DATE :10/09/2022

COURSE NAME:DATA WAREHOUSING AND DATA MINING FOR MEDICAL APPLICATIONS

COURSE CODE: CSA1654

NAME OF THE STUDENT: M.OOHA

REG NO: 192111142

DEPARTMENT :CSE

EXPERIMENT:1

@relation playing

@attribute condition {sunny,cloudy,raining}

@attribute temperature numeric

@attribute class{yes,no}

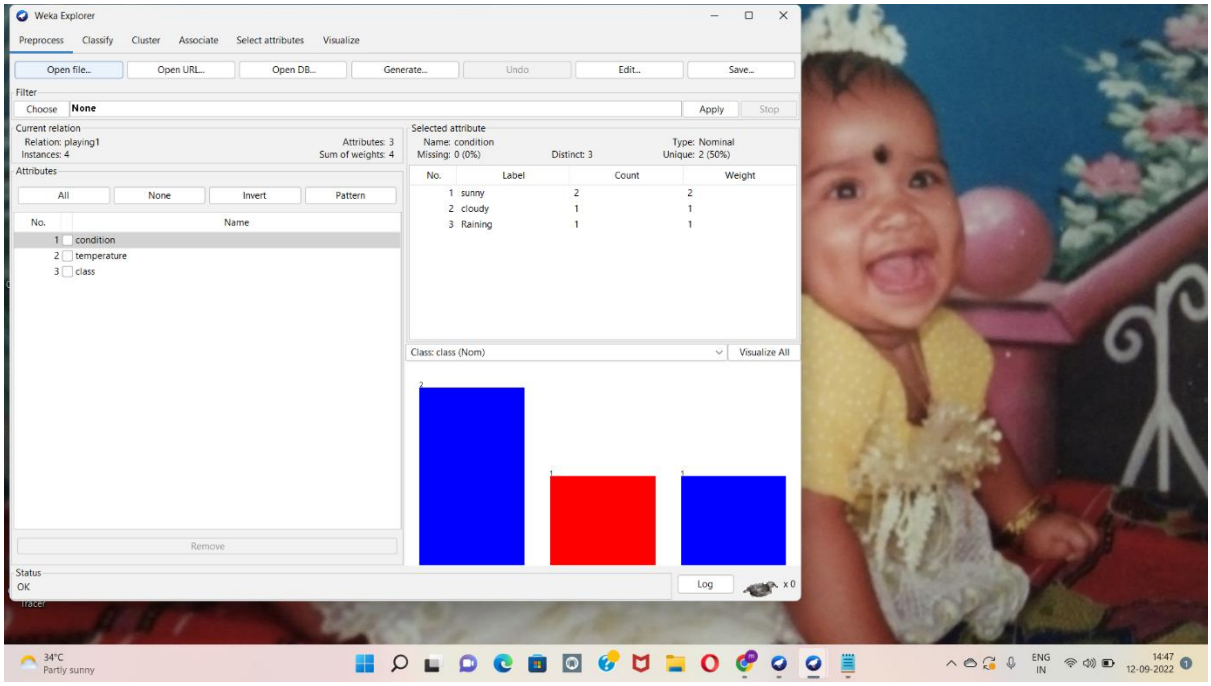
@data

sunny 85 yes

sunny 76 yes

cloudy 95 no

raining 87 yes



EXPERIMENT:2

MODE

Ctrl+Enter

R

Run

Save

```r
getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}

# Create the vector with numbers.
v <- c(2,1,2,3,1,2,3,4,1,5,5,3,2,3)

# Calculate the mode using the user function.
result <- getmode(v)
print(result)

# Create the vector with characters.
charv <- c("o","it","the","it","it")

# Calculate the mode using the user function.
result <- getmode(charv)
print(result)
```

Program input

Output

```
[1] 2
[1] "it"

[Execution complete with exit code 0]
```

MEDIAN

Ctrl+Enter

R

Run

Save

```r
x <- c(12,7,3,4.2,18,2,54,-21,8,-5)

# Find the median.
median.result <- median(x)
print(median.result)
```

Program input

Output

```
[1] 5.6

[Execution complete with exit code 0]
```

MEAN

R

Run Save

```r
x <- c(12,7,3,4.2,18,2,54,-21,8,-5)

# Find Mean.
result.mean <- mean(x)
print(result.mean)

```

Program input

Output

```
[1] 8.22

[Execution complete with exit code 0]
```

EXPRIMENT:3

Enter a title...

R

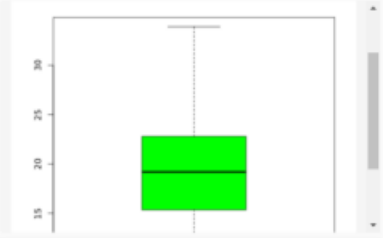
Run Save

```r
boxplot(mtcars$mpg, col="green")

```

Program input

Output



R

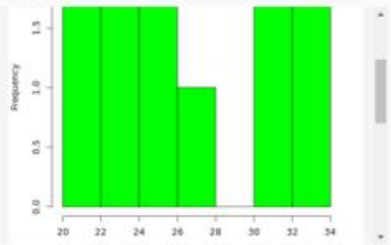
Run Save

```r
hist(subset(mtcars, cyl == 4)$mpg, col = "green")  ## Plot 1
hist(subset(mtcars, cyl == 8)$mpg, col = "green")  ## Plot 2

```

Program input

Output



```
with(mtcars, plot(mpg, qsec))
```

Output

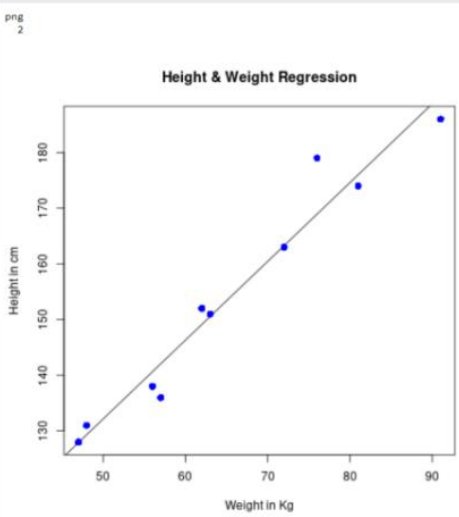
EXPERIMENT:4

```
# Create the predictor and response variable.
x <- c(151, 174, 138, 186, 128, 136, 179, 163, 152, 131)
y <- c(63, 81, 56, 91, 47, 57, 76, 72, 62, 48)
relation <- lm(y~x)

# Give the chart file a name.
png(file = "linearregression.png")

# Plot the chart.
plot(y,x,col = "blue",main = "Height & Weight Regression",
abline(lm(x~y)),cex = 1.3,pch = 16,xlab = "Weight in Kg",ylab = "Height in cm")

# Save the file.
dev.off()
```




```
# Generate random IQ values with mean = 30 and sd =2
IQ <- rnorm(40, 30, 2)

# Sorting IQ level in ascending order
IQ <- sort(IQ)

# Generate vector with pass and fail values of 40 students
result <- c(0, 0, 0, 1, 0, 0, 0, 0, 0, 1,
1, 0, 0, 0, 1, 1, 0, 0, 1, 0,
0, 0, 1, 0, 0, 1, 1, 0, 1, 1,
1, 1, 1, 0, 1, 1, 1, 1, 0, 1)

# Data Frame
df <- as.data.frame(cbind(IQ, result))

# Print data frame
print(df)

# output to be present as PNG file
png(file="LogisticRegressionGFG.png")

# Plotting IQ on x-axis and result on y-axis
plot(IQ, result, xlab = "IQ Level",
ylab = "Probability of Passing")

# Create a logistic model
g = glm(result~IQ, family=binomial, df)
```

Output

```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -12.5774     5.6739  -2.217   0.0266 *
IQ            0.4264     0.1941   2.196   0.0281 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.'
0.1 ' ' 1

(Dispersion parameter for binomial family taken to be
1)

    Null deviance: 55.352  on 39  degrees of freedom
Residual deviance: 48.909  on 38  degrees of freedom
AIC: 52.909
```