APPLIED DATA SCIENCE

ASSIGNMENT NO : 1

TITLE : PYTHON FOR DATA HANDLING

NAME: Uday Annasaheb Thete

ROLL NO : 145

CLASS : TY-B

GITHUB LINK :

DATASET LINK: https://www.kaggle.com/datasets/nehalbirla/vehicle-dataset-from-cardekho

CODE:

```
# ============================================================
# APPLIED DATA SCIENCE
# Assignment 1 – Python for Data Handling
# Dataset: Vehicle Car Data (Kaggle)
# ============================================================
# 1. Import Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
# ------------------------------------------------------------
# 2. Load Dataset
# ------------------------------------------------------------
df = pd.read_csv("car data.csv")
print("Dataset Loaded Successfully")
print("="*100)
# ------------------------------------------------------------
# 3. Clean Column Names
# ------------------------------------------------------------
df.columns = df.columns.str.strip().str.lower()
print("Columns after cleaning:")
print(df.columns.tolist())
```

```
print("="*100)
# ------------------------------------------------------------
# 4. Dataset Exploration
# ------------------------------------------------------------
print("First 5 Records:")
print(df.head())
print("\nDataset Shape:", df.shape)
print("\nDataset Information:")
df.info()
print("="*100)
# ------------------------------------------------------------
# 5. Check Missing Values and Zero Values
# ------------------------------------------------------------
print("Missing Values:")
print(df.isnull().sum())
numerical_cols = df.select_dtypes(include=np.number).columns
print("\nZero Values in Numerical Columns:")
print((df[numerical_cols] == 0).sum())
print("="*100)
# ------------------------------------------------------------
# 6. Remove Duplicate Records
# ------------------------------------------------------------
print("Duplicate Records:", df.duplicated().sum())
df.drop_duplicates(inplace=True)
print("Duplicates Removed")
print("="*100)
# ------------------------------------------------------------
# 7. Handle Missing Values
# ------------------------------------------------------------
df[numerical_cols] = df[numerical_cols].fillna(df[numerical_cols].mean())
cat_cols = df.select_dtypes(include=['object']).columns
for col in cat_cols:
```

```
    df[col] = df[col].fillna(df[col].mode()[0])
print("Missing Values Handled")
print("="*100)
# ------------------------------------------------------------
# 8. Detect Price Columns Automatically
# ------------------------------------------------------------
price_cols = [col for col in df.columns if "price" in col]
print("Detected Price Columns:", price_cols)
df[price_cols] = df[price_cols].apply(pd.to_numeric, errors='coerce')
df[price_cols] = df[price_cols].fillna(df[price_cols].mean())
print("="*100)
# ------------------------------------------------------------
# 9. Feature Engineering – Car Age
# ------------------------------------------------------------
df['car_age'] = 2024 - df['year']
print("Car Age Column Created")
print("="*100)
# ------------------------------------------------------------
# 10. Statistical Measures
# ------------------------------------------------------------
print("Selling Price Statistics")
print("Mean:", df['selling_price'].mean())
print("Median:", df['selling_price'].median())
print("Mode:", df['selling_price'].mode()[0])
print("Skewness:", df['selling_price'].skew())
print("="*100)
# ------------------------------------------------------------
# 11. Basic Visualization (For Output Screenshots)
# ------------------------------------------------------------
plt.figure(figsize=(6,4))
sns.histplot(df['selling_price'], kde=True)
plt.title("Distribution of Selling Price")
```

```
plt.show()

plt.figure(figsize=(6,4))

sns.countplot(x='fuel_type', data=df)

plt.title("Fuel Type Distribution")

plt.show()

print("Preprocessing Completed Successfully")
```

OUTPUT:

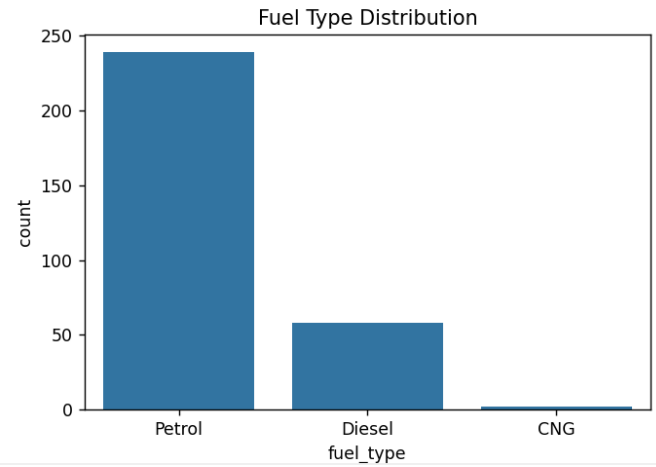
```
Dataset Loaded Successfully
============================================================================================================
Columns after cleaning:
['car_name', 'year', 'selling_price', 'present_price', 'kms_driven', 'fuel_type', 'seller_type', 'transmission', 'owner'
]
============================================================================================================
First 5 Records:
  car_name  year  selling_price  present_price  kms_driven fuel_type seller_type transmission  owner
0     ritz  2014           3.35           5.59       27000    Petrol      Dealer       Manual      0
1      sx4  2013           4.75           9.54       43000    Diesel      Dealer       Manual      0
2     ciaz  2017           7.25           9.85        6900    Petrol      Dealer       Manual      0
3  wagon r  2011           2.85           4.15        5200    Petrol      Dealer       Manual      0
4    swift  2014           4.60           6.87       42450    Diesel      Dealer       Manual      0

Dataset Shape: (301, 9)
```

```
<class 'pandas.DataFrame'>
RangeIndex: 301 entries, 0 to 300
Data columns (total 9 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   car_name       301 non-null    str
 1   year           301 non-null    int64
 2   selling_price  301 non-null    float64
 3   present_price  301 non-null    float64
 4   kms_driven     301 non-null    int64
 5   fuel_type      301 non-null    str
 6   seller_type    301 non-null    str
 7   transmission   301 non-null    str
 8   owner          301 non-null    int64
dtypes: float64(2), int64(3), str(4)
memory usage: 21.3 KB
============================================================================================================
Missing Values:
car_name         0
year             0
selling_price    0
present_price    0
kms_driven       0
fuel_type        0
```

```
Missing Values Handled
============================================================================================================
Detected Price Columns: ['selling_price', 'present_price']
============================================================================================================
Car Age Column Created
============================================================================================================
Selling Price Statistics
Mean: 4.589632107023411
Median: 3.51
Mode: 0.45
Skewness: 2.536521826497541
============================================================================================================
```

```
Zero Values in Numerical Columns:
year              0
selling_price     0
present_price     0
kms_driven        0
owner           290
dtype: int64
============================================================================================================
Duplicate Records: 2
Duplicates Removed
============================================================================================================
```



INTERPRETATION:

INTERPRETATION SECTION

(Write this below output — same style as your PDF)

Interpretation

Data Cleaning

- Column names converted into lowercase format.
- Duplicate vehicle records removed.
- Missing values replaced using mean (numerical) and mode (categorical).

Data Exploration

- Dataset contains vehicle details like year, fuel type, transmission and selling price.
- df.info() helped identify datatype structure.
- Dataset shape indicates total number of car records.

Zero Value Analysis

- Checked zero values in kms_driven and price columns.
- Helps detect unrealistic entries.

Price Column Processing

- Selling_price and present_price detected automatically.
- Converted into numeric format for analysis.

Feature Engineering

- Created new column car_age from year.
- Helps understand vehicle depreciation.