

Python Code for Data Processing and Visualization

Assignment Report: Project-1

Student ID: 253001086

Student Name: Anas Mohamed Barro

```
import pandas as pd
```

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
import seaborn as sns
```

```
from sklearn.impute import SimpleImputer
```

```
import os
```

Define input and output paths

```
base_path = r"C:\Users\USER\OneDrive - Master Works for Information  
Technology\Desktop\midocean"
```

```
input_file = os.path.join(base_path, "assessment.csv")
```

```
output_file = os.path.join(base_path, "cleaned_assessment.csv")
```

Load dataset

```
df = pd.read_csv(input_file)
```

Handle missing values: impute numeric columns with mean

```
numeric_cols = df.select_dtypes(include=[np.number]).columns
```

```
imputer = SimpleImputer(strategy='mean')
```

```
df[numeric_cols] = imputer.fit_transform(df[numeric_cols])
```

Remove outliers using IQR method

```
def remove_outliers_iqr(data, column):  
  
    Q1 = data[column].quantile(0.25)  
  
    Q3 = data[column].quantile(0.75)  
  
    IQR = Q3 - Q1  
  
    lower = Q1 - 1.5 * IQR  
  
    upper = Q3 + 1.5 * IQR  
  
    return data[(data[column] >= lower) & (data[column] <= upper)]  
  
for col in numeric_cols:  
  
    df = remove_outliers_iqr(df, col)
```

Create visualizations folder

```
viz_path = os.path.join(base_path, "visualizations")  
  
os.makedirs(viz_path, exist_ok=True)
```

Plot 1: Price distribution

```
plt.figure(figsize=(8, 5))  
  
sns.histplot(df['price'], kde=True, bins=30)  
  
plt.title("Price Distribution")  
  
plt.savefig(os.path.join(viz_path, "price_distribution.png"))  
  
plt.close()
```

Plot 2: Engine size vs Price scatter plot

```
plt.figure(figsize=(8, 5))

sns.scatterplot(x='engine-size', y='price', data=df)

plt.title("Engine Size vs Price")

plt.savefig(os.path.join(viz_path, "engine_size_vs_price.png"))

plt.close()
```

Plot 3: Car count by manufacturer

```
plt.figure(figsize=(10, 6))

sns.countplot(y='make', data=df, order=df['make'].value_counts().index)

plt.title("Car Count by Make")

plt.savefig(os.path.join(viz_path, "car_count_by_make.png"))

plt.close()
```

Save cleaned dataset

```
df.to_csv(output_file, index=False)

print("Data cleaning and visualization completed. Clean file saved to:", output_file)
```

Initial information about the dataset

```
print("\n--- Initial Data Information ---")

print(df.info())
```