```
1 import pandas as pd
2 import numpy as np
3 from google.colab import drive
4 drive.mount("/content/drive")
```

```
Mounted at /content/drive
```

```
1 df = pd.read_csv("/content/drive/MyDrive/Time series forecasting/Python+CC (1)
2 # header = 0 means our first row contain header
3 df.head(10)
```

	Customer ID	Customer Name	Segment	Age	Country	City	State	Postal Code	Regi
0	CG-12520	Claire Gute	Consumer	67	United States	Henderson	Kentucky	42420	So
1	DV-13045	Darrin Van Huff	Corporate	31	United States	Los Angeles	California	90036	W
2	SO-20335	Sean O'Donnell	Consumer	65	United States	Fort Lauderdale	Florida	33311	So
3	BH-11710	Brosina Hoffman	Consumer	20	United States	Los Angeles	California	90032	W
4	AA-10480	Andrew Allen	Consumer	50	United States	Concord	North Carolina	28027	So
5	IM-15070	Irene Maddox	Consumer	66	United States	Seattle	Washington	98103	W

```
1 df1 = pd.read_csv("/content/drive/MyDrive/Time series forecasting/Python+CC (1
2                    header = 0, index_col = 0) ##index_col =0 means Customer ID
3 # header = 0 means our first row contain header
4 df1.head()
```

Customer ID	Customer Name	Segment	Age	Country	City	State	Postal Code	Region
CG-12520	Claire Gute	Consumer	67	United States	Henderson	Kentucky	42420	South
DV-13045	Darrin Van Huff	Corporate	31	United States	Los Angeles	California	90036	West
SO-20335	Sean O'Donnell	Consumer	65	United States	Fort Lauderdale	Florida	33311	South

```
## if we want to see 1st row of our dataset
df.iloc[0]
```

```
Customer ID           CG-12520
Customer Name      Claire Gute
Segment               Consumer
Age                         67
Country          United States
City                 Henderson
State                 Kentucky
Postal Code              42420
Region                   South
Name: 0, dtype: object
```

```
df.loc[0]
```

```
Customer ID           CG-12520
Customer Name      Claire Gute
Segment               Consumer
Age                         67
Country          United States
City                 Henderson
State                 Kentucky
Postal Code              42420
Region                   South
Name: 0, dtype: object
```

```
df1.iloc[0:2]
```

Customer ID	Customer Name	Segment	Age	Country	City	State	Postal Code	Region
CG-12520	Claire Gute	Consumer	67	United States	Henderson	Kentucky	42420	South

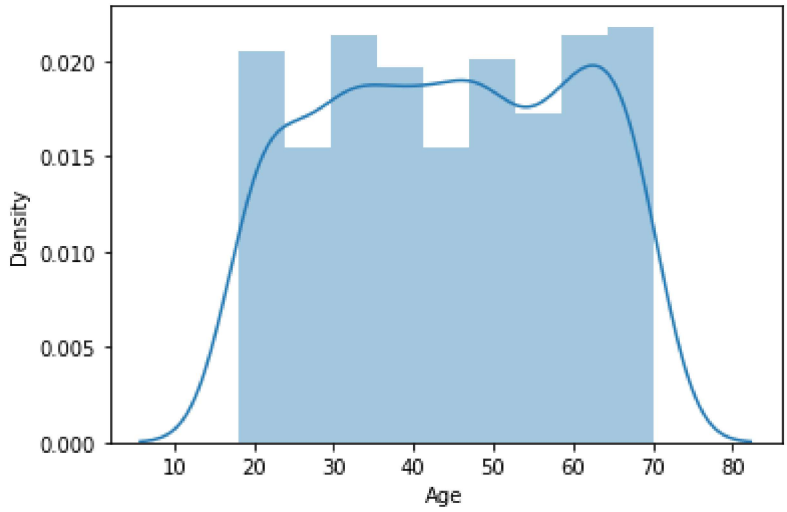
```
1
```

Seaborn

```
import seaborn as sns
```

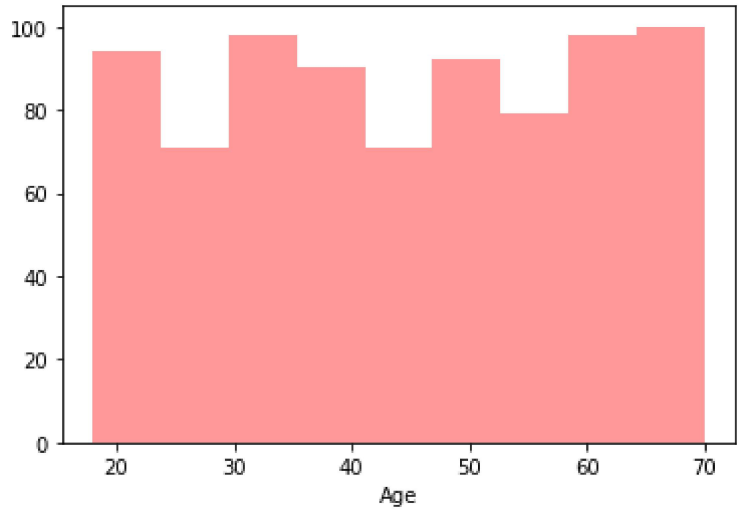
```
sns.distplot(df.Age)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarnin
  warnings.warn(msg, FutureWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7f3d93a59950>
```




```
1 sns.distplot(df.Age, kde = False, color = 'r')
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarnin
  warnings.warn(msg, FutureWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7f3d910fc8d0>
```




```
1 help(sns.distplot)
```

```
    .. plot::
        :context: close-figs

        >>> import seaborn as sns, numpy as np
        >>> sns.set_theme(); np.random.seed(0)
        >>> x = np.random.randn(100)
        >>> ax = sns.distplot(x)

    Use Pandas objects to get an informative axis label:

    .. plot::
```

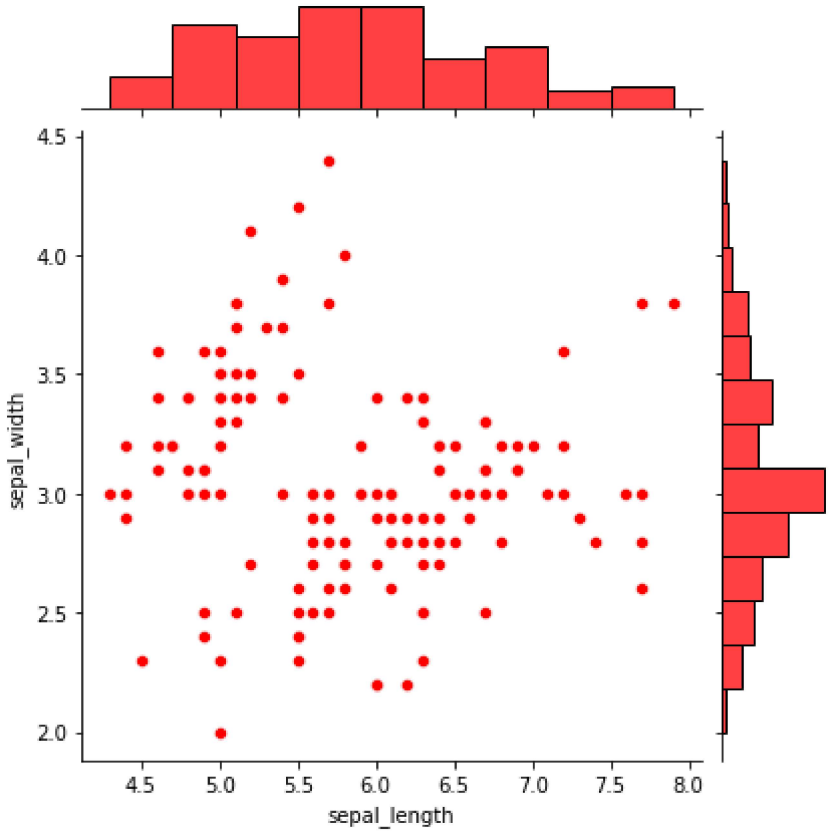
```
        :context: close-figs

        >>> import pandas as pd
        >>> x = pd.Series(x, name="x variable")
        >>> ax = sns.distplot(x)

    Plot the distribution with a kernel density estimate and rug plot:

    .. plot::
        :context: close-figs

        >>> ax = sns.distplot(x, rug=True, hist=False)

    Plot the distribution with a histogram and maximum likelihood gaussian
    distribution fit:

    .. plot::
        :context: close-figs

        >>> from scipy.stats import norm
        >>> ax = sns.distplot(x, fit=norm, kde=False)

    Plot the distribution on the vertical axis:

    .. plot::
        :context: close-figs

        >>> ax = sns.distplot(x, vertical=True)

    Change the color of all the plot elements:

    .. plot::
        :context: close-figs

        >>> sns.set_color_codes()
        >>> ax = sns.distplot(x, color="y")

    Pass specific parameters to the underlying plot functions:

    .. plot::
        :context: close-figs

        >>> ax = sns.distplot(x, rug=True, rug_kws={"color": "g"},
        ...                   kde_kws={"color": "k", "lw": 3, "label": "KDE"},
        ...                   hist_kws={"histtype": "step", "linewidth": 3,
        ...                             "alpha": 1, "color": "g"})
```

```
1 iris = sns.load_dataset("iris")
2 iris.head()
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa

```
1 sns.jointplot("sepal_length","sepal_width", data = iris, color = 'r')
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: P
  FutureWarning
<seaborn.axisgrid.JointGrid at 0x7f3d910948d0>
```

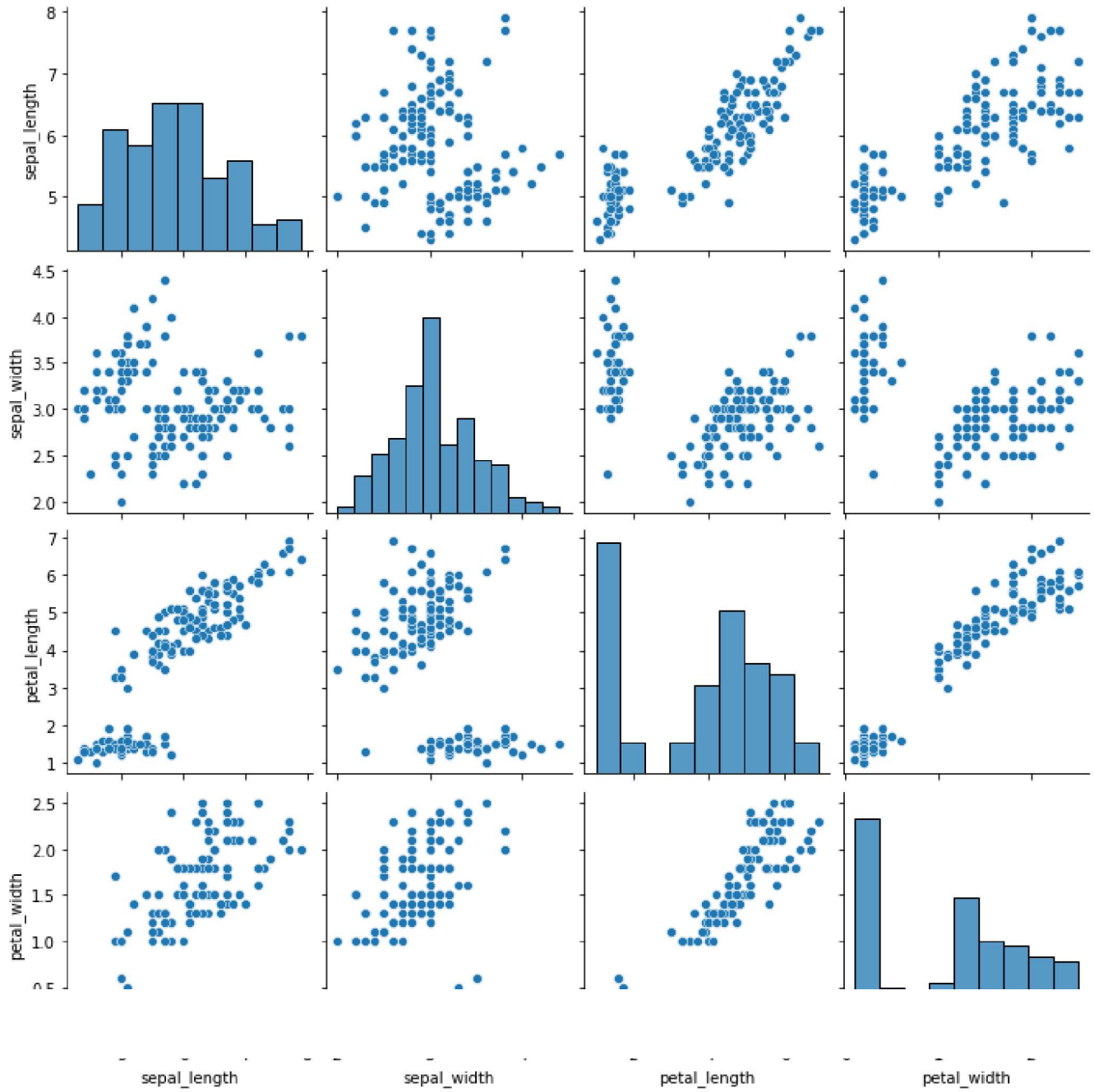



```
1
```

Scatter plot for all the variables

```
1 sns.pairplot(data = iris)
```

```
<seaborn.axisgrid.PairGrid at 0x7f3d8e290910>
```



1

✓ 5s completed at 12:37 AM