Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

Third Year B. Tech (EL & CE)

Semester: VI **Subject:** Data Science for Engineering

Name: Shreerang Mhatre **Class: TY**

Roll No: 52 **Batch: A2**

Experiment No: 03

Name of the Experiment: **Data Pre-processing.**

Performed on: 18/03/2024

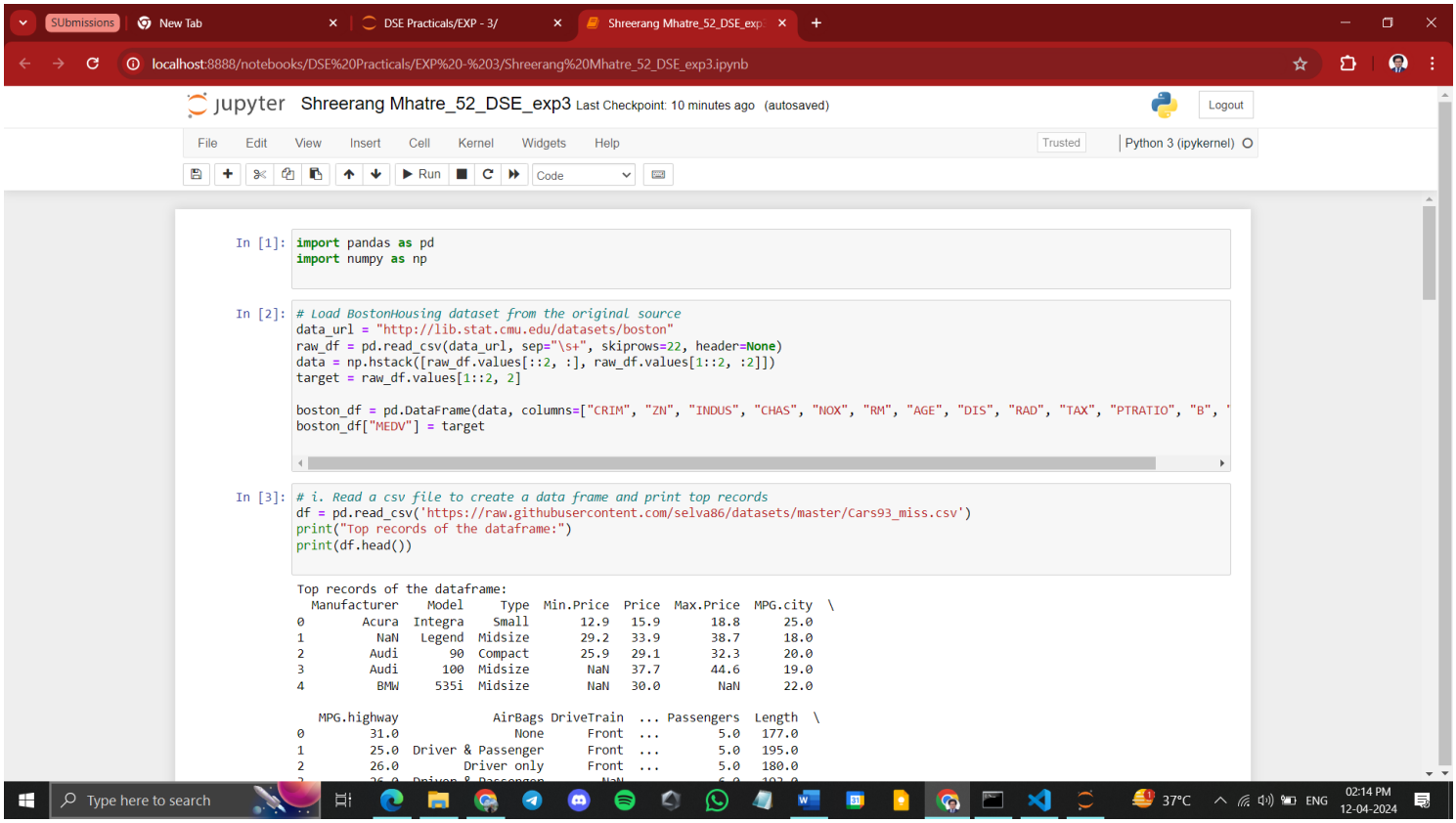
Submitted on: 18/04/2024

Problem Statement:

Load
Data and perform Data Pre-processing.

Input: df = pd.read_csv
('https://raw.githubusercontent.com/selva86/datasets/master/Cars93_miss.csv')

i. Read a csv file to create a data frame and print top records.
ii. Check if there are any missing values in the data.
iii. Drop null values / Impute the missing values with mean / median.
iv. Import ‘crim’ and ‘medv’ columns of the Boston Housing dataset as a dataframe and ge the nrows, ncolumns, datatype, summary stats of each column of a dataframe.
v. Which manufacturer, model and type has thehighest Price?
vi. How to create one-hot encodings of a categorical variable.



SUbmissions | New Tab | DSE Practicals/EXP - 3/ | Shreerang Mhatre_52_DSE_exp

localhost:8888/notebooks/DSE%20Practicals/EXP%20-%203/Shreerang%20Mhatre_52_DSE_exp3.ipynb

jupyter Shreerang Mhatre_52_DSE_exp3 Last Checkpoint: 10 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

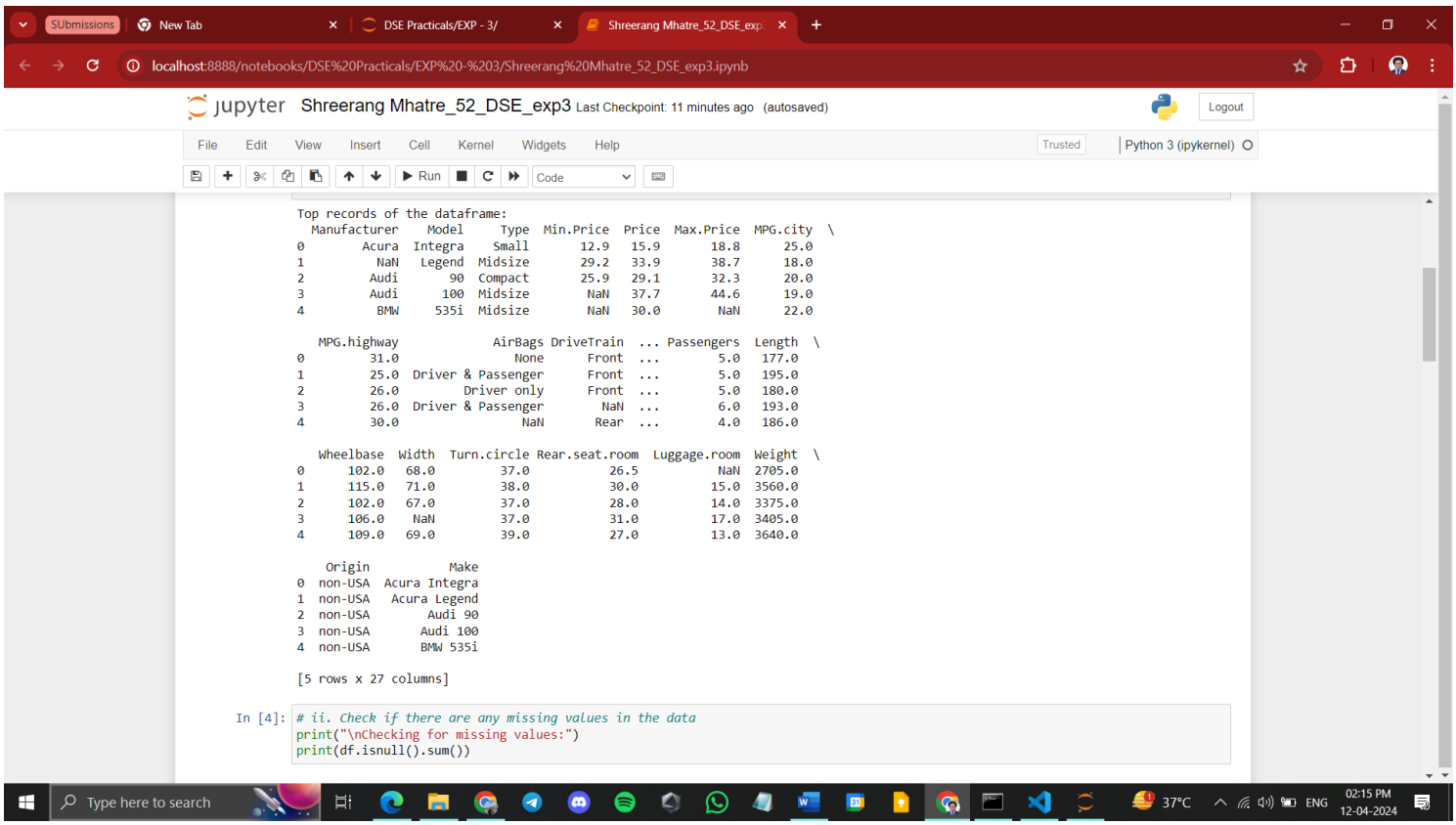
Run Code

In [1]:
```
import pandas as pd
import numpy as np
```

In [2]:
```
# Load BostonHousing dataset from the original source
data_url = "http://lib.stat.cmu.edu/datasets/boston"
raw_df = pd.read_csv(data_url, sep="\s+", skiprows=22, header=None)
data = np.hstack([raw_df.values[::2, :], raw_df.values[1::2, :2]])
target = raw_df.values[1::2, 2]

boston_df = pd.DataFrame(data, columns=["CRIM", "ZN", "INDUS", "CHAS", "NOX", "RM", "AGE", "DIS", "RAD", "TAX", "PTRATIO", "B", '
boston_df["MEDV"] = target
```

In [3]:
```
# i. Read a csv file to create a data frame and print top records
df = pd.read_csv('https://raw.githubusercontent.com/selva86/datasets/master/Cars93_miss.csv')
print("Top records of the dataframe:")
print(df.head())
```

```
Top records of the dataframe:
  Manufacturer    Model     Type  Min.Price  Price  Max.Price  MPG.city  \
0        Acura  Integra    Small       12.9   15.9       18.8      25.0
1          NaN  Legend  Midsize       29.2   33.9       38.7      18.0
2         Audi      90  Compact       25.9   29.1       32.3      20.0
3         Audi     100  Midsize        NaN   37.7       44.6      19.0
4          BMW    535i  Midsize        NaN   30.0        NaN      22.0

   MPG.highway            AirBags DriveTrain  ... Passengers  Length  \
0         31.0               None      Front  ...        5.0   177.0
1         25.0  Driver & Passenger      Front  ...        5.0   195.0
2         26.0        Driver only      Front  ...        5.0   180.0
```



SUbmissions | New Tab | DSE Practicals/EXP - 3/ | Shreerang Mhatre_52_DSE_exp

localhost:8888/notebooks/DSE%20Practicals/EXP%20-%203/Shreerang%20Mhatre_52_DSE_exp3.ipynb

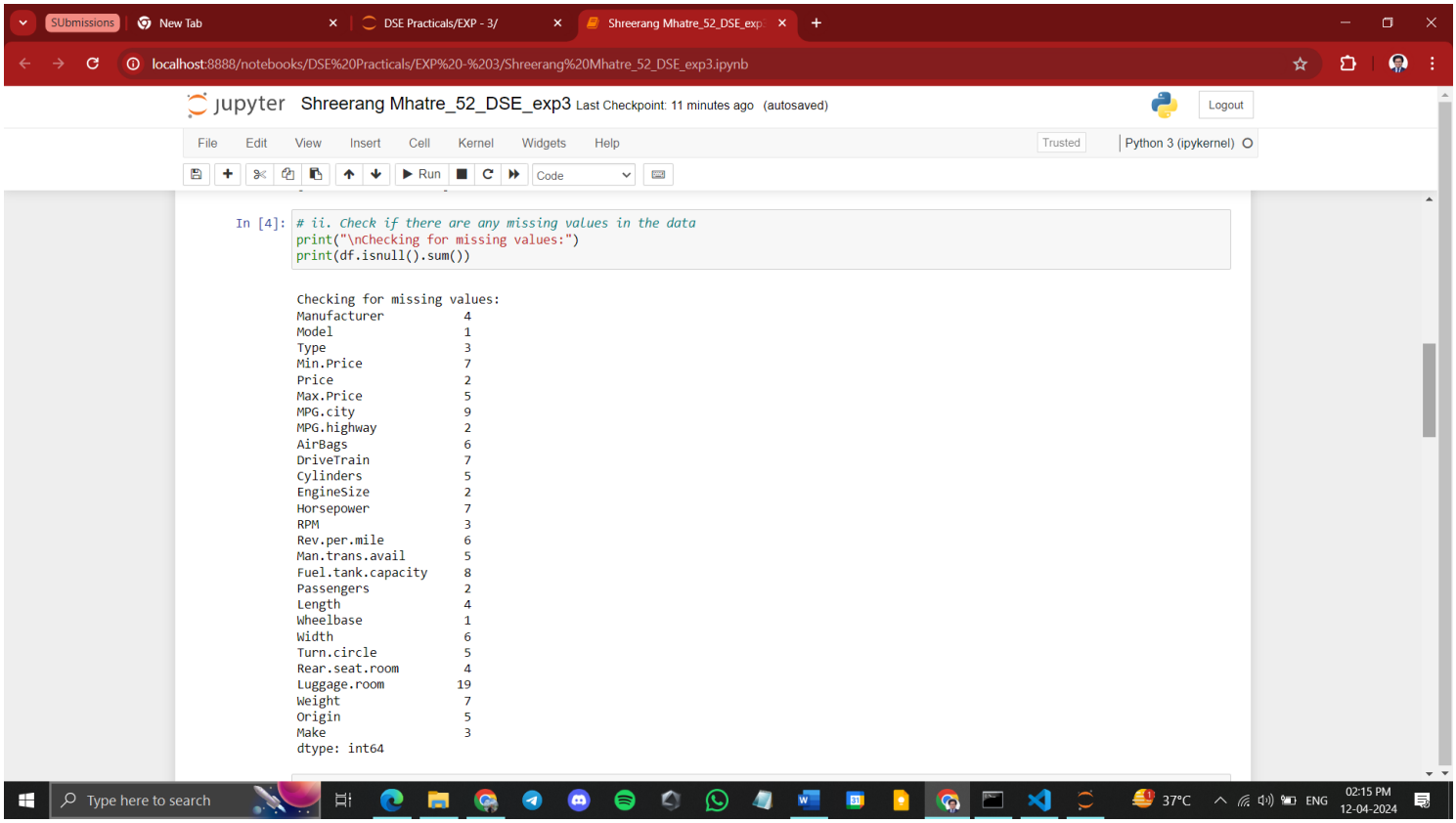
jupyter Shreerang Mhatre_52_DSE_exp3 Last Checkpoint: 11 minutes ago (autosaved) Logout

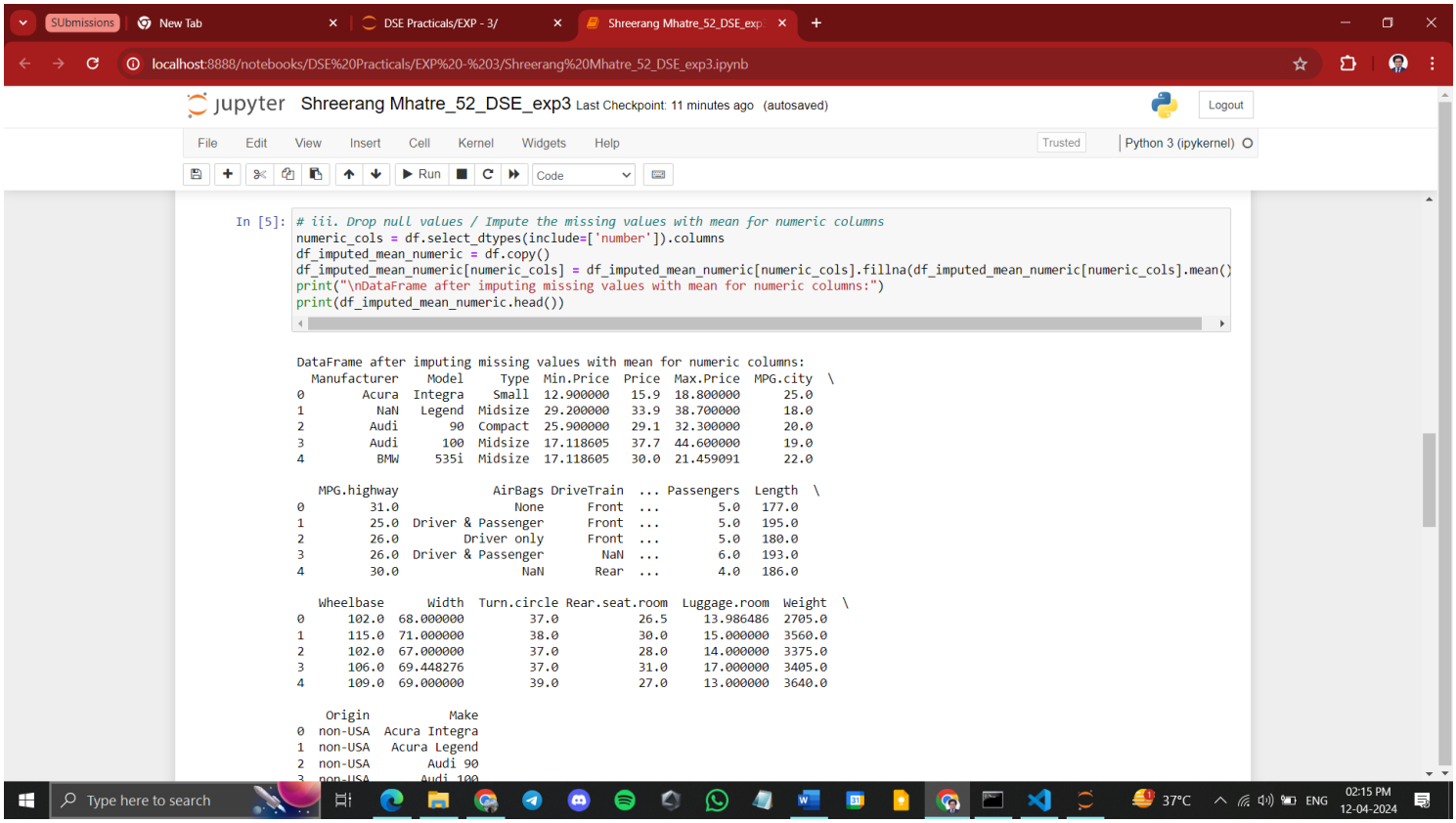
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

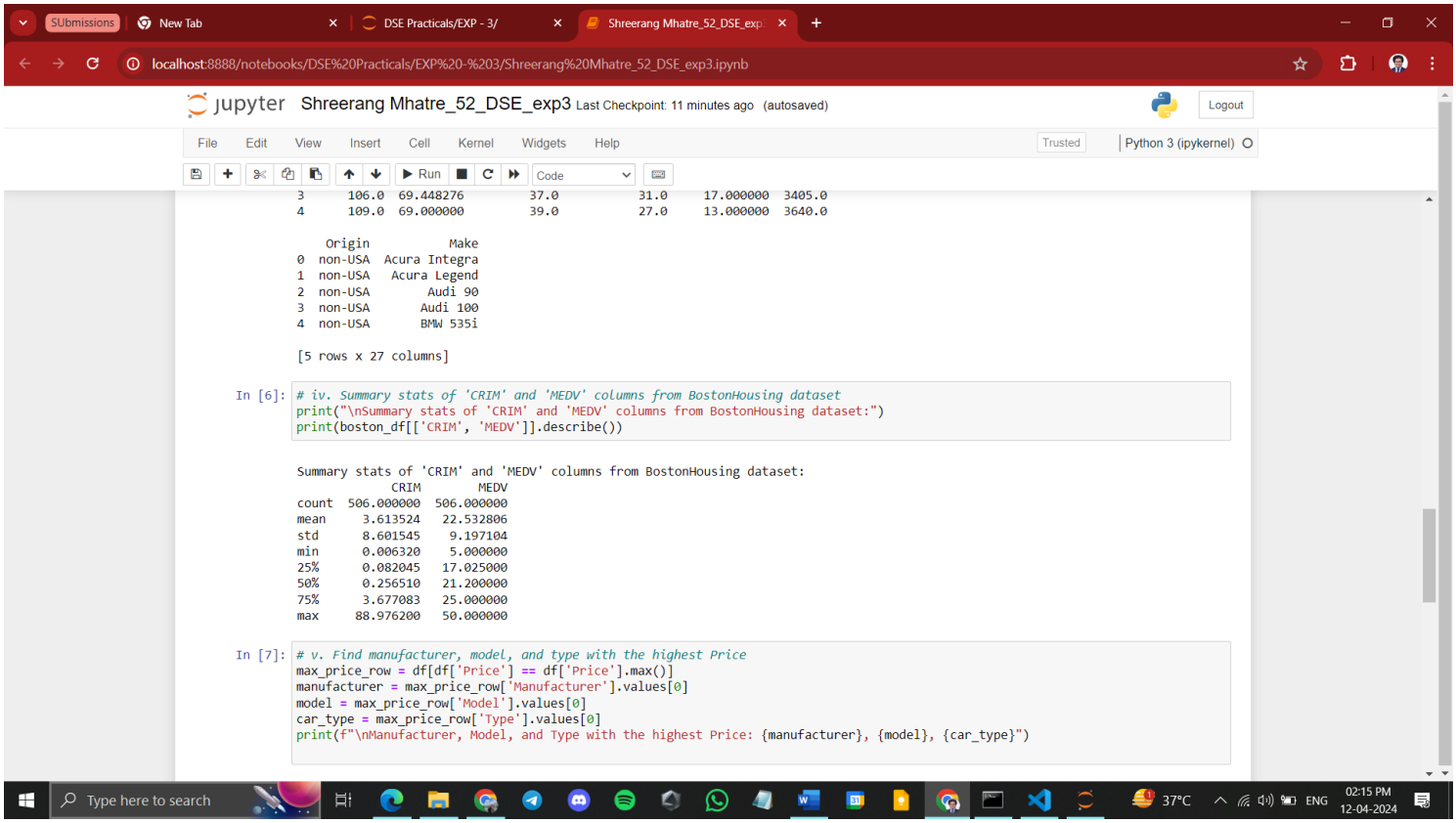
Run Code

```
Top records of the dataframe:
  Manufacturer    Model     Type  Min.Price  Price  Max.Price  MPG.city  \
0        Acura  Integra    Small       12.9   15.9       18.8      25.0
1          NaN  Legend  Midsize       29.2   33.9       38.7      18.0
2         Audi      90  Compact       25.9   29.1       32.3      20.0
3         Audi     100  Midsize        NaN   37.7       44.6      19.0
4          BMW    535i  Midsize        NaN   30.0        NaN      22.0

   MPG.highway            AirBags DriveTrain  ... Passengers  Length  \
0         31.0               None      Front  ...        5.0   177.0
1         25.0  Driver & Passenger      Front  ...        5.0   195.0
2         26.0        Driver only      Front  ...        5.0   180.0
3         26.0  Driver & Passenger        NaN  ...        6.0   193.0
4         30.0                NaN       Rear  ...        4.0   186.0

   Wheelbase  Width  Turn.circle Rear.seat.room  Luggage.room  Weight  \
0      102.0   68.0         37.0           26.5           NaN  2705.0
1      115.0   71.0         38.0           30.0          15.0  3560.0
2      102.0   67.0         37.0           28.0          14.0  3375.0
3      106.0    NaN         37.0           31.0          17.0  3405.0
4      109.0   69.0         39.0           27.0          13.0  3640.0

    Origin          Make
0  non-USA  Acura Integra
1  non-USA   Acura Legend
2  non-USA       Audi 90
3  non-USA      Audi 100
4  non-USA      BMW 535i

[5 rows x 27 columns]
```

In [4]:
```
# ii. Check if there are any missing values in the data
print("\nChecking for missing values:")
print(df.isnull().sum())
```


SUbmissions
New Tab
DSE Practicals/EXP - 3/
Shreerang Mhatre_52_DSE_exp
localhost:8888/notebooks/DSE%20Practicals/EXP%20-%203/Shreerang%20Mhatre_52_DSE_exp3.ipynb
jupyter Shreerang Mhatre_52_DSE_exp3 Last Checkpoint: 11 minutes ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code
In [4]:
# ii. Check if there are any missing values in the data
print("\nChecking for missing values:")
print(df.isnull().sum())
Checking for missing values:
Manufacturer 4
Model 1
Type 3
Min.Price 7
Price 2
Max.Price 5
MPG.city 9
MPG.highway 2
AirBags 6
DriveTrain 7
Cylinders 5
EngineSize 2
Horsepower 7
RPM 3
Rev.per.mile 6
Man.trans.avail 5
Fuel.tank.capacity 8
Passengers 2
Length 4
Wheelbase 1
Width 6
Turn.circle 5
Rear.seat.room 4
Luggage.room 19
Weight 7
Origin 5
Make 3
dtype: int64
Type here to search
37°C
ENG
02:15 PM
12-04-2024


SUbmissions
New Tab
DSE Practicals/EXP - 3/
Shreerang Mhatre_52_DSE_exp
localhost:8888/notebooks/DSE%20Practicals/EXP%20-%203/Shreerang%20Mhatre_52_DSE_exp3.ipynb
jupyter Shreerang Mhatre_52_DSE_exp3 Last Checkpoint: 11 minutes ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code
In [5]:
# iii. Drop null values / Impute the missing values with mean for numeric columns
numeric_cols = df.select_dtypes(include=['number']).columns
df_imputed_mean_numeric = df.copy()
df_imputed_mean_numeric[numeric_cols] = df_imputed_mean_numeric[numeric_cols].fillna(df_imputed_mean_numeric[numeric_cols].mean()
print("\nDataFrame after imputing missing values with mean for numeric columns:")
print(df_imputed_mean_numeric.head())
DataFrame after imputing missing values with mean for numeric columns:
Manufacturer Model Type Min.Price Price Max.Price MPG.city \
0 Acura Integra Small 12.900000 15.9 18.800000 25.0
1 NaN Legend Midsize 29.200000 33.9 38.700000 18.0
2 Audi 90 Compact 25.900000 29.1 32.300000 20.0
3 Audi 100 Midsize 17.118605 37.7 44.600000 19.0
4 BMW 535i Midsize 17.118605 30.0 21.459091 22.0
MPG.highway AirBags DriveTrain ... Passengers Length \
0 31.0 None Front ... 5.0 177.0
1 25.0 Driver & Passenger Front ... 5.0 195.0
2 26.0 Driver only Front ... 5.0 180.0
3 26.0 Driver & Passenger NaN ... 6.0 193.0
4 30.0 NaN Rear ... 4.0 186.0
Wheelbase Width Turn.circle Rear.seat.room Luggage.room Weight \
0 102.0 68.000000 37.0 26.5 13.986486 2705.0
1 115.0 71.000000 38.0 30.0 15.000000 3560.0
2 102.0 67.000000 37.0 28.0 14.000000 3375.0
3 106.0 69.448276 37.0 31.0 17.000000 3405.0
4 109.0 69.000000 39.0 27.0 13.000000 3640.0
Origin Make
0 non-USA Acura Integra
1 non-USA Acura Legend
2 non-USA Audi 90
Type here to search
37°C
ENG
02:15 PM
12-04-2024



SUbmissions | New Tab | DSE Practicals/EXP - 3/ | Shreerang Mhatre_52_DSE_exp

localhost:8888/notebooks/DSE%20Practicals/EXP%20-%203/Shreerang%20Mhatre_52_DSE_exp3.ipynb

jupyter Shreerang Mhatre_52_DSE_exp3 Last Checkpoint: 11 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3 (ipykernel)

Run | Code

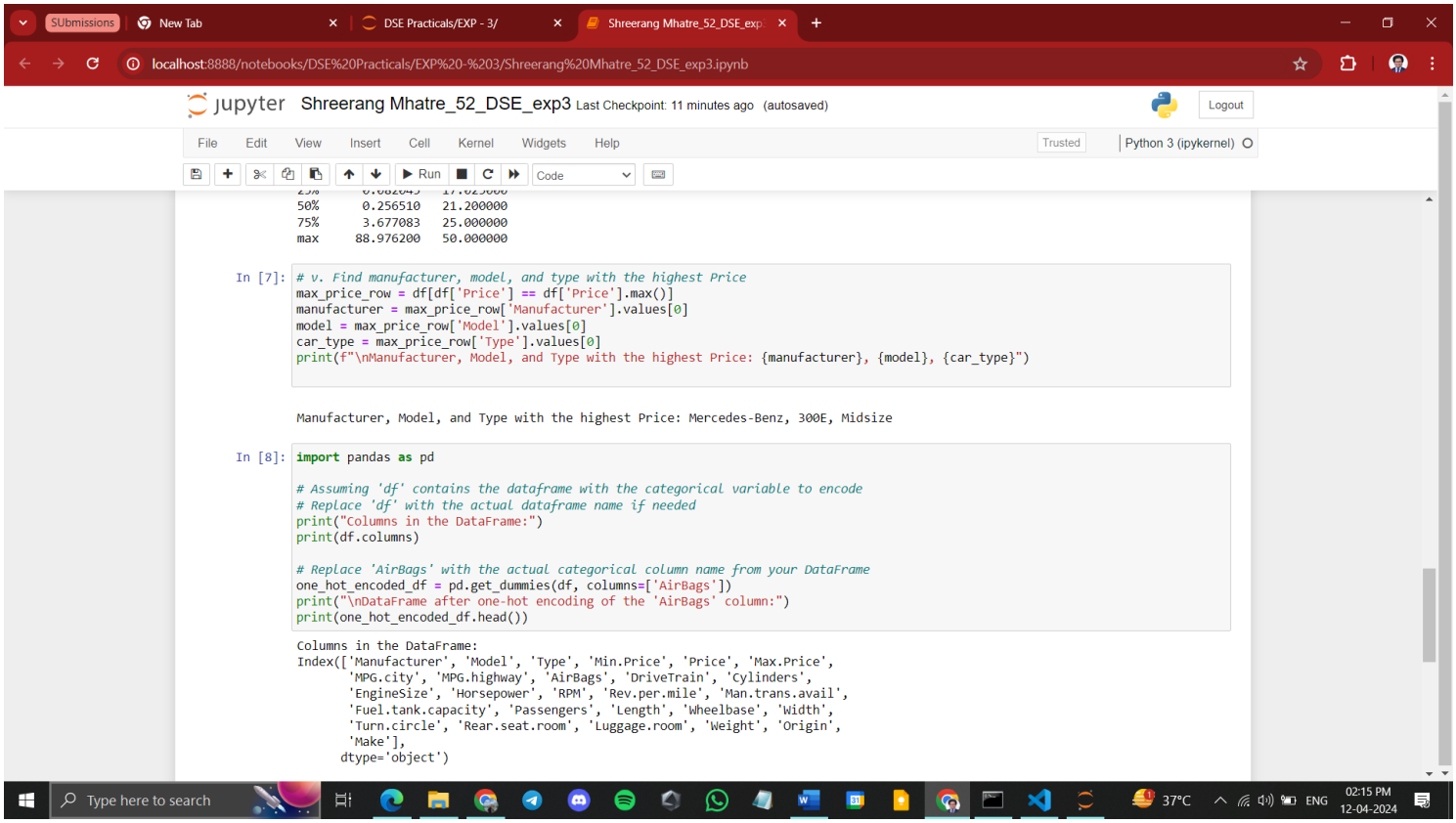
```
3      106.0  69.448276         37.0         31.0     17.000000  3405.0
4      109.0  69.000000         39.0         27.0     13.000000  3640.0

    Origin          Make
0  non-USA  Acura Integra
1  non-USA   Acura Legend
2  non-USA        Audi 90
3  non-USA       Audi 100
4  non-USA       BMW 535i

[5 rows x 27 columns]
```

In [6]:
```
# iv. Summary stats of 'CRIM' and 'MEDV' columns from BostonHousing dataset
print("\nSummary stats of 'CRIM' and 'MEDV' columns from BostonHousing dataset:")
print(boston_df[['CRIM', 'MEDV']].describe())
```

```
Summary stats of 'CRIM' and 'MEDV' columns from BostonHousing dataset:
             CRIM        MEDV
count  506.000000  506.000000
mean     3.613524   22.532806
std      8.601545    9.197104
min      0.006320    5.000000
25%      0.082045   17.025000
50%      0.256510   21.200000
75%      3.677083   25.000000
max     88.976200   50.000000
```

In [7]:
```
# v. Find manufacturer, model, and type with the highest Price
max_price_row = df[df['Price'] == df['Price'].max()]
manufacturer = max_price_row['Manufacturer'].values[0]
model = max_price_row['Model'].values[0]
car_type = max_price_row['Type'].values[0]
print(f"\nManufacturer, Model, and Type with the highest Price: {manufacturer}, {model}, {car_type}")
```

Type here to search | 37°C | ENG | 02:15 PM 12-04-2024



SUbmissions | New Tab | DSE Practicals/EXP - 3/ | Shreerang Mhatre_52_DSE_exp

localhost:8888/notebooks/DSE%20Practicals/EXP%20-%203/Shreerang%20Mhatre_52_DSE_exp3.ipynb

jupyter Shreerang Mhatre_52_DSE_exp3 Last Checkpoint: 11 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3 (ipykernel)

Run | Code

```
50%      0.256510   21.200000
75%      3.677083   25.000000
max     88.976200   50.000000
```

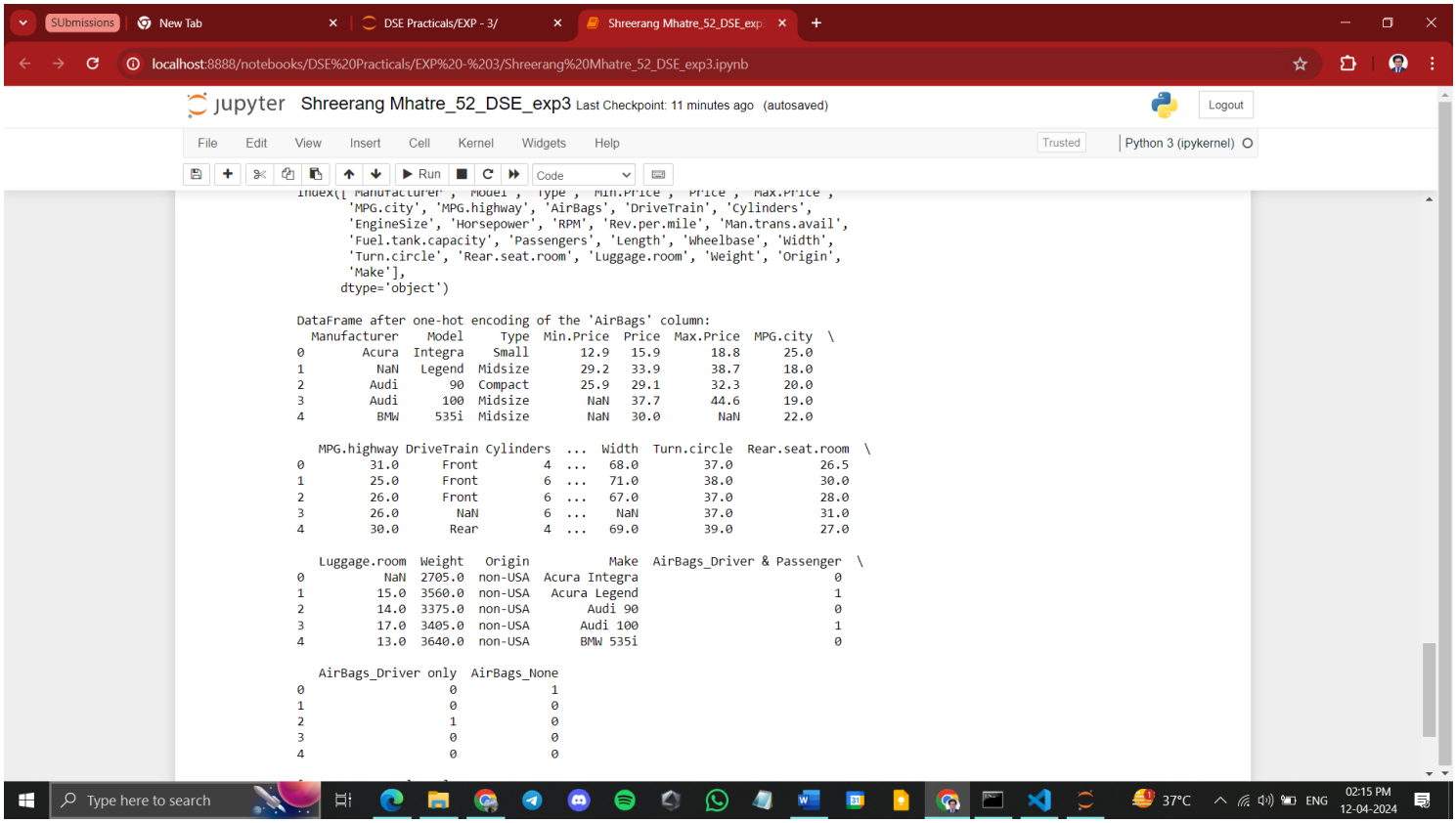
In [7]:
```
# v. Find manufacturer, model, and type with the highest Price
max_price_row = df[df['Price'] == df['Price'].max()]
manufacturer = max_price_row['Manufacturer'].values[0]
model = max_price_row['Model'].values[0]
car_type = max_price_row['Type'].values[0]
print(f"\nManufacturer, Model, and Type with the highest Price: {manufacturer}, {model}, {car_type}")
```

```
Manufacturer, Model, and Type with the highest Price: Mercedes-Benz, 300E, Midsize
```

In [8]:
```
import pandas as pd

# Assuming 'df' contains the dataframe with the categorical variable to encode
# Replace 'df' with the actual dataframe name if needed
print("Columns in the DataFrame:")
print(df.columns)

# Replace 'AirBags' with the actual categorical column name from your DataFrame
one_hot_encoded_df = pd.get_dummies(df, columns=['AirBags'])
print("\nDataFrame after one-hot encoding of the 'AirBags' column:")
print(one_hot_encoded_df.head())
```

```
Columns in the DataFrame:
Index(['Manufacturer', 'Model', 'Type', 'Min.Price', 'Price', 'Max.Price',
       'MPG.city', 'MPG.highway', 'AirBags', 'DriveTrain', 'Cylinders',
       'EngineSize', 'Horsepower', 'RPM', 'Rev.per.mile', 'Man.trans.avail',
       'Fuel.tank.capacity', 'Passengers', 'Length', 'Wheelbase', 'Width',
       'Turn.circle', 'Rear.seat.room', 'Luggage.room', 'Weight', 'Origin',
       'Make'],
      dtype='object')
```

Type here to search | 37°C | ENG | 02:15 PM 12-04-2024


SUbmissions
New Tab
DSE Practicals/EXP - 3/
Shreerang Mhatre_52_DSE_exp
localhost:8888/notebooks/DSE%20Practicals/EXP%20-%203/Shreerang%20Mhatre_52_DSE_exp3.ipynb
jupyter Shreerang Mhatre_52_DSE_exp3 Last Checkpoint: 11 minutes ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code
Index(['Manufacturer', 'Model', 'Type', 'Min.Price', 'Price', 'Max.Price',
'MPG.city', 'MPG.highway', 'AirBags', 'DriveTrain', 'Cylinders',
'EngineSize', 'Horsepower', 'RPM', 'Rev.per.mile', 'Man.trans.avail',
'Fuel.tank.capacity', 'Passengers', 'Length', 'Wheelbase', 'Width',
'Turn.circle', 'Rear.seat.room', 'Luggage.room', 'Weight', 'Origin',
'Make'],
dtype='object')
DataFrame after one-hot encoding of the 'AirBags' column:
Manufacturer Model Type Min.Price Price Max.Price MPG.city \
0 Acura Integra Small 12.9 15.9 18.8 25.0
1 NaN Legend Midsize 29.2 33.9 38.7 18.0
2 Audi 90 Compact 25.9 29.1 32.3 20.0
3 Audi 100 Midsize NaN 37.7 44.6 19.0
4 BMW 535i Midsize NaN 30.0 NaN 22.0
MPG.highway DriveTrain Cylinders ... Width Turn.circle Rear.seat.room \
0 31.0 Front 4 ... 68.0 37.0 26.5
1 25.0 Front 6 ... 71.0 38.0 30.0
2 26.0 Front 6 ... 67.0 37.0 28.0
3 26.0 NaN 6 ... NaN 37.0 31.0
4 30.0 Rear 4 ... 69.0 39.0 27.0
Luggage.room Weight Origin Make AirBags_Driver & Passenger \
0 NaN 2705.0 non-USA Acura Integra 0
1 15.0 3560.0 non-USA Acura Legend 1
2 14.0 3375.0 non-USA Audi 90 0
3 17.0 3405.0 non-USA Audi 100 1
4 13.0 3640.0 non-USA BMW 535i 0
AirBags_Driver only AirBags_None
0 0 1
1 0 0
2 1 0
3 0 0
4 0 0
Type here to search
37°C
ENG
02:15 PM
12-04-2024

Exp3

Shreerang Mhatre 52 A2

* Past Lab Questions

Q1) Explain the concept of outliers in data and their potential impact on machine learning models.

-> Outliers are data points that significantly differ from other observations in a dataset. They can occur due to various reasons such as measurement errors, data entry mistakes, or rare events.

① Model Performance -
Outliers can distort statistical analyses and modeling results, leading to biased estimate and inaccurate predictions

② Data Distribution -
outliers can skew the distribution of data, particularly in algorithms that assume a certain distribution.

③ Robustness of Models -
Some machine learning algorithms are sensitive to outliers, especially those based on distance metrices.

Q2) Why is data normalization & standardization often performed during data preprocessing?

→ Data Normalization -

It scales the data to a specific range, typically between 0 & 1 or -1 & 1. It ensures that features with different scales and units are comparable and contribute equally to the analysis.

Standardization -

Standardization transforms the data to have a mean of 0 and a standard deviation of 1. It centers the data around zero, making it easier to interpret and reducing the impact of outliers.

Q3) Why is it essential to document the data preprocessing steps applied to dataset? How does it benefit the machine process?

→ Documenting data preprocessing steps -

→

MIT-WPU
MIT WORLD PEACE UNIVERSITY | PUNE

① Reproducibility -
Documenting preprocessing steps allows others to reproduce the data cleaning, transformation, & feature engineering process.

② Debugging & Troubleshooting -
In complex data science projects, documenting preprocessing steps facilitates debugging and troubleshooting issues that arise during data analysis or model.

③ Model Interpretation -
Documenting preprocessing steps aids in model interpretion by explaining how the input data was prepared and transformed before training the model.

④ Compliance & Governance -
Documentation of preprocessing steps is crucial for compliance with regulatory requirements & data governance.

www.mitwpu.edu.in