# Q1.Solve the questions in R programming using the Diamond Price dataset

In [ ]:
```
install.packages(c("ggplot2", "dplyr", "tidyverse", "caret", "cluster", "future"
library(ggplot2)
library(dplyr)
library(tidyverse)
library(caret)
library(cluster)
library(future)
library(foreach)
library(e1071)
```

Loading Diamond Price Dataset

In [2]:
```
diamond_data <- read.csv("https://raw.githubusercontent.com/mwaskom/seaborn-data
```

1. What are the different columns in the dataset?

In [3]:
```
colnames(diamond_data)
```

'carat' · 'cut' · 'color' · 'clarity' · 'depth' · 'table' · 'price' · 'x' · 'y' · 'z'

2. How many rows and columns are there?

In [4]:
```
dim(diamond_data)
```

53940 · 10

3. Remove the missing values

In [5]:
```
diamond_data <- na.omit(diamond_data)
```

4. What are the data types of each column?

In [6]:
```
str(diamond_data)
```

```
'data.frame':   53940 obs. of  10 variables:
 $ carat  : num  0.23 0.21 0.23 0.29 0.31 0.24 0.24 0.26 0.22 0.23 ...
 $ cut    : chr  "Ideal" "Premium" "Good" "Premium" ...
 $ color  : chr  "E" "E" "E" "I" ...
 $ clarity: chr  "SI2" "SI1" "VS1" "VS2" ...
 $ depth  : num  61.5 59.8 56.9 62.4 63.3 62.8 62.3 61.9 65.1 59.4 ...
 $ table  : num  55 61 65 58 58 57 57 55 61 61 ...
 $ price  : int  326 326 327 334 335 336 336 337 337 338 ...
 $ x      : num  3.95 3.89 4.05 4.2 4.34 3.94 3.95 4.07 3.87 4 ...
 $ y      : num  3.98 3.84 4.07 4.23 4.35 3.96 3.98 4.11 3.78 4.05 ...
 $ z      : num  2.43 2.31 2.31 2.63 2.75 2.48 2.47 2.53 2.49 2.39 ...
```

5. The average price of diamonds

In [7]:
```
mean(diamond_data$price)
```

3932.79972191324

6. The highest and lowest price recorded in the dataset

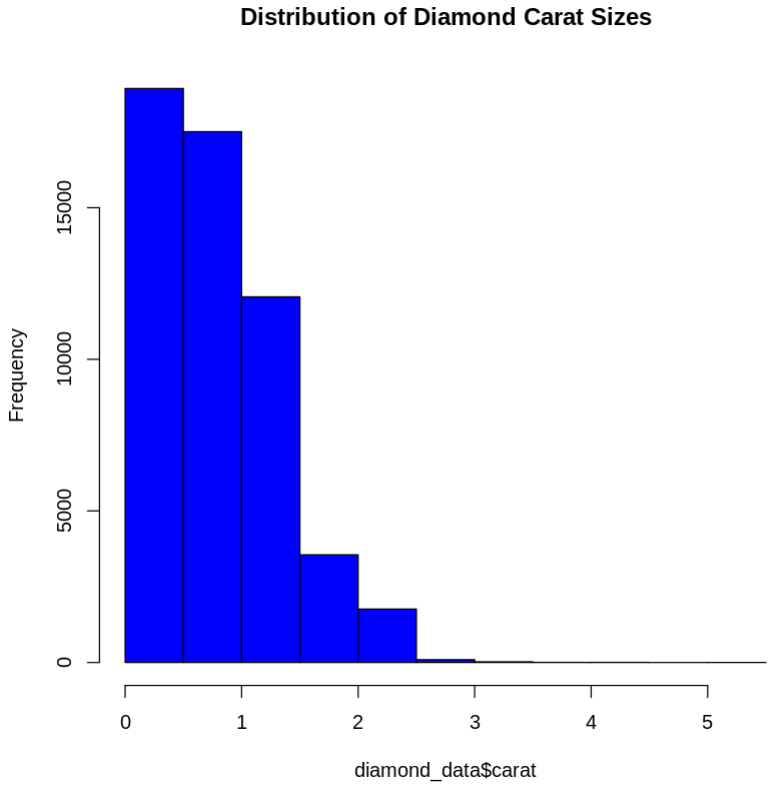
In [8]:
```
max(diamond_data$price)
min(diamond_data$price)
```

18823

326

7. The distribution of diamond carat sizes

In [9]:
```
hist(diamond_data$carat, main="Distribution of Diamond Carat Sizes", col="blue")
```



8. The correlation between carat and price

In [10]:
```
cor(diamond_data$carat, diamond_data$price)
```

0.921591301193477

9. Which cut type has the highest average price?

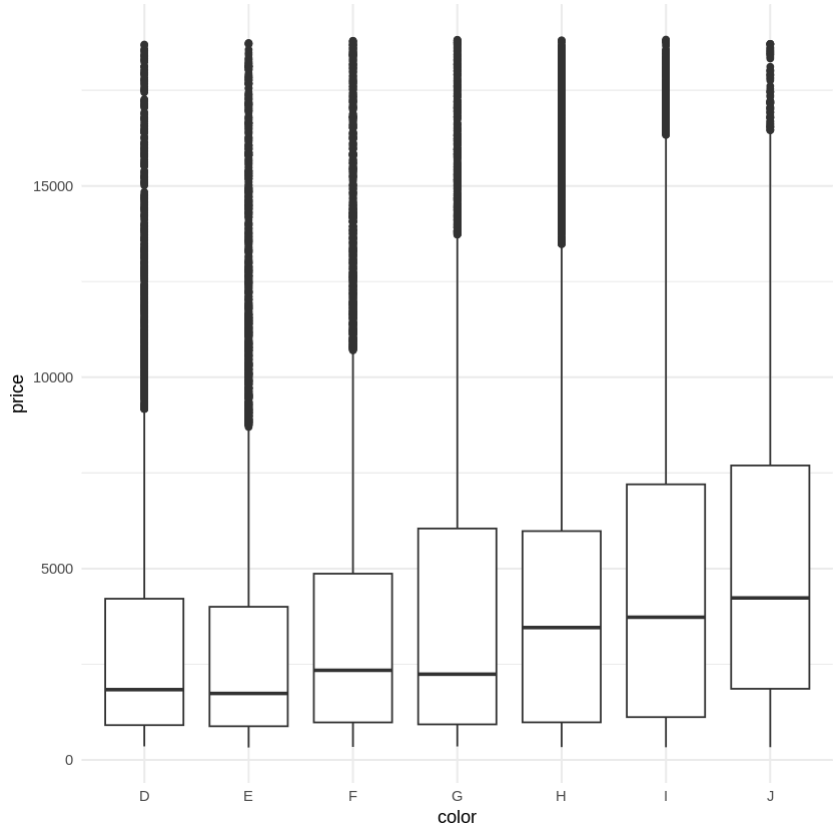
In [11]:
```
diamond_data %>% group_by(cut) %>% summarize(avg_price = mean(price)) %>% arrang
```

A tibble: 5 × 2

| cut | avg_price |
|---|---|
| <chr> | <dbl> |
| Premium | 4584.258 |
| Fair | 4358.758 |
| Very Good | 3981.760 |
| Good | 3928.864 |
| Ideal | 3457.542 |

10. How do diamond prices vary by color?

In [12]:
```
ggplot(diamond_data, aes(x=color, y=price)) + geom_boxplot() + theme_minimal()
```



11. The most common clarity level in the dataset

In [13]:
```
table(diamond_data$clarity)
```

```
   I1    IF   SI1   SI2   VS1   VS2  VVS1  VVS2 
  741  1790 13065  9194  8171 12258  3655  5066 
```

12. The percentage of diamonds belong to each cut category

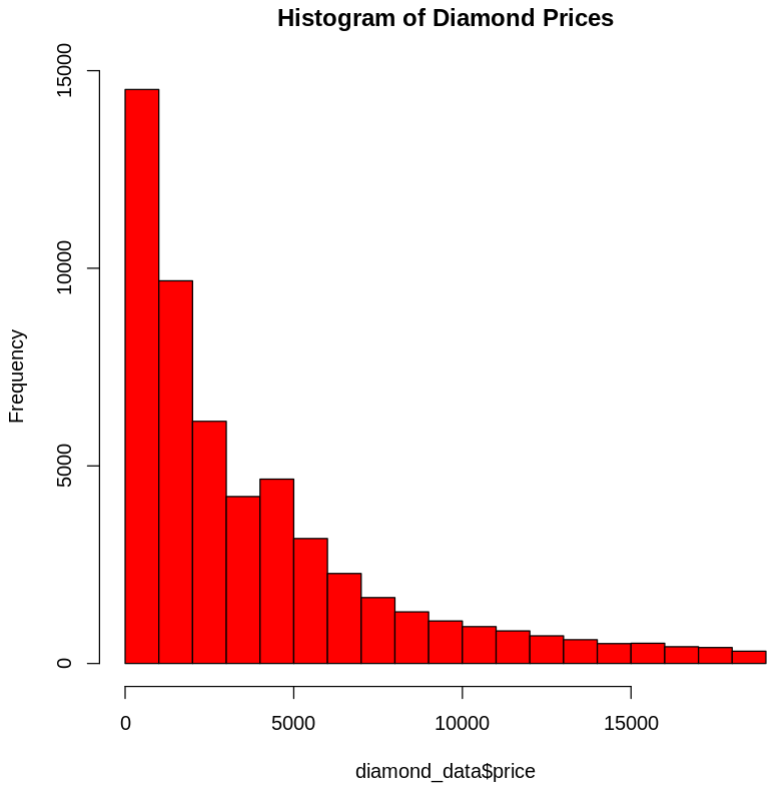
In [14]:
```
diamond_data %>% count(cut) %>% mutate(percentage = n / sum(n) * 100)
```

A data.frame: 5 × 3

| cut | n | percentage |
|---|---|---|
| <chr> | <int> | <dbl> |
| Fair | 1610 | 2.984798 |
| Good | 4906 | 9.095291 |
| Ideal | 21551 | 39.953652 |
| Premium | 13791 | 25.567297 |
| Very Good | 12082 | 22.398962 |

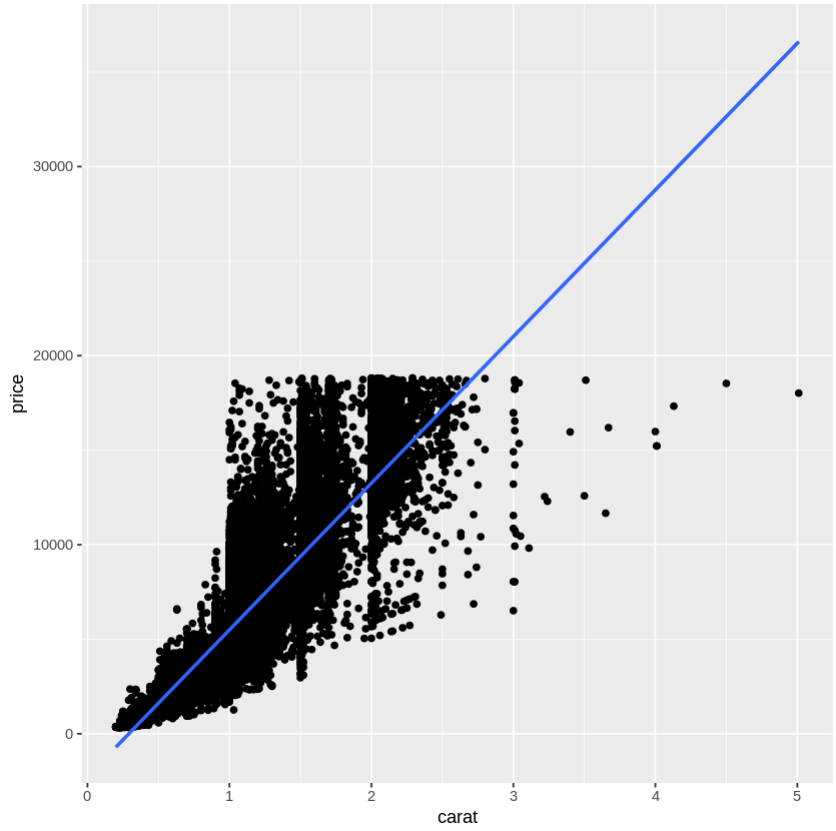
13. Plot a histogram of diamond prices

In [15]:
```
hist(diamond_data$price, main="Histogram of Diamond Prices", col="red")
```



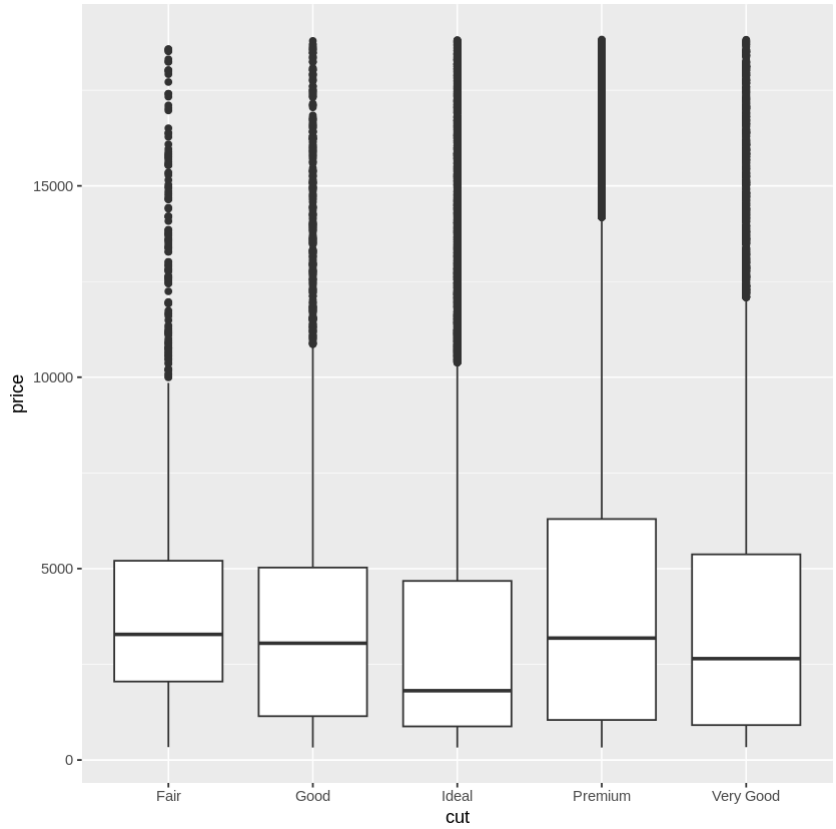
14. Visualize the relationship between carat and price using a scatter plot

In [23]:
```
ggplot(diamond_data, aes(x=carat, y=price)) + geom_point() + geom_smooth(formula
```



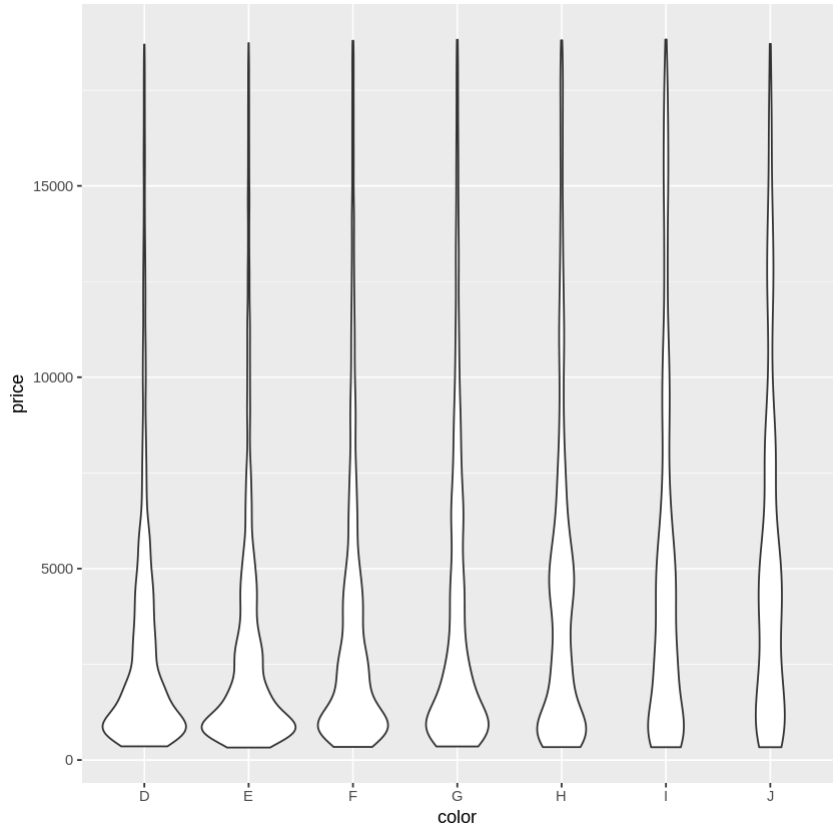

15. Create a boxplot of diamond prices across different cut types

In [17]:
```
ggplot(diamond_data, aes(x=cut, y=price)) + geom_boxplot()
```



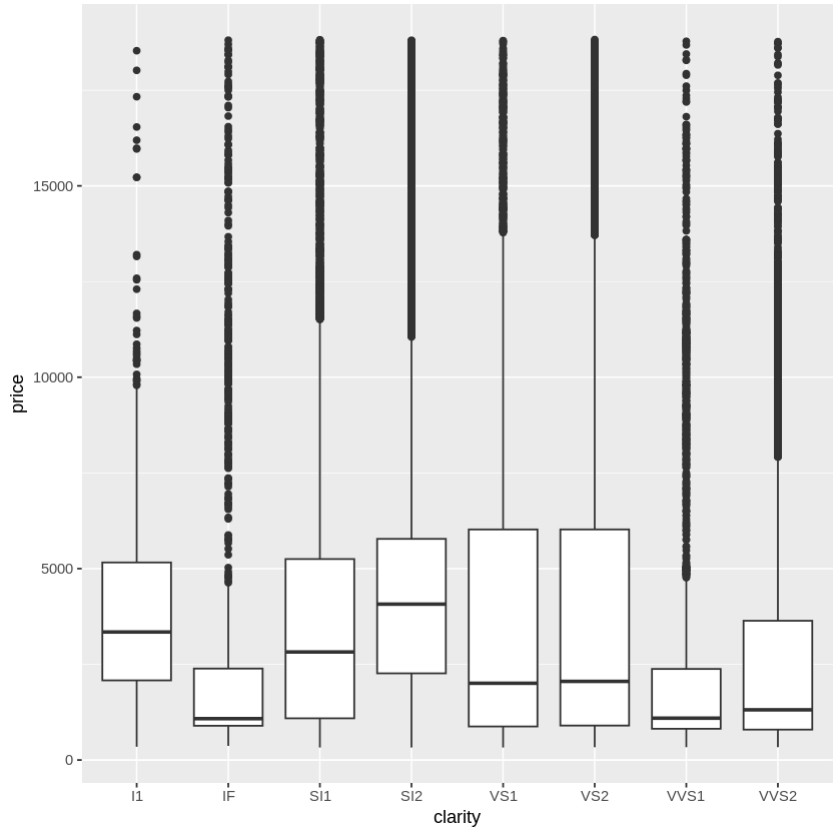

16. Is there a pattern in diamond price fluctuations based on color?

```
In [18]: ggplot(diamond_data, aes(x=color, y=price)) + geom_violin()
```




17. Visualize diamonds by clarity and their price distribution

In [19]:
```
ggplot(diamond_data, aes(x=clarity, y=price)) + geom_boxplot()
```



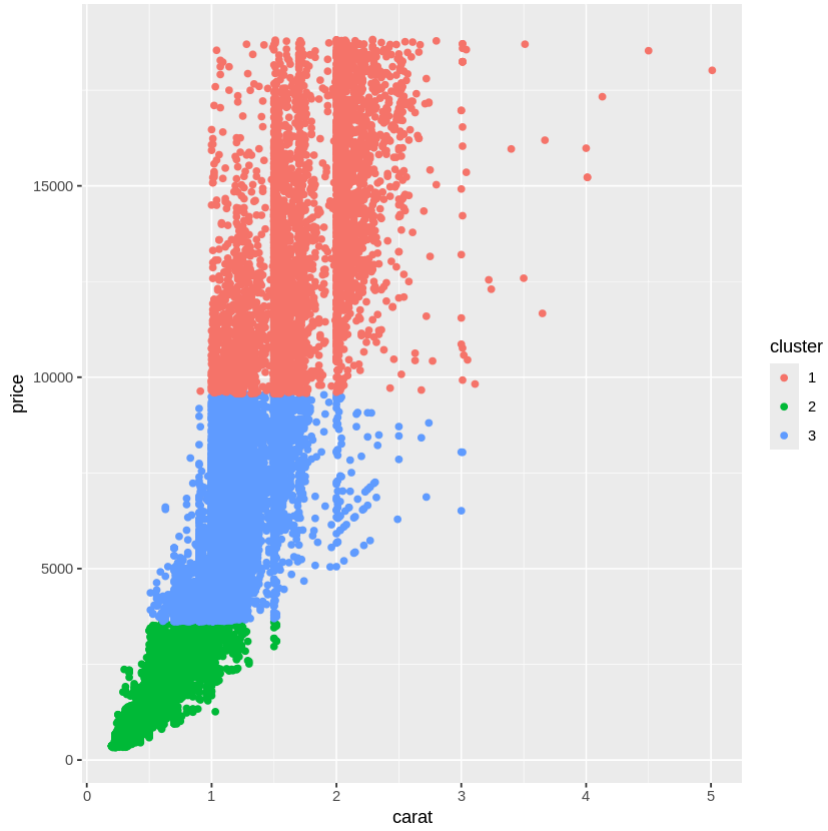

18. Predict the price of a diamond using linear regression based on carat, cut, clarity, and color

In [20]:
```
lm_model <- lm(price ~ carat + cut + clarity + color, data=diamond_data)
summary(lm_model)
```

```
Call:
lm(formula = price ~ carat + cut + clarity + color, data = diamond_data)

Residuals:
     Min       1Q   Median       3Q      Max
-16813.5   -680.4   -197.6    466.4  10394.9

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -7362.80      51.68 -142.46   <2e-16 ***
carat         8886.13      12.03  738.44   <2e-16 ***
cutGood        655.77      33.63   19.50   <2e-16 ***
cutIdeal       998.25      30.66   32.56   <2e-16 ***
cutPremium     869.40      30.93   28.11   <2e-16 ***
cutVery Good   848.72      31.28   27.14   <2e-16 ***
clarityIF     5419.65      52.14  103.95   <2e-16 ***
claritySI1    3573.69      44.60   80.13   <2e-16 ***
claritySI2    2625.95      44.79   58.63   <2e-16 ***
clarityVS1    4534.88      45.54   99.59   <2e-16 ***
clarityVS2    4217.83      44.84   94.06   <2e-16 ***
clarityVVS1   5072.03      48.21  105.20   <2e-16 ***
clarityVVS2   4967.20      46.89  105.93   <2e-16 ***
colorE        -211.68      18.32  -11.56   <2e-16 ***
colorF        -303.31      18.51  -16.39   <2e-16 ***
colorG        -506.20      18.12  -27.93   <2e-16 ***
colorH        -978.70      19.27  -50.78   <2e-16 ***
colorI       -1440.30      21.65  -66.54   <2e-16 ***
colorJ       -2325.22      26.72  -87.01   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1157 on 53921 degrees of freedom
Multiple R-squared:  0.9159,    Adjusted R-squared:  0.9159
F-statistic: 3.264e+04 on 18 and 53921 DF,  p-value: < 2.2e-16
```

19. Cluster the diamonds into different price groups using K-Means clustering

In [21]:
```
diamond_cluster <- kmeans(diamond_data$price, centers=3)
diamond_data$cluster <- as.factor(diamond_cluster$cluster)
ggplot(diamond_data, aes(x=carat, y=price, color=cluster)) + geom_point()
```




20. Features that contribute the most to predicting the diamond price?

In [22]:
```
varImp(lm_model)
```

A data.frame: 18 × 1

| | Overall |
|---|---|
| | <dbl> |
| **carat** | 738.43711 |
| **cutGood** | 19.49714 |
| **cutIdeal** | 32.56293 |
| **cutPremium** | 28.10755 |
| **cutVery Good** | 27.13545 |
| **clarityIF** | 103.95177 |
| **claritySI1** | 80.13173 |
| **claritySI2** | 58.63016 |
| **clarityVS1** | 99.59064 |
| **clarityVS2** | 94.06104 |
| **clarityVVS1** | 105.20509 |
| **clarityVVS2** | 105.93158 |
| **colorE** | 11.55732 |
| **colorF** | 16.38674 |
| **colorG** | 27.93292 |
| **colorH** | 50.78361 |
| **colorI** | 66.53814 |
| **colorJ** | 87.01342 |