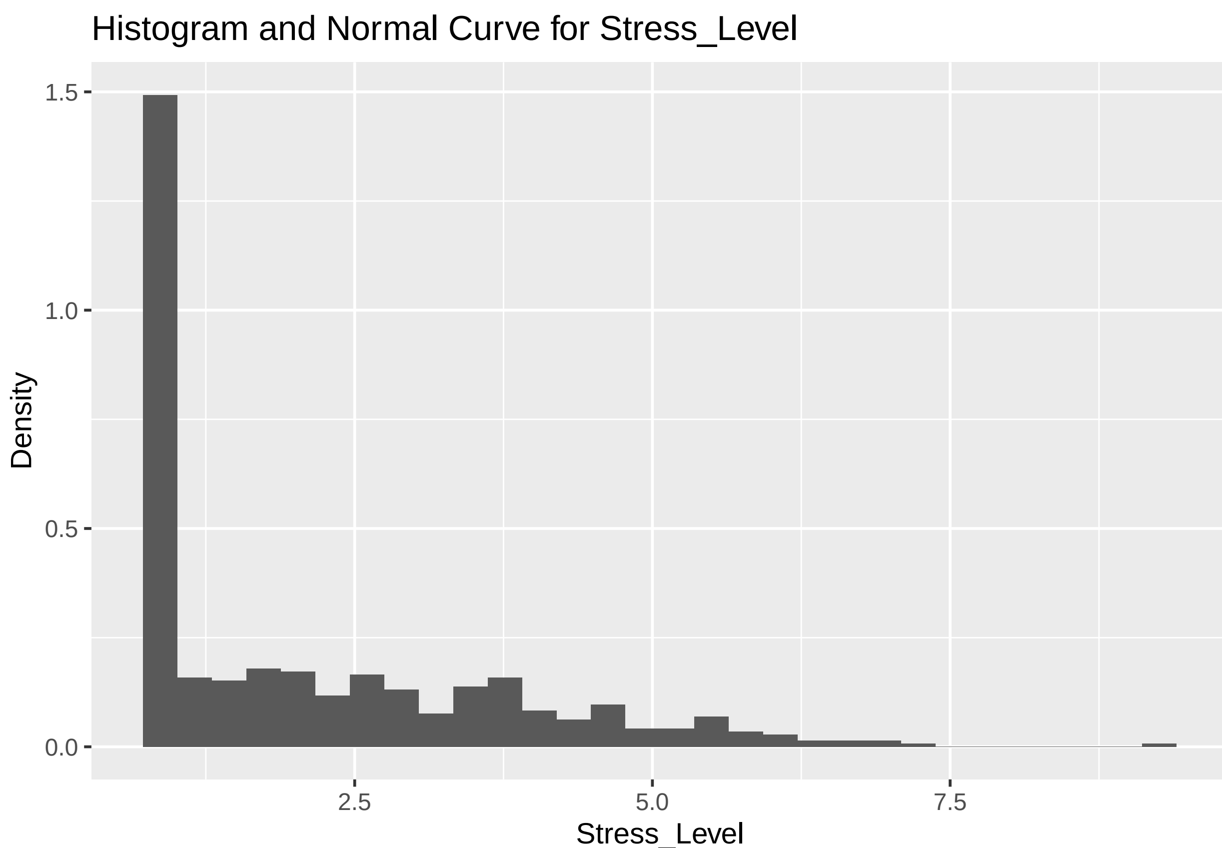
```
## Warning: `aes_string()` was deprecated in ggplot2 3.0.0.
## i Please use tidy evaluation idioms with `aes()`.
## i See also `vignette("ggplot2-in-packages")` for more information.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

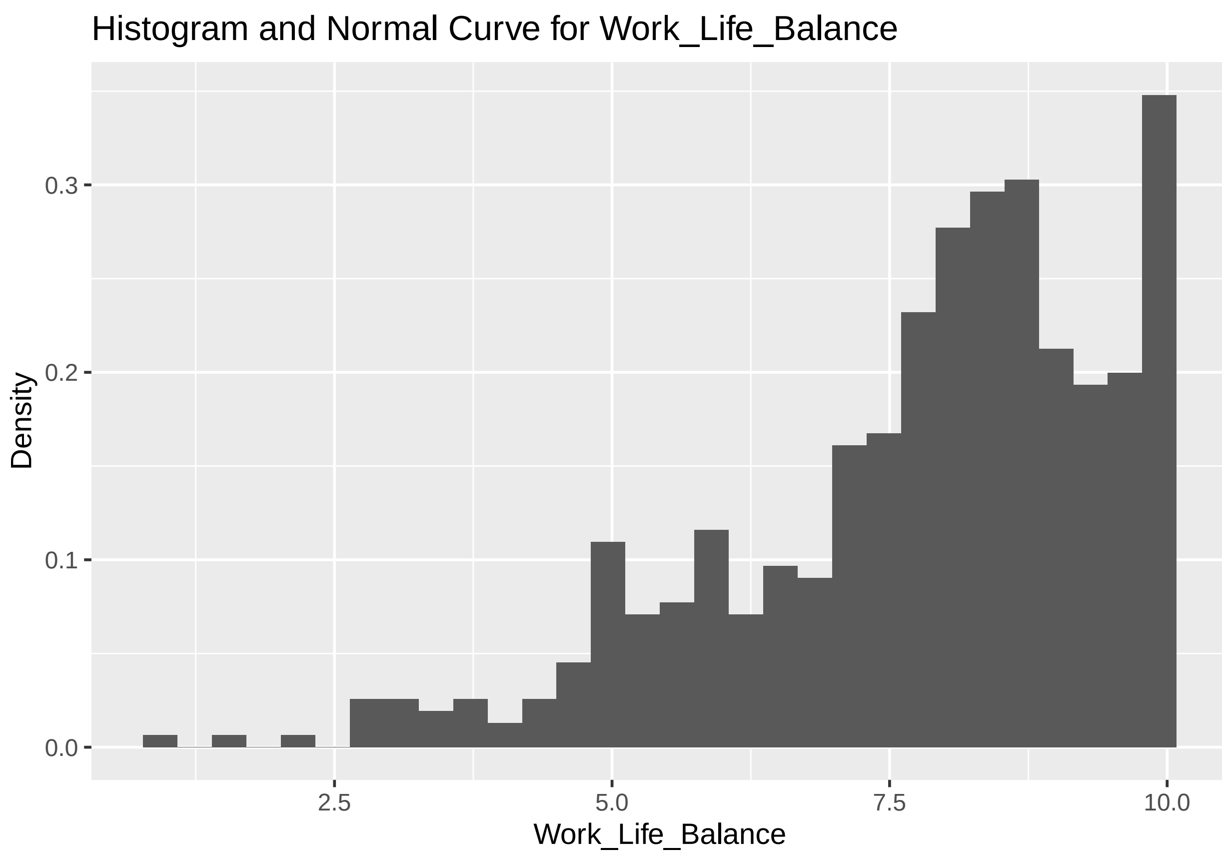
Histogram and Normal Curve for Age



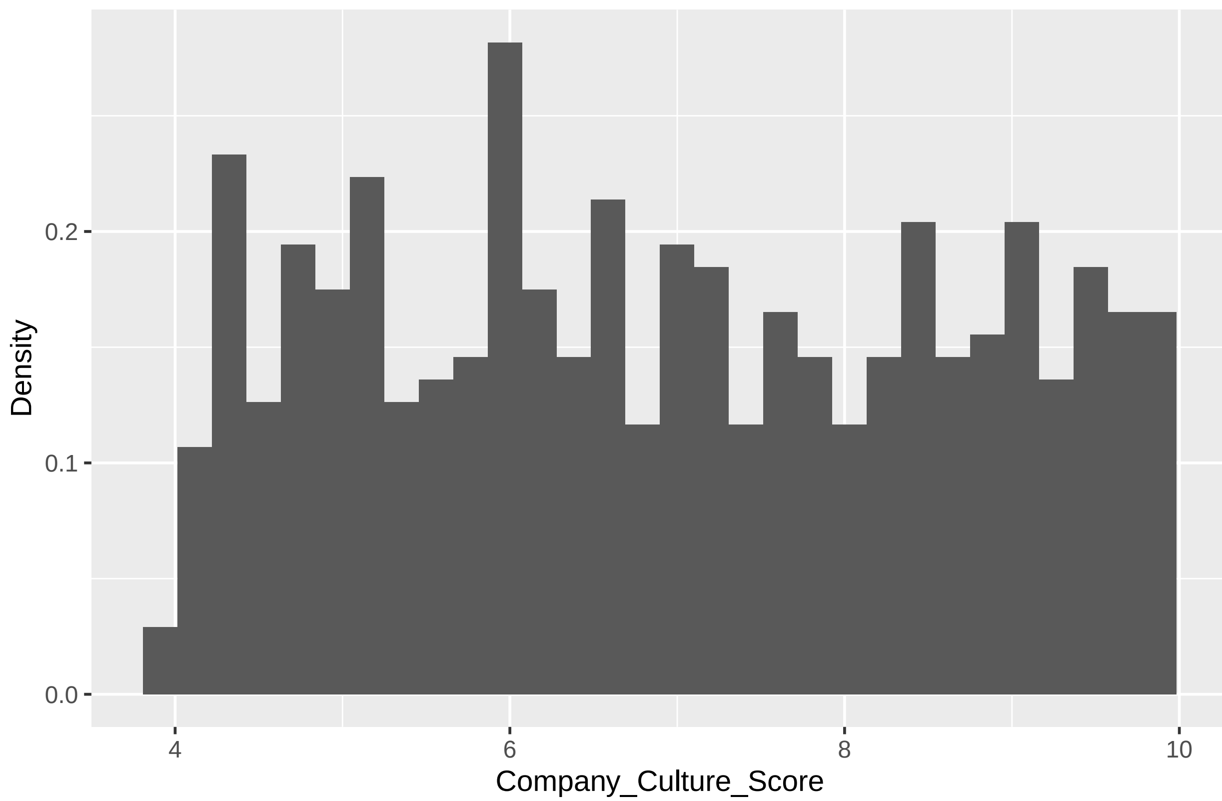

Histogram and Normal Curve for Work_Experience
Density
0.08
0.06
0.04
0.02
0.00
0
10
20
30
40
Work_Experience

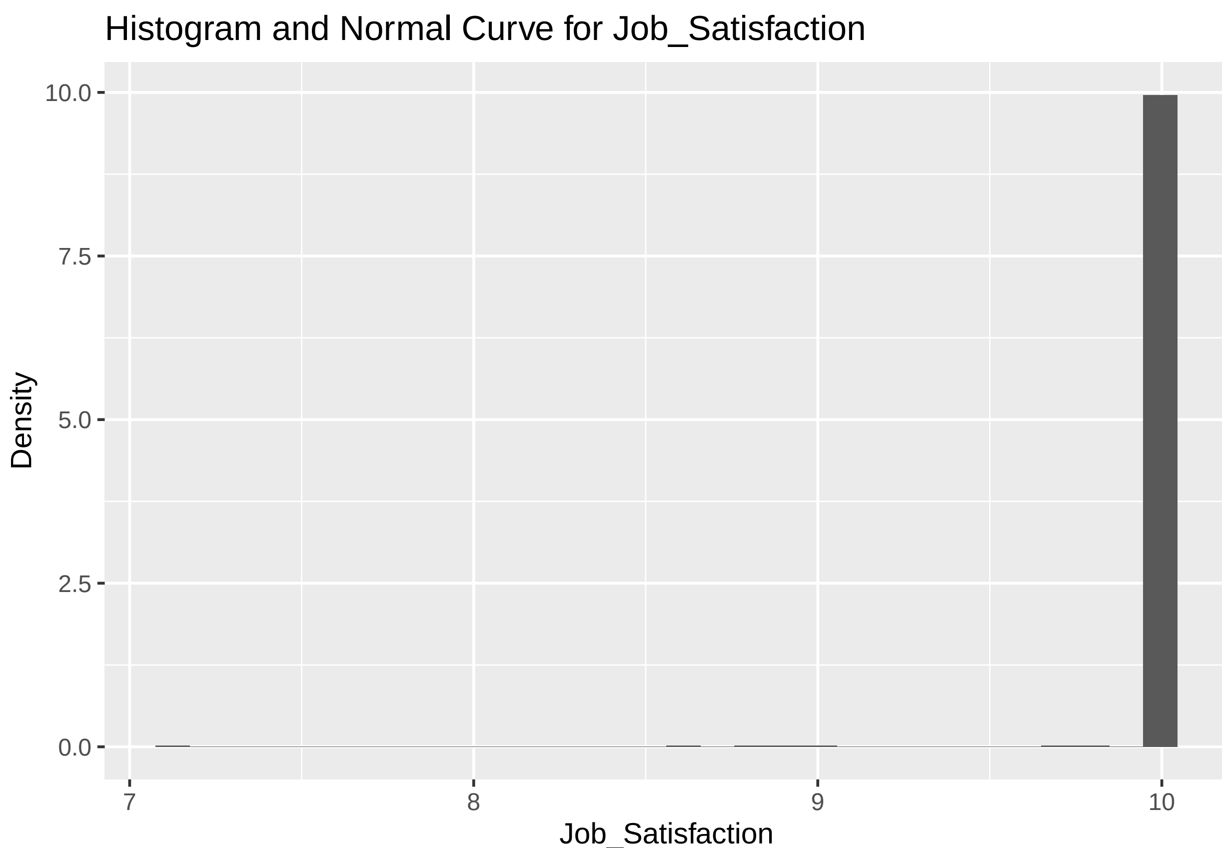

Histogram and Normal Curve for Weekly_Work_Hours
Density
0.06
0.04
0.02
0.00
30
40
50
60
Weekly_Work_Hours


Histogram and Normal Curve for Stress_Level
1.5
1.0
0.5
0.0
Density
2.5
5.0
7.5
Stress_Level


Histogram and Normal Curve for Work_Life_Balance
Density
0.0
0.1
0.2
0.3
2.5
5.0
7.5
10.0
Work_Life_Balance

Histogram and Normal Curve for Company_Culture_Score





Upon inspection, only the histogram for Weekly Work Hours looks normally distributed.

Now, let's test normality using the Shaprio-Wilk test. The null hypothesis for the Shapiro-Wilk test is that data "is" normally distributed.

```r
shapiro_test_results <- lapply(selected_columns, shapiro.test)
shapiro_test_results
```

```
## $Age
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.95187, p-value = 1.09e-11
##
##
## $Work_Experience
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.93356, p-value = 4.059e-14
##
##
## $Weekly_Work_Hours
##
##  Shapiro-Wilk normality test
```

```
##
## data:  X[[i]]
## W = 0.98772, p-value = 0.000324
##
##
## $Stress_Level
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.79145, p-value < 2.2e-16
##
##
## $Work_Life_Balance
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.92522, p-value = 4.498e-15
##
##
## $Company_Culture_Score
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.95249, p-value = 1.347e-11
##
##
## $Job_Satisfaction
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.07328, p-value < 2.2e-16
```

All of the variables have a very small p-value, which means we have to reject the null hypothesis and accept the alternative hypothesis, which is that data is not normally distributed.

D. Pearson's correlation analysis to determine relationships between numerical variables.

```r
correlation_matrix <- cor(selected_columns, method = "pearson")
correlation_matrix
```

```
##                                  Age Work_Experience Weekly_Work_Hours
## Age                      1.000000000     0.987086183        0.01744307
## Work_Experience          0.987086183     1.000000000        0.02139529
## Weekly_Work_Hours        0.017443066     0.021395294        1.00000000
## Stress_Level             0.008798213     0.007724859        0.54266322
## Work_Life_Balance       -0.015995840    -0.013348554       -0.45760676
```

```
## Company_Culture_Score  0.039178325     0.036543080        0.02054579
## Job_Satisfaction       0.045018754     0.053545371       -0.11285555
##                       Stress_Level Work_Life_Balance Company_Culture_Score
## Age                    0.008798213       -0.01599584            0.03917832
## Work_Experience        0.007724859       -0.01334855            0.03654308
## Weekly_Work_Hours      0.542663219       -0.45760676            0.02054579
## Stress_Level           1.000000000       -0.83785799            0.07744288
## Work_Life_Balance     -0.837857994        1.00000000           -0.08224776
## Company_Culture_Score  0.077442879       -0.08224776            1.00000000
## Job_Satisfaction      -0.340317728        0.31453687            0.04922892
##                       Job_Satisfaction
## Age                         0.04501875
## Work_Experience             0.05354537
## Weekly_Work_Hours          -0.11285555
## Stress_Level               -0.34031773
## Work_Life_Balance           0.31453687
## Company_Culture_Score       0.04922892
## Job_Satisfaction            1.00000000
```

E. Consider transforming or normalizing the data.

Let's try z-score normalization

```
z_scores <- sapply(selected_columns, function(x) {
  (x - mean(x)) / sd(x)
}, simplify = FALSE)

trans_shapiro_test_results <- lapply(z_scores, shapiro.test)
trans_shapiro_test_results
```

```
## $Age
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.95187, p-value = 1.09e-11
##
##
## $Work_Experience
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.93356, p-value = 4.059e-14
##
##
## $Weekly_Work_Hours
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.98772, p-value = 0.000324
##
```

```
##
## $Stress_Level
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.79145, p-value < 2.2e-16
##
##
## $Work_Life_Balance
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.92522, p-value = 4.498e-15
##
##
## $Company_Culture_Score
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.95249, p-value = 1.347e-11
##
##
## $Job_Satisfaction
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.07328, p-value < 2.2e-16
```

Even after z-score normalization, we can see how all the p-values are very small. This means that we have to reject the null hypothesis again, so once again, according to the Shapiro test, our transformed data is not normally distributed.

Part 2

Our goal now is to test whether employees in high-stress industries report lower job satisfaction scores than those in low-stress industries.

Since we failed to transform our data so that it follows normal distribution, we will use the Mann-Whitney U test, which does not have that as a requirement.

Null Hypothesis (H0): There is no relationship between stress level and job satisfaction. Alternative Hypothesis (H1): There is a relationship between stress level and job satisfaction.

```r
# Perform the Mann-Whitney U test (also known as Wilcoxon rank-sum test)
wilcox_test_results <- wilcox.test(data$Job_Satisfaction, data$Stress_Level)
wilcox_test_results
```

```
##
##  Wilcoxon rank sum test with continuity correction
```

```
##
## data:  data$Job_Satisfaction and data$Stress_Level
## W = 249994, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

Based on the results, we can conclude that there is a statistically significant difference in job satisfaction scores between individuals with high and low stress levels. The p-value 2.2e-16 is extremely small, so we accept the alternative hypothesis. 2.2e-16 is much, much smaller than alpha=0.05, so we our findings are definetely statistically significant at alpha=0.05

For the last step, let's get the correlation coefficient

```r
correlation_coefficient <- cor.test(data$Job_Satisfaction, data$Stress_Level, method = "pearson")
correlation_coefficient
```

```
##
##  Pearson's product-moment correlation
##
## data:  data$Job_Satisfaction and data$Stress_Level
## t = -8.0766, df = 498, p-value = 5.057e-15
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.4156055 -0.2603981
## sample estimates:
##        cor
## -0.3403177
```

Again, p-value = 5.057e-15 is extremely small so these are statistically significant results. The correlation coefficient is -0.34, meaning that for every unit increase in job stress level, job satisfaction decreases by 0.34.

Hence, to answer the question, yes, employees who report high-stress lower satisfaction scores.