Collect the appropriate business dataset from any repository and analyze the summary of the dataset and further divide dataset for training and testing purpose.

a. Download the official dataset and import it as input.

b. Give the details of the dataset such as no of samples, no of features, target vector etc by using python command.

c. By using python command, divided the dataset into training, testing dataset and target dataset for training and testing along with validation dataset.

In [ ]:
```
from sklearn.model_selection import train_test_split
import pandas as pd
import numpy as np
import seaborn as sns
```

In [ ]:
```
df = pd.read_csv("https://github.com/YBI-Foundation/Dataset/raw/main/Big%20Sales%20
df.head(2)
```

Out[ ]:

	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Id
0	FDT36	12.3	Low Fat	0.111448	Baking Goods	33.4874	
1	FDT36	12.3	Low Fat	0.111904	Baking Goods	33.9874	

In [ ]:
```
df.describe()
```

Out[ ]:

	Item_Weight	Item_Visibility	Item_MRP	Outlet_Establishment_Year	Item_Outlet_Sales
count	11815.000000	14204.000000	14204.000000	14204.000000	14204.000000
mean	12.788355	0.065953	141.004977	1997.830681	2185.836320
std	4.654126	0.051459	62.086938	8.371664	1827.479550
min	4.555000	0.000000	31.290000	1985.000000	33.290000
25%	8.710000	0.027036	94.012000	1987.000000	922.135101
50%	12.500000	0.054021	142.247000	1999.000000	1768.287680
75%	16.750000	0.094037	185.855600	2004.000000	2988.110400
max	30.000000	0.328391	266.888400	2009.000000	31224.726950

In [ ]:
```
df.isna().sum()
```

Out[ ]:
```
Item_Identifier                 0
Item_Weight                  2389
Item_Fat_Content                0
Item_Visibility                 0
Item_Type                       0
Item_MRP                        0
Outlet_Identifier               0
Outlet_Establishment_Year       0
Outlet_Size                     0
Outlet_Location_Type            0
Outlet_Type                     0
Item_Outlet_Sales               0
dtype: int64
```

In [ ]:
```
df.columns
```

Out[ ]:
```
Index(['Item_Identifier', 'Item_Weight', 'Item_Fat_Content', 'Item_Visibility',
       'Item_Type', 'Item_MRP', 'Outlet_Identifier',
       'Outlet_Establishment_Year', 'Outlet_Size', 'Outlet_Location_Type',
       'Outlet_Type', 'Item_Outlet_Sales'],
      dtype='object')
```

In [ ]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14204 entries, 0 to 14203
Data columns (total 12 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Item_Identifier            14204 non-null  object
 1   Item_Weight                11815 non-null  float64
 2   Item_Fat_Content           14204 non-null  object
 3   Item_Visibility            14204 non-null  float64
 4   Item_Type                  14204 non-null  object
 5   Item_MRP                   14204 non-null  float64
 6   Outlet_Identifier          14204 non-null  object
 7   Outlet_Establishment_Year  14204 non-null  int64
 8   Outlet_Size                14204 non-null  object
 9   Outlet_Location_Type       14204 non-null  object
 10  Outlet_Type                14204 non-null  object
 11  Item_Outlet_Sales          14204 non-null  float64
dtypes: float64(4), int64(1), object(7)
memory usage: 1.3+ MB
```

In [ ]:
```
df.shape
```

Out[ ]:
```
(14204, 12)
```

In [ ]:
```
df.corr()['Item_Outlet_Sales']
```

Out[ ]:
```
Item_Weight                 0.228297
Item_Visibility            -0.158813
Item_MRP                    0.532261
Outlet_Establishment_Year  -0.110786
Item_Outlet_Sales           1.000000
Name: Item_Outlet_Sales, dtype: float64
```

In [ ]:
```
X= df.drop(['Item_Identifier','Outlet_Establishment_Year','Item_Outlet_Sales'],axis
Y = df['Item_Outlet_Sales'] # target variable
```

In [ ]:
```
x_main,x_test,y_main,y_test = train_test_split(X,Y,random_state=42,test_size=0.2)
```

In [ ]:
```
x_train,x_val,y_train,y_val = train_test_split(x_main,y_main,random_state=42,test_s
```

In [ ]:
```
x_train.shape, y_train.shape, x_val.shape, y_val.shape, x_test.shape,y_test.shape
```

Out[ ]:
```
((9090, 9), (9090,), (2273, 9), (2273,), (2841, 9), (2841,))
```

In [ ]:
```
print('training set: ', x_train.shape, y_train.shape)
print('validation set: ', x_val.shape, y_val.shape)
print('testing set: ',x_test.shape,y_test.shape)
```

```
training set:  (9090, 9) (9090,)
validation set:  (2273, 9) (2273,)
testing set:  (2841, 9) (2841,)
```