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

```
dataset=pd.read_csv("/content/train.csv")
print(dataset)
dataset.shape
```

```
           x          y
0    24.0  21.549452
1    50.0  47.464463
2    15.0  17.218656
3    38.0  36.586398
4    87.0  87.288984
..    ...        ...
695  58.0  58.595006
696  93.0  94.625094
697  82.0  88.603770
698  66.0  63.648685
699  97.0  94.975266

[700 rows x 2 columns]
(700, 2)
```

```
type(dataset)
```

```
pandas.core.frame.DataFrame
```

```
dataset.shape
```

```
(700, 2)
```

```
dataset.describe()
```

	x	y
count	700.000000	699.000000
mean	54.985939	49.939869
std	134.681703	29.109217
min	0.000000	-3.839981
25%	25.000000	24.929968
50%	49.000000	48.973020
75%	75.000000	74.929911
max	3530.157369	108.871618

```
x=dataset.iloc[0:700,0:1]
```

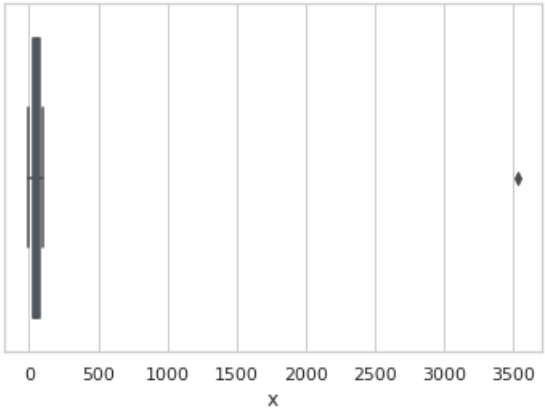
```
y=dataset.iloc[0:700,1:2]
```

```
numpy.ndarray
```

```
import seaborn as sns
```

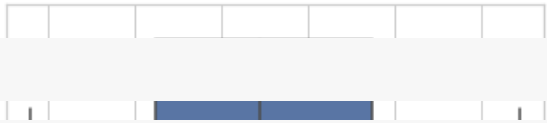
```
sns.boxplot(dataset['x'])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass the following var
  FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f93b477f250>
```




```
sns.boxplot(dataset['y'])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass the following var
  FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f93b47c9190>
```



```
!pip install kaggle
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: kaggle in /usr/local/lib/python3.7/dist-packages (1.5.12)
Requirement already satisfied: six>=1.10 in /usr/local/lib/python3.7/dist-packages (from kaggle) (1.15.0)
Requirement already satisfied: python-slugify in /usr/local/lib/python3.7/dist-packages (from kaggle) (6.1.2)
Requirement already satisfied: tqdm in /usr/local/lib/python3.7/dist-packages (from kaggle) (4.64.1)
Requirement already satisfied: python-dateutil in /usr/local/lib/python3.7/dist-packages (from kaggle) (2.8.2)
Requirement already satisfied: requests in /usr/local/lib/python3.7/dist-packages (from kaggle) (2.23.0)
Requirement already satisfied: certifi in /usr/local/lib/python3.7/dist-packages (from kaggle) (2022.9.24)
Requirement already satisfied: urllib3 in /usr/local/lib/python3.7/dist-packages (from kaggle) (1.24.3)
Requirement already satisfied: text-unidecode>=1.3 in /usr/local/lib/python3.7/dist-packages (from python-slugify->kaggle) (1.3)
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.7/dist-packages (from requests->kaggle) (2.10)
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.7/dist-packages (from requests->kaggle) (3.0.4)
```

```
!pwd
```

```
/content
```

```
#create the interface between kaggle and content of colab
import os
os.environ['KAGGLE_CONFIG_DIR']="/content"
```

```
#load the kaggle dataset
!kaggle datasets download stackoverflow/stack-overflow-2018-developer-survey
```

```
Warning: Your Kaggle API key is readable by other users on this system! To fix this, you can run 'chmod 600 /content/kaggle.json'
Downloading stack-overflow-2018-developer-survey.zip to /content
 71% 14.0M/19.6M [00:00<00:00, 143MB/s]
100% 19.6M/19.6M [00:00<00:00, 170MB/s]
```

```
!unzip /content/stack-overflow-2018-developer-survey.zip
```

```
Archive:  /content/stack-overflow-2018-developer-survey.zip
  inflating: survey_results_public.csv
  inflating: survey_results_schema.csv
```

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

```
#libraries for ploting
import plotly.offline as pyo
import plotly.graph_objs as go
import plotly.offline as py
from plotly import tools
import pandas as pd
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
from plotly.offline import iplot
import warnings
warnings.filterwarnings("ignore")
import cufflinks as cf
cf.go_offline()
```

```
import plotly.io as pio
pio.renderers.default='colab' #set the rendering process of the graph on colab
```

```
data=pd.read_csv('/content/survey_results_public.csv')
schema_data=pd.read_csv('/content/survey_results_schema.csv')
```

```
/usr/local/lib/python3.7/dist-packages/IPython/core/interactiveshell.py:3326: DtypeWarning:

Columns (8,12,13,14,15,16,50,51,52,53,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,93,94,95,96,97,98,99,100,101,102,103,104,1
```

```
data.shape
```

```
(98855, 129)
```

```
data.tail()
```

	Respondent	Hobby	OpenSource	Country	Student	Employment	FormalEducation	UndergradMajor
98850	101513	Yes	Yes	United States	NaN	NaN	NaN	NaN
98851	101531	No	Yes	Spain	Yes, full-time	Not employed, but looking for work	NaN	NaN
98852	101541	Yes	Yes	India	Yes, full-time	Employed full-time	Bachelor's degree (BA, BS, B.Eng., etc.)	NaN
				Russian		Independent contractor	Some college/university	

```
schema_data.head()
```

	Column	QuestionText
0	Respondent	Randomized respondent ID number (not in order ...
1	Hobby	Do you code as a hobby?
2	OpenSource	Do you contribute to open source projects?
3	Country	In which country do you currently reside?
4	Student	Are you currently enrolled in a formal, degree...

```
data.describe()
```

	Respondent	AssessJob1	AssessJob2	AssessJob3	AssessJob4	AssessJob5	AssessJob6	AssessJob7	AssessJob8	AssessJob9	...	JobEmailPriorities6	Jo
count	98855.000000	66985.000000	66985.000000	66985.000000	66985.000000	66985.000000	66985.000000	66985.000000	66985.000000	66985.000000	...	46213.00000	
mean	50822.971635	6.397089	6.673524	5.906875	4.065791	3.953243	4.407196	5.673181	4.225200	7.640009	...	4.97425	
std	29321.650410	2.788428	2.531202	2.642734	2.541196	2.520499	2.502069	2.923998	2.507411	2.407457	...	1.86063	
min	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	...	1.00000	
25%	25443.500000	4.000000	5.000000	4.000000	2.000000	2.000000	2.000000	3.000000	2.000000	6.000000	...	4.00000	
50%	50823.000000	7.000000	7.000000	6.000000	4.000000	3.000000	4.000000	6.000000	4.000000	8.000000	...	5.00000	
75%	76219.500000	9.000000	9.000000	8.000000	6.000000	6.000000	6.000000	8.000000	6.000000	10.000000	...	7.00000	
max	101592.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	...	7.00000	

8 rows × 42 columns

```
#total number of data=98855
data.count()
```

```
Respondent       98855
Hobby            98855
OpenSource       98855
Country          98443
Student          94901
                 ...
Age              64574
Dependents       62596
MilitaryUS       15781
SurveyTooLong    65941
SurveyEasy       65879
Length: 129, dtype: int64
```

```
#to see the percentage of the missing data=(number of missing data/total data)*100
total_data=data.isnull().sum().sort_values(ascending=False)   #shows the count of the missing data
```

```
percent=(data.isnull().sum()/98855*100).sort_values(ascending=False) #shows the percent of missing data
```

```
percent
```

```
TimeAfterBootcamp    93.270952
MilitaryUS           84.036215
HackathonReasons     74.011431
ErgonomicDevices     65.547519
AdBlockerReasons     61.817814
                       ...
Employment            3.574933
Country               0.416772
Hobby                 0.000000
OpenSource            0.000000
Respondent            0.000000
Length: 129, dtype: float64
```

```
missing_data=pd.concat([total_data,percent],axis=1,keys=['Total missing data','Percent'])
```

```
missing_data
```

	Total missing data	Percent
TimeAfterBootcamp	92203	93.270952
MilitaryUS	83074	84.036215
HackathonReasons	73164	74.011431
ErgonomicDevices	64797	65.547519
AdBlockerReasons	61110	61.817814
...	...	...
Country	412	0.416772
OpenSource	0	0.000000

Data Visualization (Data Exploration)

```
temp=data['Hobby'].value_counts()
```

```
temp
```

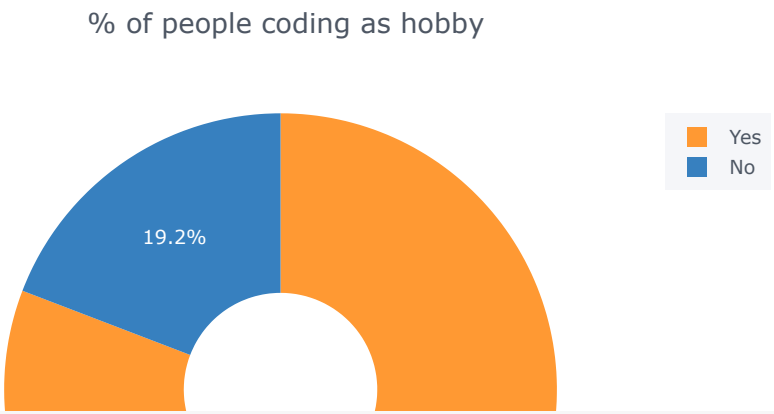
```
Yes    79897
No     18958
Name: Hobby, dtype: int64
```

```
data_plot=pd.DataFrame({'labels':temp.index,'counts':temp.values})
```

```
data_plot
```

	labels	counts
0	Yes	79897
1	No	18958

```
data_plot.iplot(kind='pie',labels='labels',values='counts',title="% of people coding as hobby",hole=0.35)
```

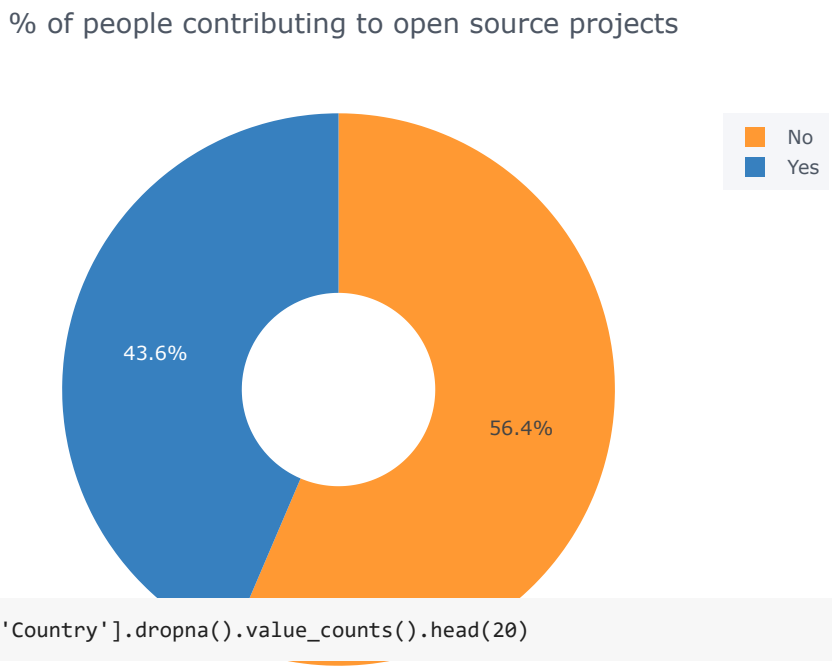



```
schema_data
```

	Column	QuestionText
0	Respondent	Randomized respondent ID number (not in order ...
1	Hobby	Do you code as a hobby?
2	OpenSource	Do you contribute to open source projects?
3	Country	In which country do you currently reside?
4	Student	Are you currently enrolled in a formal, degree...
...	...	...
124	Age	What is your age? If you prefer not to answer,...
125	Dependents	Do you have any children or other dependents t...
126	MilitaryUS	Are you currently serving or have you ever ser...
127	SurveyTooLong	How do you feel about the length of the survey...
128	SurveyEasy	How easy or difficult was this survey to compl...

129 rows × 2 columns

```
data_OS=data['OpenSource'].value_counts()
data_plot=pd.DataFrame({'labels':data_OS.index,'counts':data_OS.values})
data_plot.iplot(kind='pie',labels='labels',values='counts',title="% of people contributing to open source projects",hole=0.35)
```

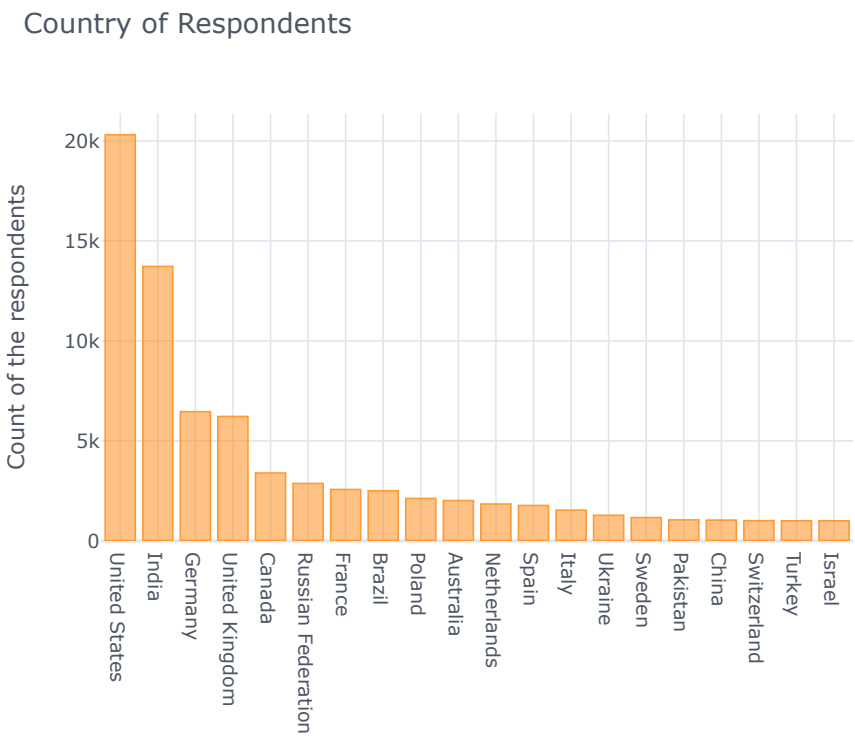



```
data_country=data['Country'].dropna().value_counts().head(20)
```

```
data_country
```

```
United States         20309
India                 13721
Germany                6459
United Kingdom         6221
Canada                 3393
Russian Federation     2869
France                 2572
Brazil                 2505
Poland                 2122
Australia              2018
Netherlands            1841
Spain                  1769
Italy                  1535
Ukraine                1279
Sweden                 1164
Pakistan               1050
China                  1037
Switzerland            1010
Turkey                 1004
Israel                 1003
Name: Country, dtype: int64
```

```
data_country.iplot(kind='bar',xTitle='Country Name',yTitle='Count of the respondents',title='Country of Respondents')
```



✓ 1s completed at 14:14