

Install required packages

```
install.packages("ggplot2") install.packages("dplyr") install.packages("randomForest") install.packages("caTools")  
install.packages("openxlsx") # For reading Excel files
```

```
library(ggplot2) library(dplyr) library(randomForest) library(caTools) library(openxlsx)
```

Load the openxlsx library

```
library(openxlsx)
```

Load the dataset (ensure file path is correct)

```
data <- read.xlsx("C:/Users/haris/Desktop/Bike rental/1657875746_day.xlsx")
```

View the structure of the dataset

```
str(data)
```

```
install.packages("openxlsx") library(openxlsx)
```

```
#Performing Data Type Conversion
```

Convert data types

```
data$dteday <- as.Date(data$dteday) data$season <- as.factor(data$season) data$yr <- as.factor(data$yr) data$mnth <-  
as.factor(data$mnth) data$holiday <- as.factor(data$holiday) data$weekday <- as.factor(data$weekday)  
data$workingday <- as.factor(data$workingday) data$weathersit <- as.factor(data$weathersit)
```

Verify the changes

```
str(data)
```

```
#Missing Value Analysis
```

Check for missing values

```
sum(is.na(data))
```

Handle missing values if any (example using mean imputation)

```
data[is.na(data)] <- mean(data, na.rm = TRUE)
```

```
#Plot Monthly Distribution of Total Bikes Rented ggplot(data, aes(x = mnth, y = cnt, fill = season)) + geom_bar(stat =  
"summary", fun = "mean") + labs(title = "Monthly Distribution of Bike Rentals", x = "Month", y = "Average Bike  
Rentals") + theme_minimal()
```

```
#Plot Yearly Distribution of Total Bikes Rented ggplot(data, aes(x = yr, y = cnt, fill = yr)) + geom_bar(stat = "summary",
fun = "mean") + labs(title = "Yearly Distribution of Bike Rentals", x = "Year", y = "Average Bike Rentals") +
theme_minimal()

#Outliers Analysis using Boxplot ggplot(data, aes(x = season, y = cnt, fill = season)) + geom_boxplot() + labs(title =
"Outliers Analysis by Season", x = "Season", y = "Bike Rentals") + theme_minimal()

#Split Dataset into Train and Test set.seed(123) # For reproducibility split <- sample.split(data$cnt, SplitRatio = 0.7)
train <- subset(data, split == TRUE) test <- subset(data, split == FALSE)

#Building Random Forest Model
```

Train the model

```
rf_model <- randomForest(cnt ~ temp + atemp + hum + windspeed + season + weathersit + yr + holiday + workingday,
data = train, ntree = 500)
```

View the model summary

```
print(rf_model)
```

```
#Predict on Test Data
```

Predict on the test set

```
predictions <- predict(rf_model, test)
```

Evaluate the model

```
mse <- mean((predictions - test$cnt)^2) cat("Mean Squared Error: ", mse, "\n")
```

Plot actual vs predicted

```
ggplot(test, aes(x = cnt, y = predictions)) + geom_point(color = "blue") + geom_abline(intercept = 0, slope = 1, linetype
= "dashed", color = "red") + labs(title = "Actual vs Predicted Rentals", x = "Actual Rentals", y = "Predicted Rentals") +
theme_minimal()
```