TASK 2

In [ ]:
```
import pandas as pd

df = pd.read_excel("Vehicle_Data.xlsx")

print(df)
```

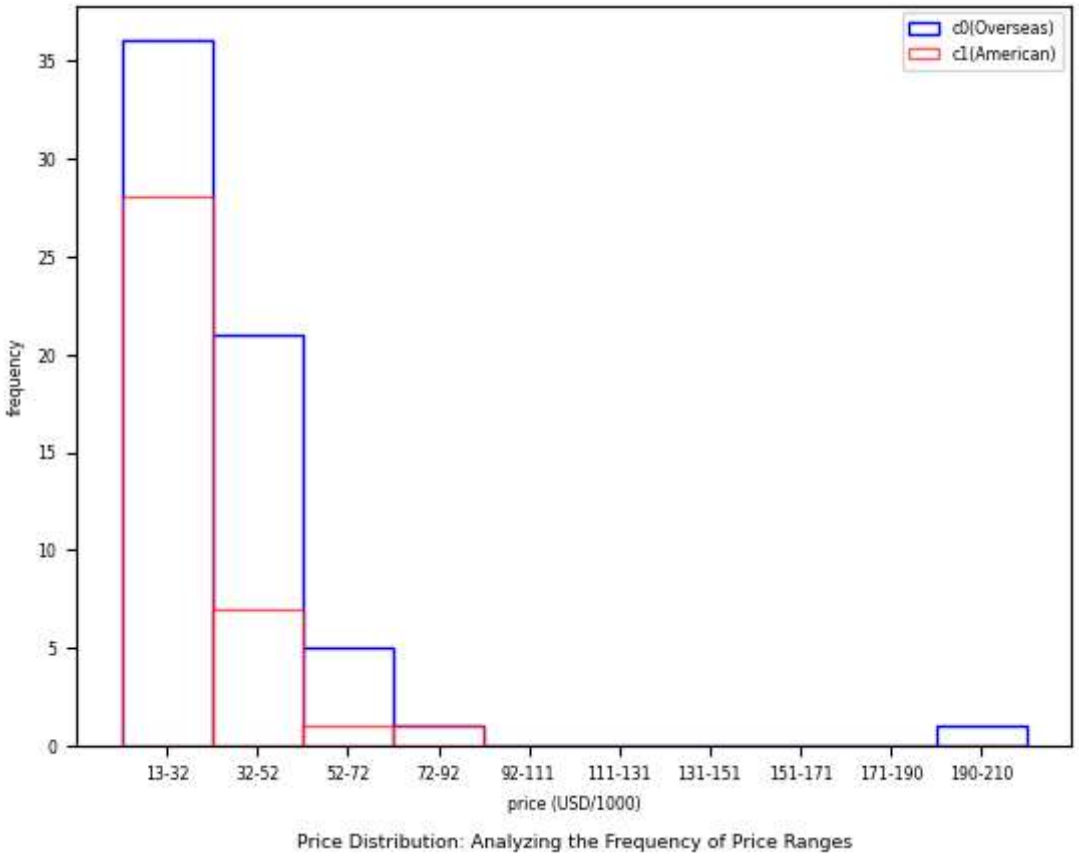
```
         Car Make   Price (USD)  Horsepower (hp)  Top Speed (mph)  \
0     Nissan Micra         17000              109              115
1    Nissan Sentra         23000              149              120
2      Nissan 370Z         30000              332              160
3      Nissan GT-R        210740              600              205
4     Infiniti Q50         53500              400              155
..             ...           ...              ...              ...
96        Kia Niro         24690              139              105
97      Kia Sedona         42000              276              130
98       Acura TLX         50000              355              155
99    Jeep patriot         28000              172              115
100   Chrysler 300         50000              363              140

    Category (class)
0           Overseas
1           Overseas
2           Overseas
3           Overseas
4           Overseas
..               ...
96          Overseas
97          Overseas
98          Overseas
99          American
100         American

[101 rows x 5 columns]
```

TASK 3

In [ ]:
```
import math

import matplotlib.pyplot as plt
fig, ax = plt.subplots()
plt.rcParams.update({'font.size': 5.6}) #Change the font size of the table to 6 for

import numpy as np
category = np.array(df.iloc[:, 4]) # Get an array of all values in Category column

## Separate the instances into their different categories

pos = 0 # position of the price in the dataframe
overseas = np.array([]) # Store the values of prices that fall under the Overseas c
american = np.array([]) # Store the values of prices that fall under the American c

for row in category:
```

```
    # Filter categories
    if row == "Overseas":
        overseas = np.append(overseas, df.iloc[pos, 1])
    else:
        american = np.append(american, df.iloc[pos, 1])
    pos += 1

## Create class ranges for the price attribute

# Statistics of Price column
price = np.array(df.iloc[:, 1]) # Get an array of all values in Price column
class_ranges = [] # store class ranges in bins format
max_price = np.max(price) # Minimum price
min_price = np.min(price) # Maximum price

# Class boundary calculations
n_classes = math.floor(math.pow(len(price), 1/2)) # Number of classes
interval_len = (max_price - min_price) / n_classes # Interval length

# Class boundaries
lcb = min_price # Lower class boundary
ucb = lcb + interval_len # Upper class boundary

for i in range(n_classes):
    # Adding class boundaries
    class_ranges.append(lcb)
    class_ranges.append(ucb)

    # Update class boundry
    lcb = ucb
    ucb = lcb + interval_len

# Make x labels math the class boundaries
xticks_val = [] # Store values of xticks
xticks_label = [] # Store labels name xtick

# Class boundaries
lcb = min_price # Lower class boundary
ucb = lcb + interval_len # Upper class boundary

for i in range(n_classes):
    # Position and name  x labels
    xticks_val.append(lcb + ((ucb - lcb) / 2)) # Current value of xtick
    xticks_label.append(str(math.floor(lcb/1000)) + "-" + str(math.floor(ucb/1000))

    # Update class boundry
    lcb = ucb
    ucb = lcb + interval_len


## Draw Histogram
plt.hist(overseas, bins=class_ranges, color='w', edgecolor='b', alpha=1.0, label='c
plt.hist(american, bins=class_ranges, color = 'w', edgecolor='r', alpha=0.65, label
plt.xticks(xticks_val, xticks_label)
plt.title("Price Distribution: Analyzing the Frequency of Price Ranges", y = -0.16)
plt.xlabel("price (USD/1000)")
```

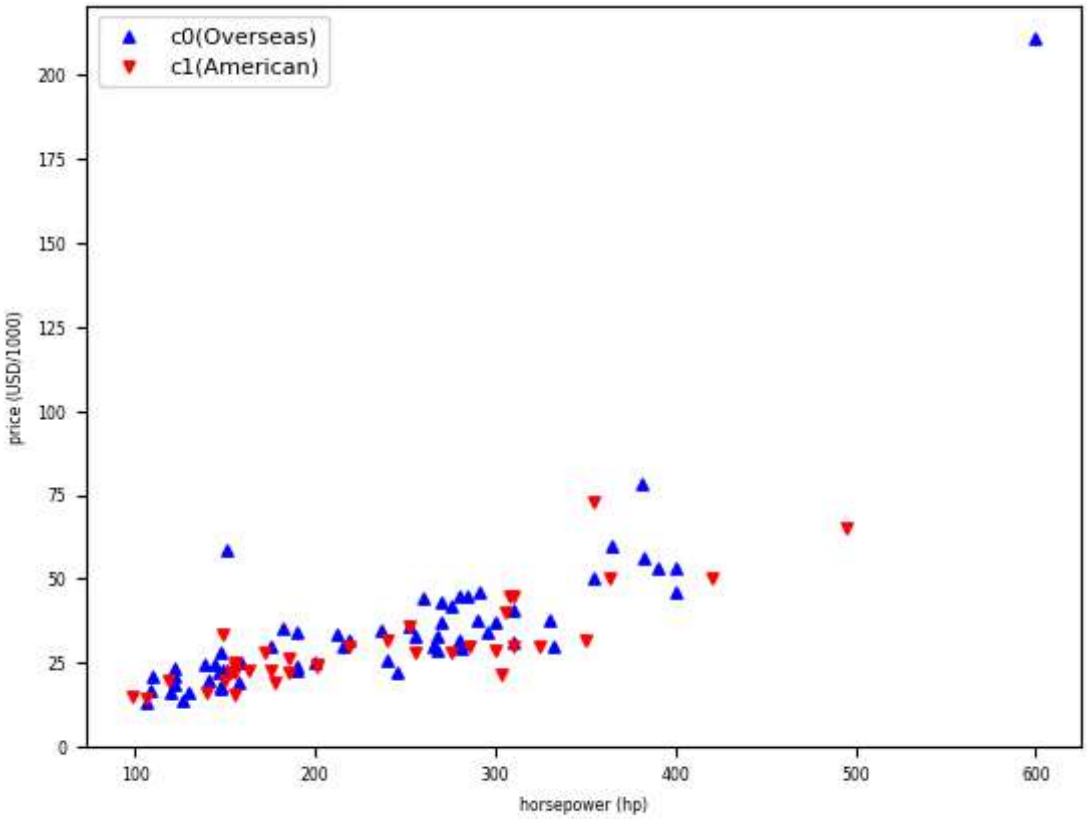
```
plt.ylabel("frequency")
plt.legend()
plt.show()
```



Price Distribution: Analyzing the Frequency of Price Ranges

In [ ]:
```
import math

import matplotlib.pyplot as plt
fig, ax = plt.subplots()
plt.rcParams.update({'font.size': 8}) #Change the font size of the table to 6 for b

import numpy as np
category = np.array(df.iloc[:, 4]) # Get an array of all values in Category column

## Separate the instances into their different categories

pos = 0 # position of the price in the dataframe
price_overseas = np.array([]) # Store the values of prices that fall under the Over
price_american = np.array([]) # Store the values of prices that fall under the Amer
hp_overseas = np.array([]) # Store the values of horsepowers that fall under the Ov
hp_american = np.array([]) # Store the values of horsepowers that fall under the Ov

for row in category:
    # Filter categories
    if row == "Overseas":
        price_overseas = np.append(price_overseas, df.iloc[pos, 1])
        hp_overseas = np.append(hp_overseas, df.iloc[pos, 2])
    else:
        price_american = np.append(price_american, df.iloc[pos, 1])
        hp_american = np.append(hp_american, df.iloc[pos, 2])
```

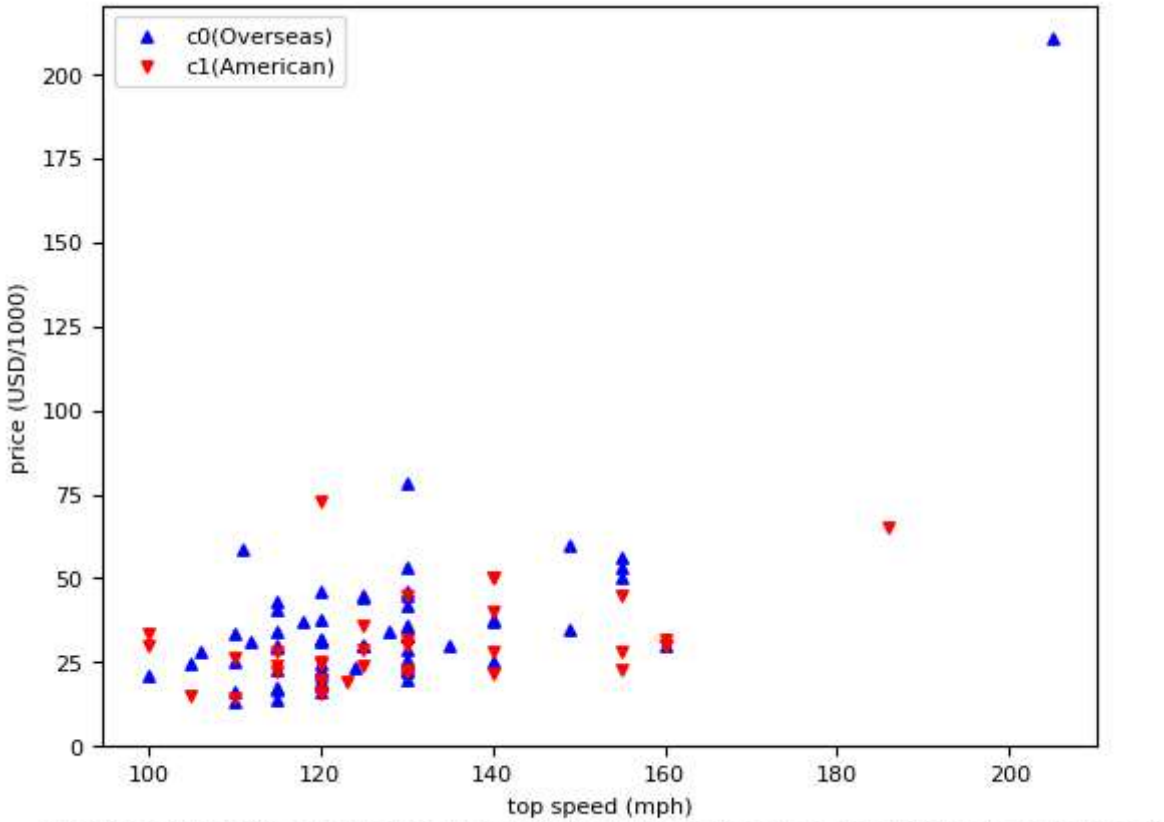
```
    pos += 1


## Adjust y labels
# Statistics of Price column
price = np.array(df.iloc[:, 1]) # Get an array of all values in Price column
class_ranges = [] # store class ranges in bins format
max_price = np.max(price) # Minimum price
min_price = 0

# Make intervals
yticks_val = [] # Store values of xticks
yticks_label = [] # Store labels name xtick

cur_price = 0

while cur_price <= max_price:
    yticks_label.append(str(math.floor(cur_price/1000)))
    yticks_val.append(cur_price)
    cur_price += 25000


## Draw scatter plot
ax.scatter(hp_overseas, price_overseas, c='b', marker='^', s=15, label="c0(Overseas
ax.scatter(hp_american, american, c='r', marker='v', s=15, label="c1(American)")

plt.yticks(yticks_val, yticks_label)
plt.title("Horsepower vs. Price: A Comparison of Car Performance and Cost", y = -0.
plt.xlabel("horsepower (hp)")
plt.ylabel("price (USD/1000)")
plt.legend()
plt.show()
```



Horsepower vs. Price: A Comparison of Car Performance and Cost

In [ ]:
```
import math

import matplotlib.pyplot as plt
fig, ax = plt.subplots()
plt.rcParams.update({'font.size': 8}) #Change the font size of the table to 6 for b

import numpy as np
category = np.array(df.iloc[:, 4]) # Get an array of all values in Category column

## Separate the instances into their different categories

pos = 0 # position of the price in the dataframe
price_overseas = np.array([]) # Store the values of prices that fall under the Over
price_american = np.array([]) # Store the values of prices that fall under the Amer
ts_overseas = np.array([]) # Store the values of top speeds that fall under the Ove
ts_american = np.array([]) # Store the values of top speeds that fall under the Ove

for row in category:
    # Filter categories
    if row == "Overseas":
        price_overseas = np.append(price_overseas, df.iloc[pos, 1])
        ts_overseas = np.append(ts_overseas, df.iloc[pos, 3])
    else:
        price_american = np.append(price_american, df.iloc[pos, 1])
        ts_american = np.append(ts_american, df.iloc[pos, 3])
    pos += 1


## Adjust y labels
```

```
# Statistics of Price column
price = np.array(df.iloc[:, 1]) # Get an array of all values in Price column
class_ranges = [] # store class ranges in bins format
max_price = np.max(price) # Minimum price
min_price = 0

# Make intervals
yticks_val = [] # Store values of yticks
yticks_label = [] # Store labels name ytick

cur_price = 0

while cur_price <= max_price:
    yticks_label.append(str(math.floor(cur_price/1000)))
    yticks_val.append(cur_price)
    cur_price += 25000


## Draw scatter plot
ax.scatter(ts_overseas, price_overseas, c='b', marker='^', s=15, label="c0(Overseas
ax.scatter(ts_american, american, c='r', marker='v', s=15, label="c1(American)")

plt.yticks(yticks_val, yticks_label)
plt.title("Top Speed vs. Price: Exploring the Relationship between Car Performance
plt.xlabel("top speed (mph)")
plt.ylabel("price (USD/1000)")
plt.legend()
plt.show()
```



Top Speed vs. Price: Exploring the Relationship between Car Performance and Cost