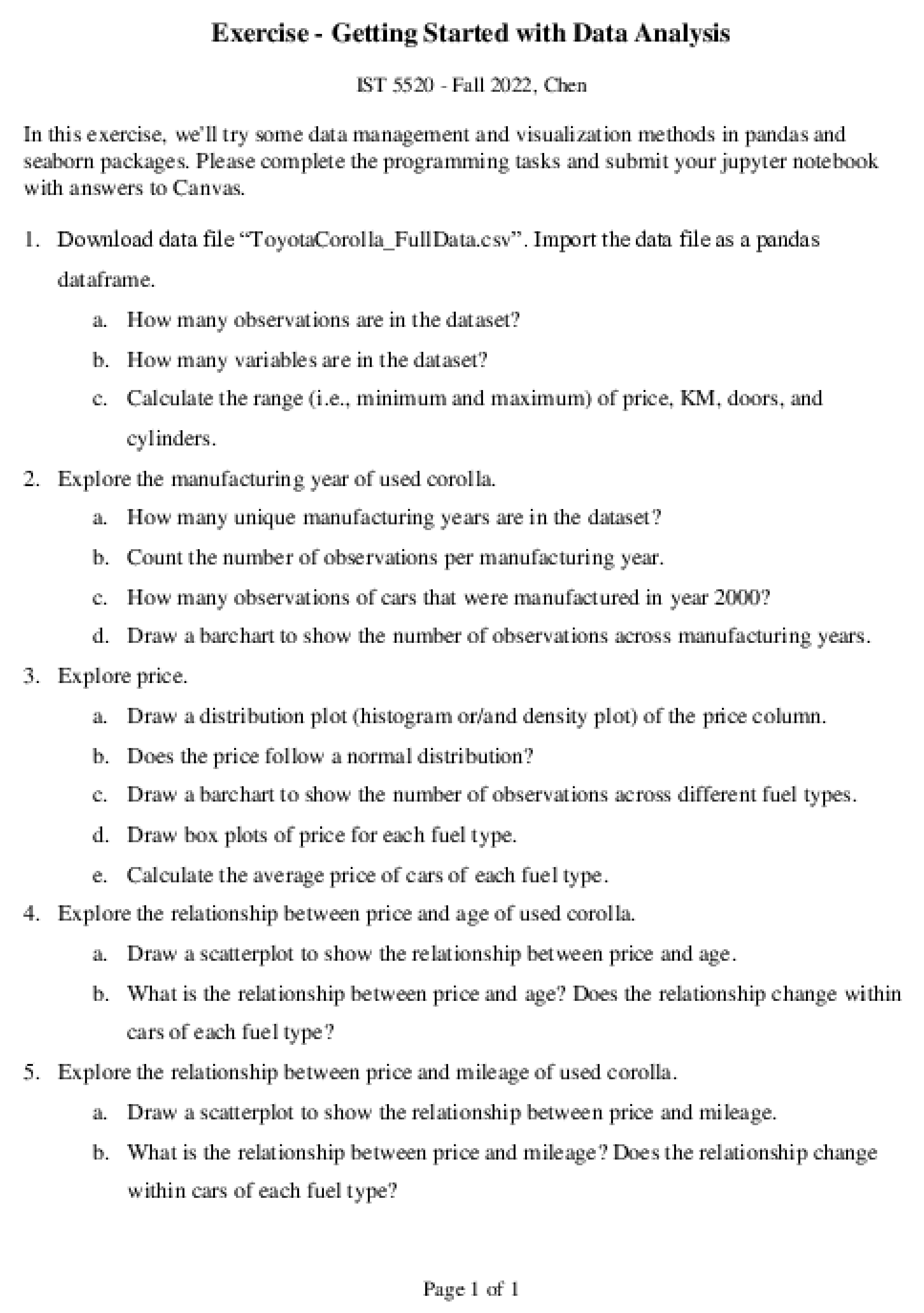
In [11]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from wand.image import Image as WImage
```

In [14]:

```python
img = WImage(filename = 'Lab - Getting Started with Data Analysis.pdf')
img
```

Out[14]:



## Exercise - Getting Started with Data Analysis

IST 5520 - Fall 2022, Chen

In this exercise, we'll try some data management and visualization methods in pandas and seaborn packages. Please complete the programming tasks and submit your jupyter notebook with answers to Canvas.

1. Download data file "ToyotaCorolla_FullData.csv". Import the data file as a pandas dataframe.
   a. How many observations are in the dataset?
   b. How many variables are in the dataset?
   c. Calculate the range (i.e., minimum and maximum) of price, KM, doors, and cylinders.
2. Explore the manufacturing year of used corolla.
   a. How many unique manufacturing years are in the dataset?
   b. Count the number of observations per manufacturing year.
   c. How many observations of cars that were manufactured in year 2000?
   d. Draw a barchart to show the number of observations across manufacturing years.
3. Explore price.
   a. Draw a distribution plot (histogram or/and density plot) of the price column.
   b. Does the price follow a normal distribution?
   c. Draw a barchart to show the number of observations across different fuel types.
   d. Draw box plots of price for each fuel type.
   e. Calculate the average price of cars of each fuel type.
4. Explore the relationship between price and age of used corolla.
   a. Draw a scatterplot to show the relationship between price and age.
   b. What is the relationship between price and age? Does the relationship change within cars of each fuel type?
5. Explore the relationship between price and mileage of used corolla.
   a. Draw a scatterplot to show the relationship between price and mileage.
   b. What is the relationship between price and mileage? Does the relationship change within cars of each fuel type?

Page 1 of 1

1. Download data file "ToyotaCorolla_FullData.csv". Import the data file as a pandas dataframe.

a. How many observations are in the dataset?

b. How many variables are in the dataset?

c. Calculate the range (i.e., minimum and maximum) of price, KM, doors, and cylinders.

In [17]:
```
df = pd.read_csv('ToyotaCorolla_FullData.csv')
df.head()
```

Out[17]:

|  | Id | Model | Price | Age_08_04 | Mfg_Month | Mfg_Year | KM | Fuel_Type | HP | Met_Color | ... | Pc |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | TOYOTA Corolla 2.0 D4D HATCHB TERRA 2/3-Doors | 13500 | 23 | 10 | 2002 | 46986 | Diesel | 90 | 1 | ... |  |
| **1** | 2 | TOYOTA Corolla 2.0 D4D HATCHB TERRA 2/3-Doors | 13750 | 23 | 10 | 2002 | 72937 | Diesel | 90 | 1 | ... |  |
| **2** | 3 | TOYOTA Corolla 2.0 D4D HATCHB TERRA 2/3-Doors | 13950 | 24 | 9 | 2002 | 41711 | Diesel | 90 | 1 | ... |  |
| **3** | 4 | TOYOTA Corolla 2.0 D4D HATCHB TERRA 2/3-Doors | 14950 | 26 | 7 | 2002 | 48000 | Diesel | 90 | 0 | ... |  |
| **4** | 5 | TOYOTA Corolla 2.0 D4D HATCHB SOL 2/3-Doors | 13750 | 30 | 3 | 2002 | 38500 | Diesel | 90 | 0 | ... |  |

5 rows × 39 columns

In [18]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1436 entries, 0 to 1435
Data columns (total 39 columns):
 #   Column             Non-Null Count  Dtype 
---  ------             --------------  ----- 
 0   Id                 1436 non-null   int64 
 1   Model              1436 non-null   object
 2   Price              1436 non-null   int64 
 3   Age_08_04          1436 non-null   int64 
 4   Mfg_Month          1436 non-null   int64 
 5   Mfg_Year           1436 non-null   int64 
 6   KM                 1436 non-null   int64 
 7   Fuel_Type          1436 non-null   object
 8   HP                 1436 non-null   int64 
 9   Met_Color          1436 non-null   int64 
 10  Color              1436 non-null   object
 11  Automatic          1436 non-null   int64 
 12  CC                 1436 non-null   int64 
 13  Doors              1436 non-null   int64 
 14  Cylinders          1436 non-null   int64 
 15  Gears              1436 non-null   int64 
 16  Quarterly_Tax      1436 non-null   int64 
 17  Weight             1436 non-null   int64 
 18  Mfr_Guarantee      1436 non-null   int64 
 19  BOVAG_Guarantee    1436 non-null   int64 
 20  Guarantee_Period   1436 non-null   int64 
 21  ABS                1436 non-null   int64 
 22  Airbag_1           1436 non-null   int64 
 23  Airbag_2           1436 non-null   int64 
 24  Airco              1436 non-null   int64 
 25  Automatic_airco    1436 non-null   int64 
 26  Boardcomputer      1436 non-null   int64 
 27  CD_Player          1436 non-null   int64 
 28  Central_Lock       1436 non-null   int64 
 29  Powered_Windows    1436 non-null   int64 
 30  Power_Steering     1436 non-null   int64 
 31  Radio              1436 non-null   int64 
 32  Mistlamps          1436 non-null   int64 
 33  Sport_Model        1436 non-null   int64 
 34  Backseat_Divider   1436 non-null   int64 
 35  Metallic_Rim       1436 non-null   int64 
 36  Radio_cassette     1436 non-null   int64 
 37  Parking_Assistant  1436 non-null   int64 
 38  Tow_Bar            1436 non-null   int64 
dtypes: int64(36), object(3)
memory usage: 437.7+ KB
```

### a: 1436 Observations

### b: 38 Variables

### c: range =

In [22]:

```python
for col in df.columns:
    print(f'Column: {col}, min:{df[col].min()}, max:{df[col].min()}')
```

```
Column: Id, min:1, max:1
Column: Model, min:TOYOTA Corolla , max:TOYOTA Corolla
Column: Price, min:4350, max:4350
Column: Age_08_04, min:1, max:1
Column: Mfg_Month, min:1, max:1
Column: Mfg_Year, min:1998, max:1998
Column: KM, min:1, max:1
Column: Fuel_Type, min:CNG, max:CNG
Column: HP, min:69, max:69
Column: Met_Color, min:0, max:0
Column: Color, min:Beige, max:Beige
Column: Automatic, min:0, max:0
Column: CC, min:1300, max:1300
Column: Doors, min:2, max:2
Column: Cylinders, min:4, max:4
Column: Gears, min:3, max:3
Column: Quarterly_Tax, min:19, max:19
Column: Weight, min:1000, max:1000
Column: Mfr_Guarantee, min:0, max:0
Column: BOVAG_Guarantee, min:0, max:0
Column: Guarantee_Period, min:3, max:3
Column: ABS, min:0, max:0
Column: Airbag_1, min:0, max:0
Column: Airbag_2, min:0, max:0
Column: Airco, min:0, max:0
Column: Automatic_airco, min:0, max:0
Column: Boardcomputer, min:0, max:0
Column: CD_Player, min:0, max:0
Column: Central_Lock, min:0, max:0
Column: Powered_Windows, min:0, max:0
Column: Power_Steering, min:0, max:0
Column: Radio, min:0, max:0
Column: Mistlamps, min:0, max:0
Column: Sport_Model, min:0, max:0
Column: Backseat_Divider, min:0, max:0
Column: Metallic_Rim, min:0, max:0
Column: Radio_cassette, min:0, max:0
Column: Parking_Assistant, min:0, max:0
Column: Tow_Bar, min:0, max:0
```

2. Explore the manufacturing year of used corolla.

a. How many unique manufacturing years are in the dataset?

b. Count the number of observations per manufacturing year.

c. How many observations of cars that were manufactured in year 2000?

d. Draw a barchart to show the number of observations across manufacturing years.

## a: 7 unique manufacturing years

In [23]:

```
print(len(df['Mfg_Year'].unique()))
df['Mfg_Year'].unique()
```

```
7
```

Out[23]:

```
array([2002, 2003, 2004, 2001, 2000, 1999, 1998])
```

## b:

In [25]:

```
sumCheck = 0
for year in df['Mfg_Year'].unique():
    obs = len(df[df['Mfg_Year']==year])
    print(f'Year: {year}, no. of Observation:{obs}')
    sumCheck += obs
sumCheck
```

```
Year: 2002, no. of Observation:87
Year: 2003, no. of Observation:75
Year: 2004, no. of Observation:24
Year: 2001, no. of Observation:192
Year: 2000, no. of Observation:225
Year: 1999, no. of Observation:441
Year: 1998, no. of Observation:392
```

Out[25]:

```
1436
```

## c: 225 cars in year 2000

3. Explore price.

a. Draw a distribution plot (histogram or/and density plot) of the price column.

b. Does the price follow a normal distribution?

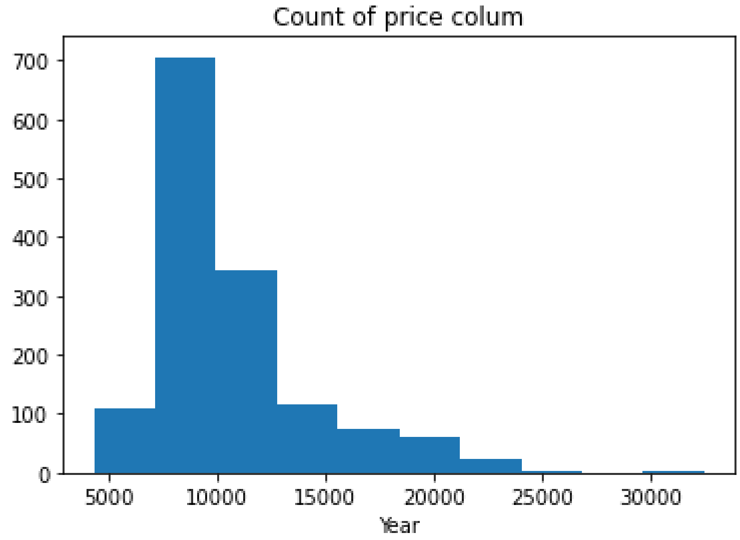
c. Draw a barchart to show the number of observations across different fuel types.

d. Draw box plots of price for each fuel type.

e. Calculate the average price of cars of each fuel type.

## a: Plot of price colum

```
In [32]:
plt.hist(df['Price']);
#matplotlib inline
plt.title('Count of price colum');
plt.xlabel('Year');
```

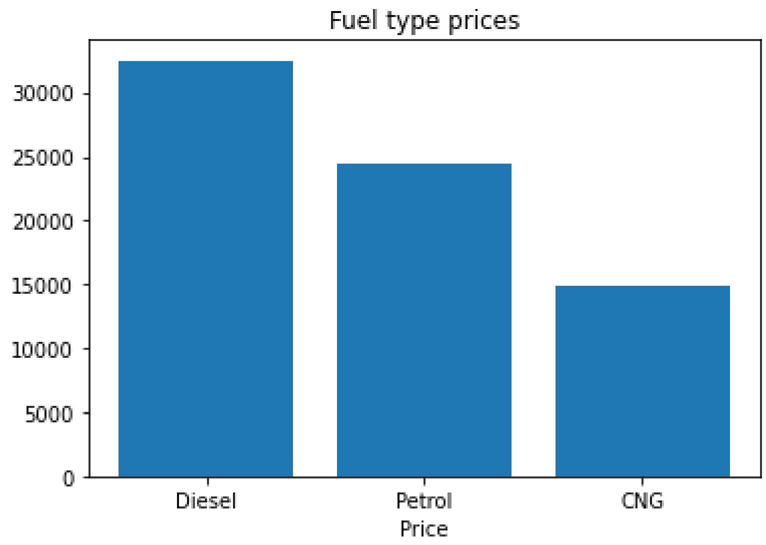


**b: No, the price follows a Skewed bell curve or Logarithmic curve.**

**c: Barchart of Fuel type prices.**

In [39]:

```
#plt.bar?
plt.bar(df['Fuel_Type'],df['Price']);
#matplotlib inline
plt.title('Fuel type prices');
plt.xlabel('Type');
plt.xlabel('Price');
```



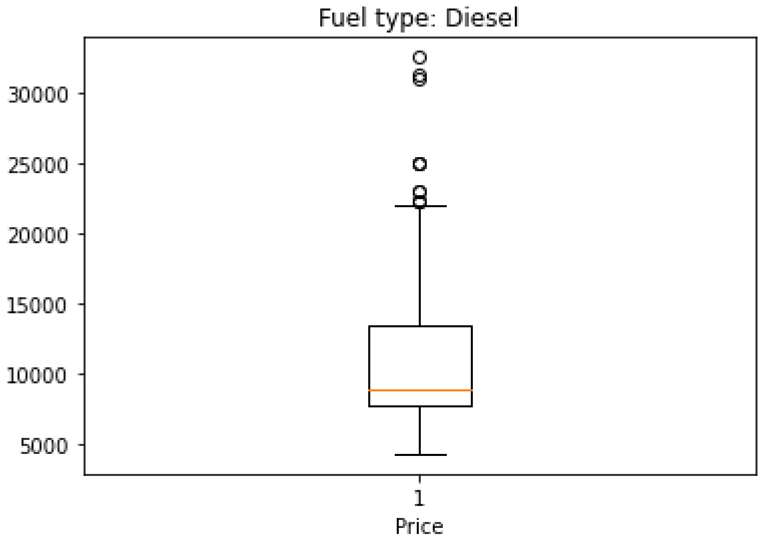
### d: Box plot for each fuel type

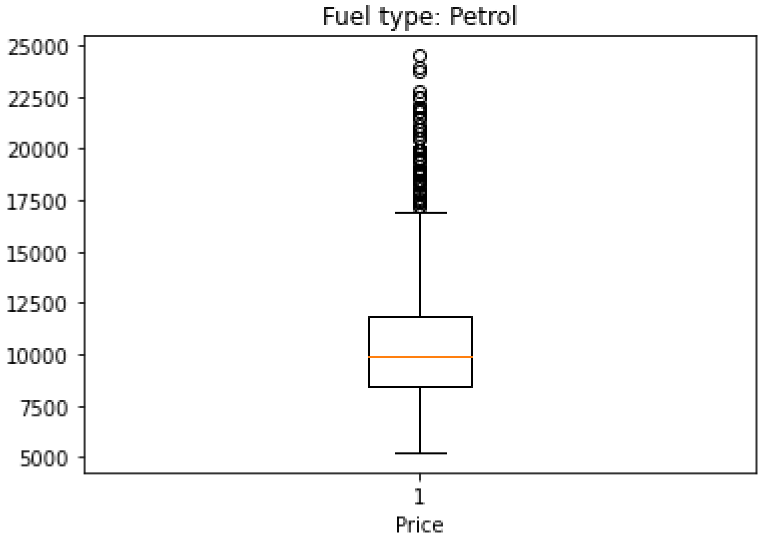
In [50]:

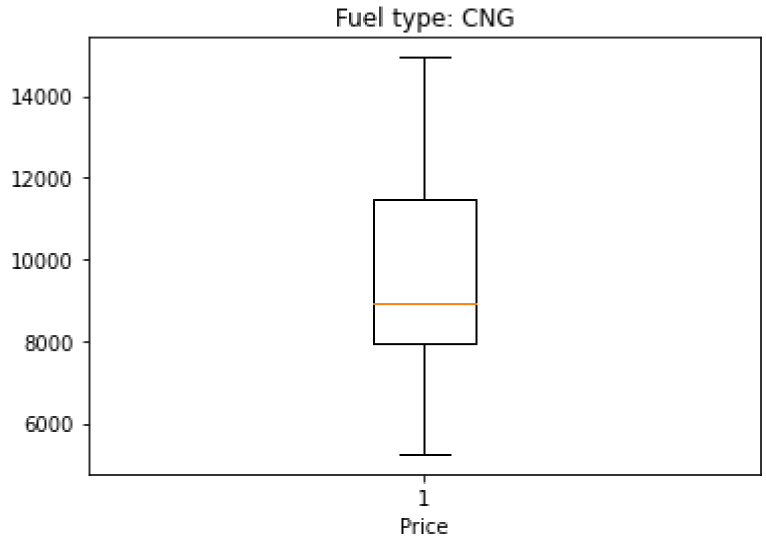
```
#df['Fuel_Type'].unique()
```

In [59]:

```
for fuel in df['Fuel_Type'].unique():

    #plt.box?
    #print(fuel)
    'sub data frame'
    #print(df[df['Fuel_Type']==fuel])
    'True/ False'
    #print(df['Fuel_Type']==fuel)

    #print(np.array(df[df['Fuel_Type']==fuel]['Price']))
    plt.boxplot( np.array(df[df['Fuel_Type']==fuel]['Price']) )
    plt.title(f'Fuel type: {fuel}');
    #plt.xlabel(f'Type')
    plt.xlabel(f'Price')

    #plt.close()
    'Allow all to display when using loop'
    plt.show()
plt.close()
```







## e: Average price of car per fuel type

In [67]:

```
for fuel in df['Fuel_Type'].unique():
    prices = df[df['Fuel_Type']==fuel]['Price']
    #print(prices)
    #print(np.array(prices))
    mean = np.array(prices).mean()
    print(f'Type: {fuel}, mean: ${np.round(mean,2)}')
```

```
Type: Diesel, mean: $11294.55
Type: Petrol, mean: $10679.31
Type: CNG, mean: $9421.18
```

4. Explore the relationship between price and age of used corolla.

a. Draw a scatterplot to show the relationship between price and age.

b. What is the relationship between price and age? Does the relationship change within cars of each fuel type?

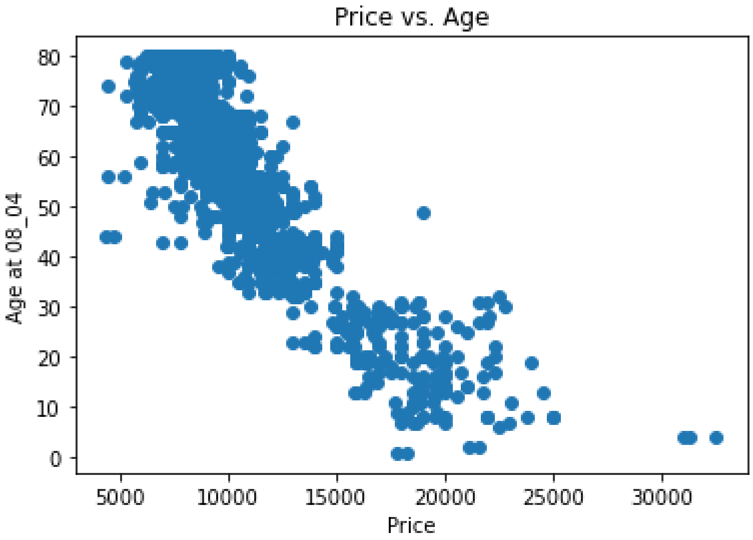
## a: Price vs Age scatter plot

In [71]:

```
plt.scatter(df['Price'],df['Age_08_04']);
plt.title('Price vs. Age')
plt.ylabel('Age at 08_04')
plt.xlabel('Price');
```

Out[71]:

```
Text(0.5, 0, 'Price')
```



```
### b: Negative trend: Price decrease linearly with increasing age.
```

5. Explore the relationship between price and mileage of used corolla.

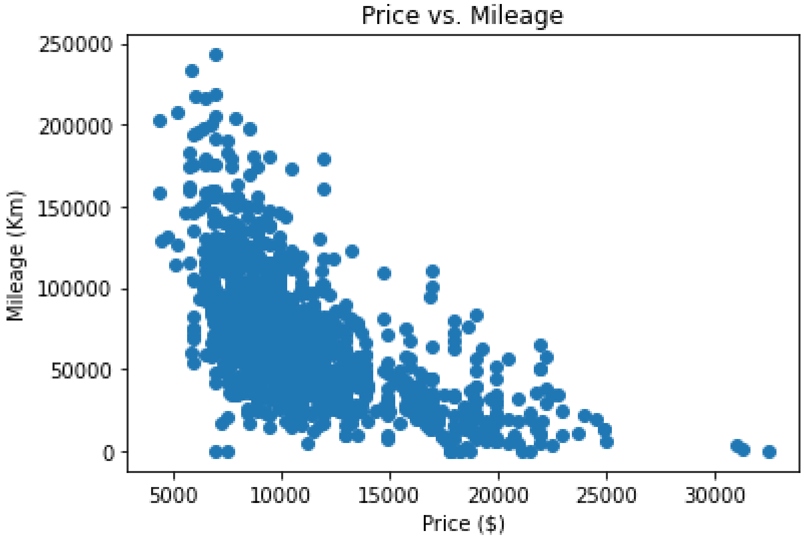
a. Draw a scatterplot to show the relationship between price and mileage.

b. What is the relationship between price and mileage? Does the relationship change within cars of each fuel type?

## a: Price vs Mileage of used corolla

In [75]:

```
plt.scatter(df['Price'],df['KM']);
plt.title('Price vs. Mileage')
plt.ylabel('Mileage (Km)')
plt.xlabel('Price ($)');
```

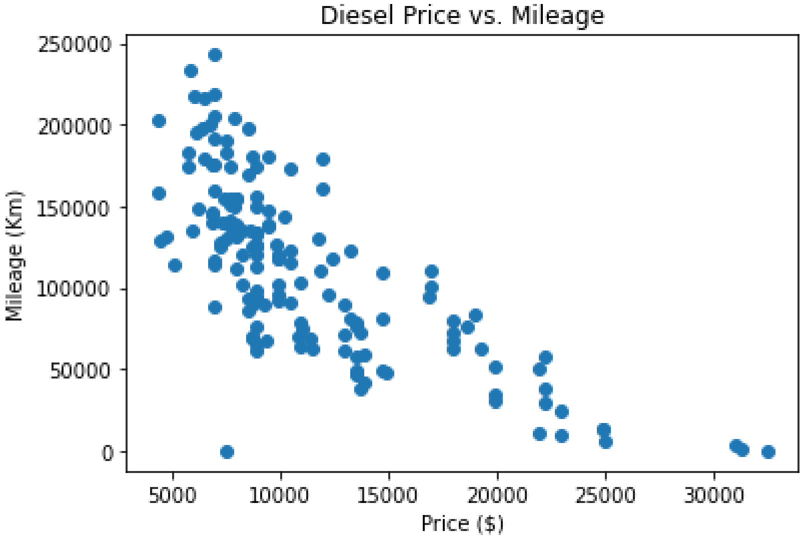


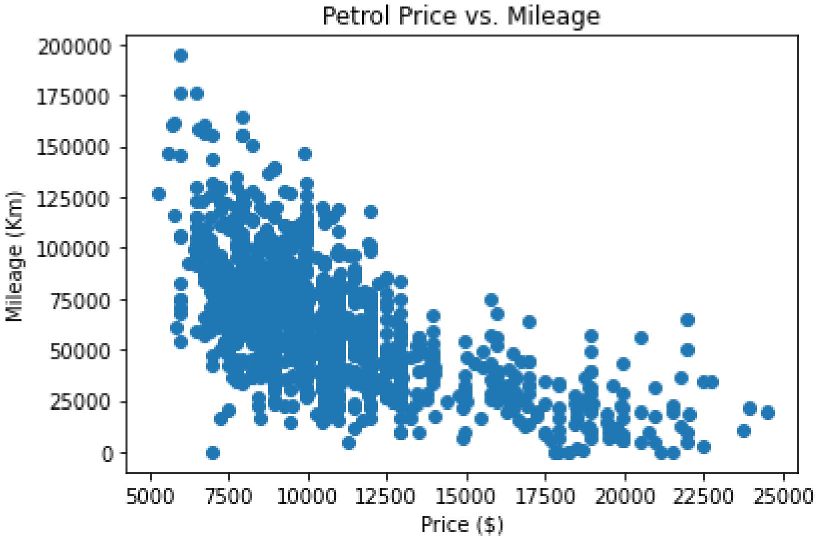
## b: Weak negative trend: Price decrease linearly with increasing mileage.

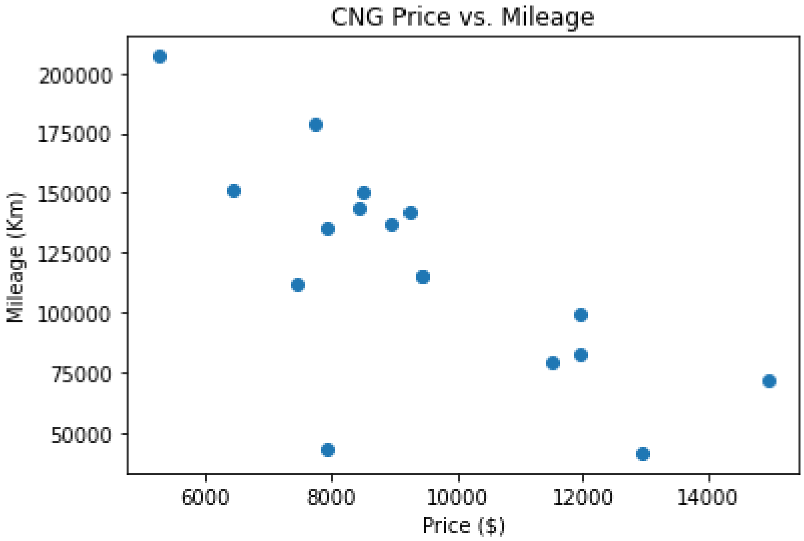
#### There is a higher correlation of a negative trend when comparing price with mileage per fuel type for Petrol.

In [77]:

```
for fuel in df['Fuel_Type'].unique():
    fuel_df = df[df['Fuel_Type']==fuel]
    plt.scatter(fuel_df['Price'],fuel_df['KM'])
    plt.title(f'{fuel} Price vs. Mileage')
    plt.ylabel(f'Mileage (Km)')
    plt.xlabel(f'Price ($)')
    plt.show()
```






CNG Price vs. Mileage
Mileage (Km)
200000
175000
150000
125000
100000
75000
50000
6000
8000
10000
12000
14000
Price ($)

In [ ]: