Lab 09 - Finish the data cleanup.

Given the following code:

https://github.com/Univ-Wyo-Education/F21-1010/blob/main/class/lect/Lect-21/lab-09_start.py

```
import numpy as np
import pandas as pd
import re
import matplotlib.pyplot as plt

dataset_path = "./train-data.csv"

column_names = ['Ind', 'Name', 'Location', 'Year', 'Kilometers_Driven',
    'Fuel_Type', 'Transmission', 'Owner_Type', 'Mileage', 'Engine',
    'Power', 'Seats', 'New_Price', 'Price']
raw_dataset = pd.read_csv(dataset_path, names=column_names,
    na_values = "?", comment='\t', skiprows=1, sep=",",
    skipinitialspace=True)

dataset = raw_dataset.copy()
print ( dataset.head() )

dataset = dataset.drop(columns=['Ind', 'Name', 'Location', 'New_Price'])
print ( dataset.head() )

# To see a good description of the dataset

print ( dataset.describe() )

# Cleaning the data
# The dataset contains a few unknown values. Let’s find them and drop them.

dataset.isna().sum()
dataset = dataset.dropna()
dataset = dataset.reset_index(drop=True)

print ( dataset.head() )



dataset['Mileage'] = pd.Series([re.sub('[^.0-9]', '',
    str(val)) for val in dataset['Mileage']], index = dataset.index)
dataset['Engine'] = pd.Series([re.sub('[^.0-9]', '',
    str(val)) for val in dataset['Engine']], index = dataset.index)
dataset['Power'] = pd.Series([re.sub('[^.0-9]', '',
    str(val)) for val in dataset['Power']], index = dataset.index)
```

```
# The prices are by default in INR Lakhs. So, we have to convert them to USD

dataset['Price'] = pd.Series([int(float(val)*1521.22) for val in dataset['Price']],
        index = dataset.index)

print ( dataset.head() )

dataset = dataset.replace(r'^\s*$', np.nan, regex=True)
dataset.isna().sum()
dataset = dataset.dropna()

dataset = dataset.reset_index(drop=True)
print ( dataset.head() )

dataset['Mileage'] = pd.Series([int(float(str(val))*2.3521458)
    for val in dataset['Mileage']], index = dataset.index)
dataset['Engine'] = pd.Series([float(str(val))
    for val in dataset['Engine']], index = dataset.index)

## Lab 09 - TODO - for the column 'Power' in the dataset, convert it to a float
## Lab 09 - TODO - for the column 'Seats' in the dataset, convert it to a float
## Lab 09 - TODO - create the column 'Miles_Driven' from the column
##                'Kilometers_Driven' by converting to a float and
##                 Multiplying by 0.621371, then convert to an integer so
##                 that we don't have small fractional values.
##
##                 Example of Conversion in just code
##                 x = "23.0"      # A string, with a number in it.
##                 r = int(float(x)*0.621371)
##                     # Convert from string to float,
##                     # Km to Mi, then back to an integer.

dataset = dataset.drop(columns=['Kilometers_Driven'])

print ( dataset.head() )

dataset.to_csv(path_or_buf="new-car-data.csv")




## One-Hot the Fule_Type

print(dataset['Fuel_Type'].unique())

dataset['Fuel_Type'] = pd.Categorical(dataset['Fuel_Type'])
dfFuel_Type = pd.get_dummies(dataset['Fuel_Type'], prefix = 'Fuel_Type')
print ( dfFuel_Type.head() )

## One-Hot the Transmission
## Lab -09 - TODO - do a similar one-hot encoding for the values in
##                 the Transmission column.
## Lab -09 - TODO - do a similar one-hot encoding for the values in
## Lab - 09 - TODO - do a similar one-hot encoding for the values in
##                 the Owner_Type column.

## Concat it all together

## TODO - when you get the 2 sections above working you will need:
#### dataset = pd.concat([dataset, dfFuel_Type, dfTransmission, dfOwner_Type], axis=1)

## instead of just the dfFule_type
dataset = pd.concat([dataset, dfFuel_Type], axis=1)

dataset = dataset.drop(columns=['Owner_Type', 'Transmission', 'Fuel_Type'])
print ( dataset.head() )


# Save the data again - take a look at it.

dataset.to_csv(path_or_buf="new-car-data2.csv")

############################### ###############################
# Plot some stuff.
############################### ###############################


dataset.plot(kind='scatter',x='Price',y='Year',color='blue')
plt.show()

## Lab - 09 - TODO - Plot Price v.s. Miles_Driven
## Lab - 09 - TODO - Plot Price v.s. Power
## Lab - 09 - TODO - Plot Price v.s. Milage
## Lab - 09 - TODO - Plot Price v.s. Seats
```

And the datq

https://github.com/Univ-Wyo-Education/F21-1010/blob/main/class/lect/Lect-21/train-data.csv

Take the 3 sections with the TODO’s and implement them.

1. The conversion of columns from strings to float and from km to mi. Lines 62 to 73 in the file.
2. The one-hot encoding section. Lines 90 to 99 in the file.
3. The plots of data. Lines 120 to 123 in the file.

Turn in your finished code and a screen copy of the 4 plots.