In [1]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [9]:

```
data = pd.read_csv("women_entreprenuer.csv")
```

In [10]:

```
data
```

Out[10]:

	No	Country	Level of development	European Union Membership	Currency	Women Entrepreneurship Index	Entrepreneurship Index	Inflation rate	Labo Partic
0	4	Austria	Developed	Member	Euro	54.9	64.9	0.90	
1	6	Belgium	Developed	Member	Euro	63.6	65.5	0.60	
2	17	Estonia	Developed	Member	Euro	55.4	60.2	-0.88	
3	18	Finland	Developed	Member	Euro	66.4	65.7	-0.20	
4	19	France	Developed	Member	Euro	68.8	67.3	0.00	
5	20	Germany	Developed	Member	Euro	63.6	67.4	0.50	
6	22	Greece	Developed	Member	Euro	43.0	42.0	-1.70	

In [11]:

```
data.head()
```

Out[11]:

	No	Country	Level of development	European Union Membership	Currency	Women Entrepreneurship Index	Entrepreneurship Index	Inflatio rat
0	4	Austria	Developed	Member	Euro	54.9	64.9	0.9
1	6	Belgium	Developed	Member	Euro	63.6	65.5	0.6
2	17	Estonia	Developed	Member	Euro	55.4	60.2	-0.8
3	18	Finland	Developed	Member	Euro	66.4	65.7	-0.2
4	19	France	Developed	Member	Euro	68.8	67.3	0.0

In [12]:

```
data.tail()
```

Out[12]:

	No	Country	Level of development	European Union Membership	Currency	Women Entrepreneurship Index	Entrepreneurship Index	Inflat ra
46	48	Saudi Arabia	Developing	Not Member	National Currency	37.0	49.6	1
47	57	Thailand	Developing	Not Member	National Currency	36.6	32.1	-0
48	58	Tunisia	Developing	Not Member	National Currency	30.7	35.5	4
49	59	Turkey	Developing	Not Member	National Currency	39.3	54.6	7
50	60	Uruguay	Developing	Not Member	National Currency	44.5	41.4	8

In [13]:

```
data.shape
```

Out[13]:

```
(51, 9)
```

In [14]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 51 entries, 0 to 50
Data columns (total 9 columns):
 #   Column                                Non-Null Count  Dtype  
---  ------                                --------------  -----  
 0   No                                    51 non-null     int64  
 1   Country                               51 non-null     object 
 2   Level of development                  51 non-null     object 
 3   European Union Membership             51 non-null     object 
 4   Currency                              51 non-null     object 
 5   Women Entrepreneurship Index          51 non-null     float64
 6   Entrepreneurship Index                51 non-null     float64
 7   Inflation rate                        51 non-null     float64
 8   Female Labor Force Participation Rate  51 non-null     float64
dtypes: float64(4), int64(1), object(4)
memory usage: 3.7+ KB
```

In [15]:

```
data.describe()
```

Out[15]:

	No	Women Entrepreneurship Index	Entrepreneurship Index	Inflation rate	Female Labor Force Participation Rate
count	51.000000	51.000000	51.000000	51.000000	51.000000
mean	29.980392	47.835294	47.241176	2.587647	58.481765
std	18.017203	14.268480	16.193149	5.380639	13.864567
min	1.000000	25.300000	24.800000	-2.250000	13.000000
25%	14.500000	36.350000	31.900000	-0.500000	55.800000
50%	30.000000	44.500000	42.700000	0.600000	61.000000
75%	45.500000	59.150000	65.400000	3.600000	67.400000
max	60.000000	74.800000	77.600000	26.500000	82.300000

In [16]:

```
data.describe().columns
```

Out[16]:

```
Index(['No', 'Women Entrepreneurship Index', 'Entrepreneurship Index',
       'Inflation rate', 'Female Labor Force Participation Rate'],
      dtype='object')
```

In [17]:

```
data.isnull().sum()
```

Out[17]:

```
No                                    0
Country                               0
Level of development                  0
European Union Membership             0
Currency                              0
Women Entrepreneurship Index          0
Entrepreneurship Index                0
Inflation rate                        0
Female Labor Force Participation Rate    0
dtype: int64
```

In [18]:

```
import warnings
warnings.filterwarnings("ignore")
```

In [19]:

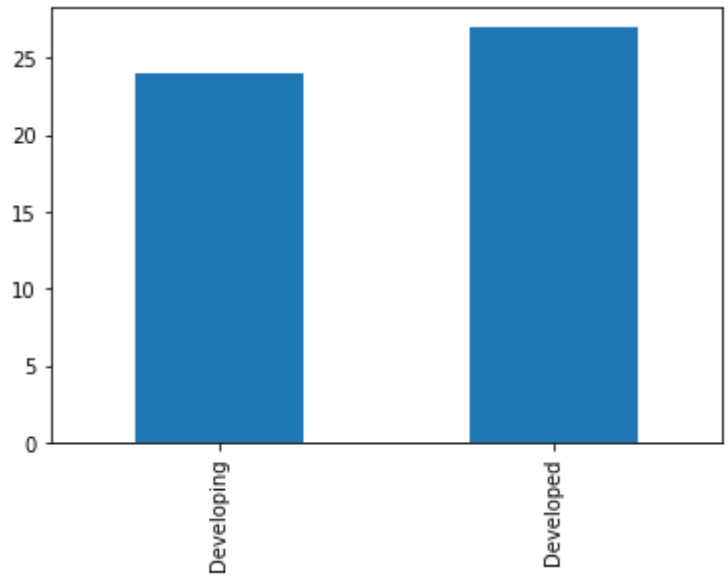
```
data_cat = data[['Country', 'Level of development', 'European Union Membership', 'Currer
data_num = data[['Women Entrepreneurship Index', 'Entrepreneurship Index',
                 'Inflation rate', 'Female Labor Force Participation Rate']]
```

In [20]:

```
data_cat['Level of development'].value_counts().sort_index(ascending=False).plot(kind='b
```

Out[20]:

```
<matplotlib.axes._subplots.AxesSubplot at 0xe49d27e670>
```

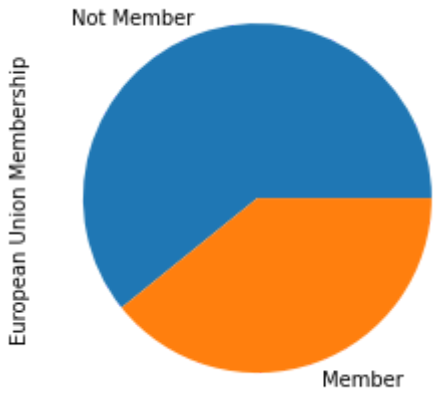


In [21]:

```
a_cat['European Union Membership'].value_counts().sort_index(ascending=False).plot(kind=
```

Out[21]:

```
<matplotlib.axes._subplots.AxesSubplot at 0xe49d386130>
```

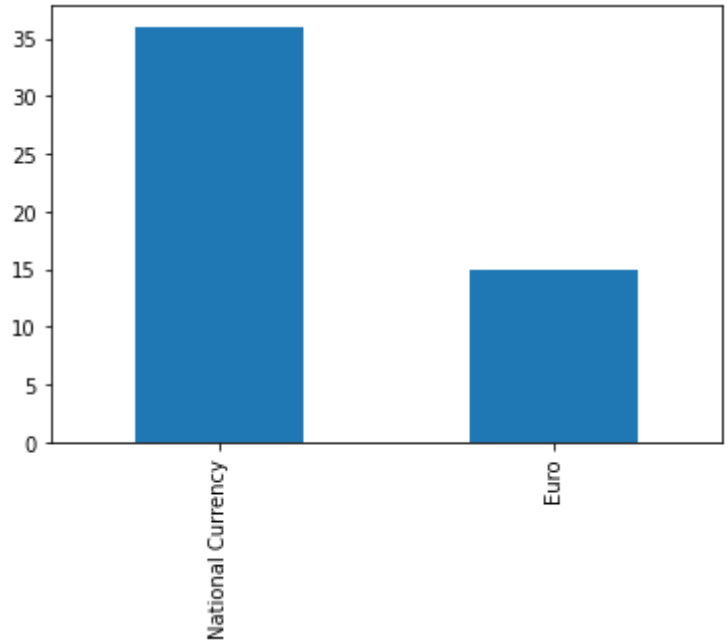


In [23]:

```
data_cat['Currency'].value_counts().sort_index(ascending=False).plot(kind='bar')
```

Out[23]:

```
<matplotlib.axes._subplots.AxesSubplot at 0xe4b1e9b910>
```



In [29]:

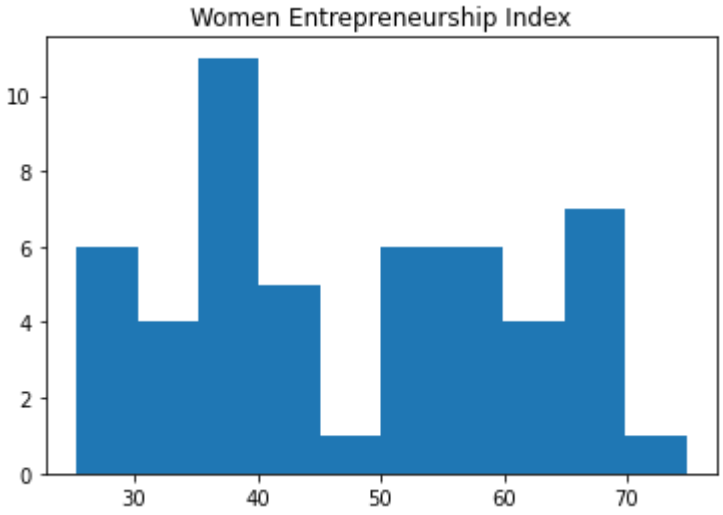
```
table = data_cat.groupby(['Currency', 'Level of development']).count()
table
```

Out[29]:

		Country	European Union Membership
Currency	**Level of development**		
Euro	**Developed**	15	15
National Currency	**Developed**	12	12
	Developing	24	24

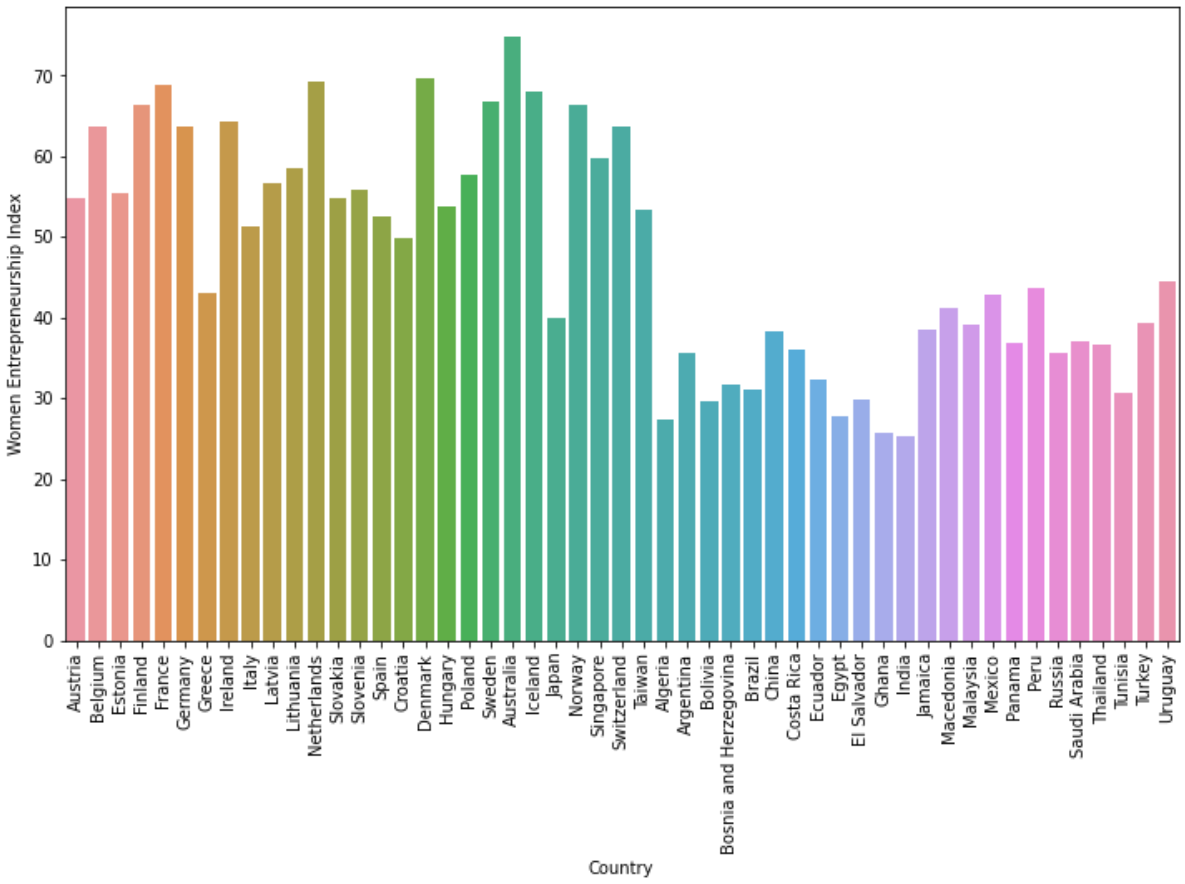
In [30]:

```
plt.hist(data_num['Women Entrepreneurship Index'], bins=10)
plt.title("Women Entrepreneurship Index")
plt.show()
```



In [35]:

```
plt.figure(figsize = (12,7))
sns.barplot(x = data_cat.Country, y = data_num['Women Entrepreneurship Index'])
plt.xticks(rotation=90)
plt.show()
```



In [47]:

```
top_WEI = data.groupby('Country').sum()
top_WEI.drop('No', axis = 1, inplace = True)
top_WEI.sort_values(by = 'Women Entrepreneurship Index', ascending = False).head()
```

Out[47]:

Country	Women Entrepreneurship Index	Entrepreneurship Index	Inflation rate	Female Labor Force Participation Rate
Australia	74.8	77.6	1.5	66.8
Denmark	69.7	71.4	0.5	70.3
Netherlands	69.3	66.5	0.6	69.2
France	68.8	67.3	0.0	60.6
Iceland	68.0	70.4	1.6	82.3

In [45]:

```
top_LOD = data.groupby('Level of development').sum()
top_LOD.drop('No', axis = 1, inplace = True)
top_LOD.sort_values(by = 'Women Entrepreneurship Index', ascending = False)
```

Out[45]:

Level of development	Women Entrepreneurship Index	Entrepreneurship Index	Inflation rate	Female Labor Force Participation Rate
Developed	1602.7	1606.1	0.38	1703.58
Developing	836.9	803.2	131.59	1278.99

In [48]:

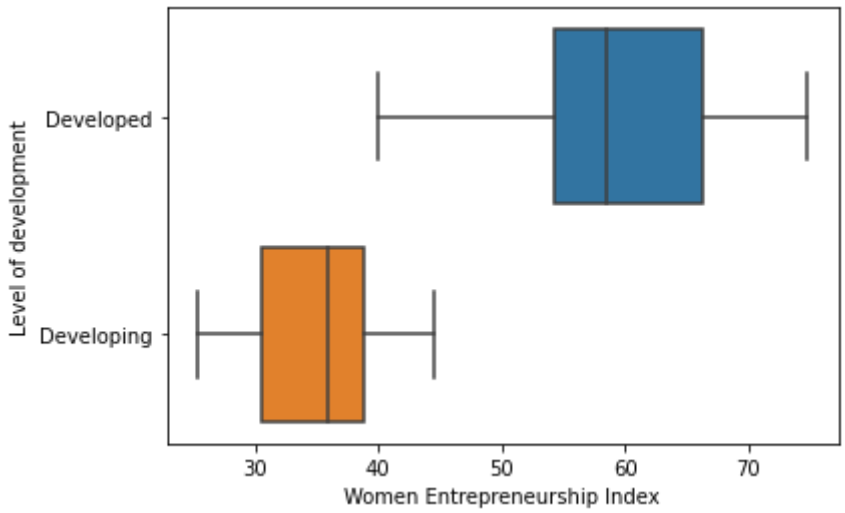
```
top_WEI = data.groupby('Country').sum()
top_WEI.drop('No', axis = 1, inplace = True)
top_WEI.sort_values(by = 'Women Entrepreneurship Index', ascending = False).tail()
```

Out[48]:

Country	Women Entrepreneurship Index	Entrepreneurship Index	Inflation rate	Female Labor Force Participation Rate
Bolivia	29.7	28.0	4.1	69.4
Egypt	27.7	28.1	11.0	64.6
Algeria	27.4	30.2	4.8	18.0
Ghana	25.8	24.8	17.2	60.8
India	25.3	25.3	5.9	61.1

In [49]:

```
sns.boxplot(x='Women Entrepreneurship Index', y='Level of development', data=data);
```



In [58]:

```
top_IR = data.groupby('Country').sum()
top_IR.drop(['No', 'Women Entrepreneurship Index', 'Entrepreneurship Index',
             'Female Labor Force Participation Rate'], axis = 1, inplace = True)
top_IR.sort_values(by = 'Inflation rate', ascending = False).head()
```

Out[58]:

Country	Inflation rate
Argentina	26.50
Ghana	17.20
Russia	15.50
Egypt	11.00
Brazil	10.67

In [59]:

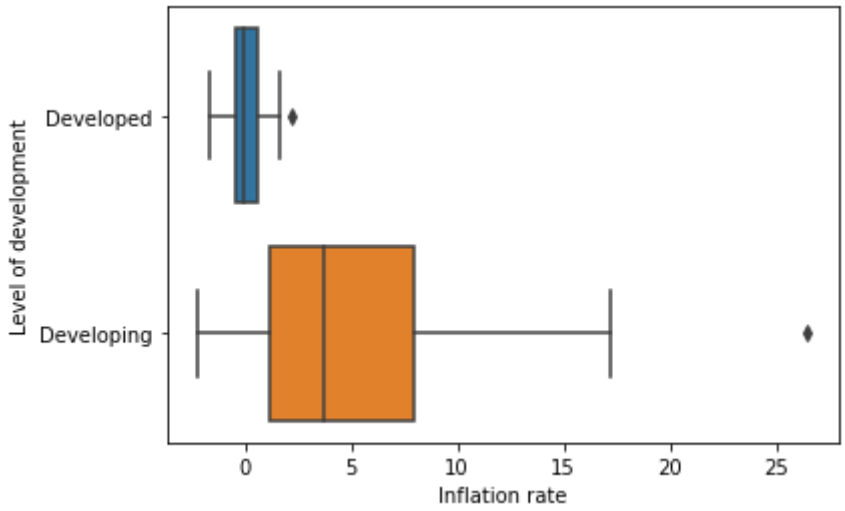
```
top_IR = data.groupby('Country').sum()
top_IR.drop(['No', 'Women Entrepreneurship Index', 'Entrepreneurship Index',
             'Female Labor Force Participation Rate'], axis = 1, inplace = True)
top_IR.sort_values(by = 'Inflation rate', ascending = False).tail()
```

Out[59]:

Country	Inflation rate
Lithuania	-0.90
Bosnia and Herzegovina	-1.00
Switzerland	-1.10
Greece	-1.70
El Salvador	-2.25

In [60]:

```
sns.boxplot(x='Inflation rate', y='Level of development', data=data);
```



In [61]:

```
top_FLPR = data.groupby('Country').sum()
top_FLPR.drop(['No', 'Women Entrepreneurship Index', 'Entrepreneurship Index',
               'Inflation rate'], axis = 1, inplace = True)
top_FLPR.sort_values(by = 'Female Labor Force Participation Rate', ascending = False).he
```

Out[61]:

Country	Female Labor Force Participation Rate
Iceland	82.3
Switzerland	74.7
Sweden	74.0
Macedonia	73.0
Denmark	70.3

In [62]:

```
top_FLPR = data.groupby('Country').sum()
top_FLPR.drop(['No', 'Women Entrepreneurship Index', 'Entrepreneurship Index',
               'Inflation rate'], axis = 1, inplace = True)
top_FLPR.sort_values(by = 'Female Labor Force Participation Rate', ascending = False).ta
```

Out[62]:

Country	Female Labor Force Participation Rate
Jamaica	37.70
Turkey	30.40
Tunisia	25.19
Algeria	18.00
Saudi Arabia	13.00

In [63]:

```
x = data[['Inflation rate']].values
y = data['Women Entrepreneurship Index'].values
```

In [64]:

```
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size = 0.2)
```

In [72]:

```
model= LinearRegression()
reg = model.fit(X_train, y_train)
```

In [79]:

```
!pip install statsmodels
```

```
WARNING: You are using pip version 22.0.3; however, version 22.0.4 is avai
lable.
You should consider upgrading via the 'c:\python\python.exe -m pip install
--upgrade pip' command.

Collecting statsmodels
  Downloading statsmodels-0.13.2-cp38-cp38-win_amd64.whl (9.1 MB)
     ---------------------------------------- 9.1/9.1 MB 514.7 kB/s eta 0:
00:00
Collecting patsy>=0.5.2
  Downloading patsy-0.5.2-py2.py3-none-any.whl (233 kB)
     ---------------------------------------- 233.7/233.7 KB 386.6 kB/s eta 0:
00:00
Requirement already satisfied: numpy>=1.17 in c:\python\lib\site-packages
(from statsmodels) (1.19.4)
Collecting packaging>=21.3
  Downloading packaging-21.3-py3-none-any.whl (40 kB)
     ---------------------------------------- 40.8/40.8 KB 391.2 kB/s eta 0:
00:00
Requirement already satisfied: pandas>=0.25 in c:\python\lib\site-packages
(from statsmodels) (1.1.3)
Requirement already satisfied: scipy>=1.3 in c:\python\lib\site-packages
(from statsmodels) (1.4.1)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in c:\python\lib\s
ite-packages (from packaging>=21.3->statsmodels) (2.4.7)
Requirement already satisfied: python-dateutil>=2.7.3 in c:\python\lib\sit
e-packages (from pandas>=0.25->statsmodels) (2.8.1)
Requirement already satisfied: pytz>=2017.2 in c:\python\lib\site-packages
(from pandas>=0.25->statsmodels) (2020.1)
Requirement already satisfied: six in c:\python\lib\site-packages (from pa
tsy>=0.5.2->statsmodels) (1.12.0)
Installing collected packages: patsy, packaging, statsmodels
  Attempting uninstall: packaging
    Found existing installation: packaging 20.4
    Uninstalling packaging-20.4:
      Successfully uninstalled packaging-20.4
Successfully installed packaging-21.3 patsy-0.5.2 statsmodels-0.13.2
```

In [80]:

```
import statsmodels.api as sm
sm.OLS(y, sm.add_constant(x)).fit()
```

In [81]:

```
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
====
Dep. Variable:                      y   R-squared:
0.208
Model:                            OLS   Adj. R-squared:
0.191
Method:                 Least Squares   F-statistic:                     1
2.83
Date:                Sat, 12 Mar 2022   Prob (F-statistic):           0.00
0782
Time:                        12:19:40   Log-Likelihood:                -20
1.49
No. Observations:                  51   AIC:                             4
07.0
Df Residuals:                      49   BIC:                             4
10.8
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
====
                 coef    std err          t      P>|t|      [0.025      0.
975]
------------------------------------------------------------------------------
----
const         50.9611      1.997     25.513      0.000      46.947      5
4.975
x1            -1.2080      0.337     -3.582      0.001      -1.886      -
0.530
==============================================================================
====
Omnibus:                        5.788   Durbin-Watson:
1.168
Prob(Omnibus):                  0.055   Jarque-Bera (JB):
2.206
Skew:                           0.054   Prob(JB):
0.332
Kurtosis:                       1.987   Cond. No.
6.62
==============================================================================
====

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is cor
rectly specified.
```

In [82]:

```
x1 = data[['Female Labor Force Participation Rate']].values
y1 = data['Women Entrepreneurship Index'].values
```

In [84]:

```
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(x1, y1, test_size = 0.2)
```

In [85]:

```
model1= LinearRegression()
model1.fit(X_train, y_train)
```

In [88]:

```
model1 = sm.OLS(y1, sm.add_constant(x1)).fit()
```

In [89]:

```
print(model1.summary())
```

```
                            OLS Regression Results
==============================================================================
====
Dep. Variable:                      y   R-squared:
0.195
Model:                            OLS   Adj. R-squared:
0.178
Method:                 Least Squares   F-statistic:                     1
1.86
Date:                Sat, 12 Mar 2022   Prob (F-statistic):            0.0
0119
Time:                        12:23:47   Log-Likelihood:                -20
1.90
No. Observations:                  51   AIC:                             4
07.8
Df Residuals:                      49   BIC:                             4
11.7
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
====
                 coef    std err          t      P>|t|      [0.025      0.
975]
------------------------------------------------------------------------------
----
const         21.2711      7.925      2.684      0.010       5.346      3
7.197
x1             0.4542      0.132      3.443      0.001       0.189
0.719
==============================================================================
====
Omnibus:                        7.393   Durbin-Watson:
0.751
Prob(Omnibus):                  0.025   Jarque-Bera (JB):
3.015
Skew:                          -0.286   Prob(JB):
0.221
Kurtosis:                       1.955   Cond. No.
263.
==============================================================================
====

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is cor
rectly specified.
```

In [90]:

```
x2 = data[['Entrepreneurship Index']].values
y2 = data['Women Entrepreneurship Index'].values
```

In [92]:

```
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(x2, y2, test_size = 0.2)
```

In [93]:

```
model2= LinearRegression()
model2.fit(X_train, y_train)
```

Out[93]:

```
LinearRegression()
```

In [94]:

```
model2 = sm.OLS(y2, sm.add_constant(x2)).fit()
```

In [95]:

```
print(model2.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.836
Model:                            OLS   Adj. R-squared:                  0.833
Method:                 Least Squares   F-statistic:                     250.6
Date:                Sat, 12 Mar 2022   Prob (F-statistic):           6.70e-21
Time:                        12:27:02   Log-Likelihood:                -161.25
No. Observations:                  51   AIC:                             326.5
Df Residuals:                      49   BIC:                             330.4
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          9.7648      2.540      3.845      0.000       4.661      14.868
x1             0.8059      0.051     15.831      0.000       0.704       0.908
==============================================================================
Omnibus:                        1.701   Durbin-Watson:                   1.389
Prob(Omnibus):                  0.427   Jarque-Bera (JB):                1.422
Skew:                          -0.406   Prob(JB):                        0.491
Kurtosis:                       2.896   Cond. No.                         155.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

In [96]:

```
x3 = data[['Inflation rate', 'Female Labor Force Participation Rate']].values
y3 = data['Women Entrepreneurship Index'].values
model3 = sm.OLS(y3, sm.add_constant(x3)).fit()
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
====
Dep. Variable:                      y   R-squared:
0.208
Model:                            OLS   Adj. R-squared:
0.191
Method:                 Least Squares   F-statistic:                     1
2.83
Date:                Sat, 12 Mar 2022   Prob (F-statistic):           0.00
0782
Time:                        12:29:39   Log-Likelihood:                -20
1.49
No. Observations:                  51   AIC:                             4
07.0
Df Residuals:                      49   BIC:                             4
10.8
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
====
                 coef    std err          t      P>|t|      [0.025      0.
975]
------------------------------------------------------------------------------
----
const         50.9611      1.997     25.513      0.000      46.947      5
4.975
x1            -1.2080      0.337     -3.582      0.001      -1.886      -
0.530
==============================================================================
====
Omnibus:                        5.788   Durbin-Watson:
1.168
Prob(Omnibus):                  0.055   Jarque-Bera (JB):
2.206
Skew:                           0.054   Prob(JB):
0.332
Kurtosis:                       1.987   Cond. No.
6.62
==============================================================================
====

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is cor
rectly specified.
```

In [97]:

```
X_train, X_test, y_train, y_test = train_test_split(x3, y3, test_size = 0.2)
```

In [99]:

```
model3 = LinearRegression()
model3.fit(X_train, y_train)
```

Out[99]:

```
LinearRegression()
```

In [100]:

```
print("Training Accuracy :", model3.score(X_train, y_train))
print("Testing Accuracy :", model3.score(X_test, y_test))
```

```
Training Accuracy : 0.36944938457602594
Testing Accuracy : 0.07361627890272748
```