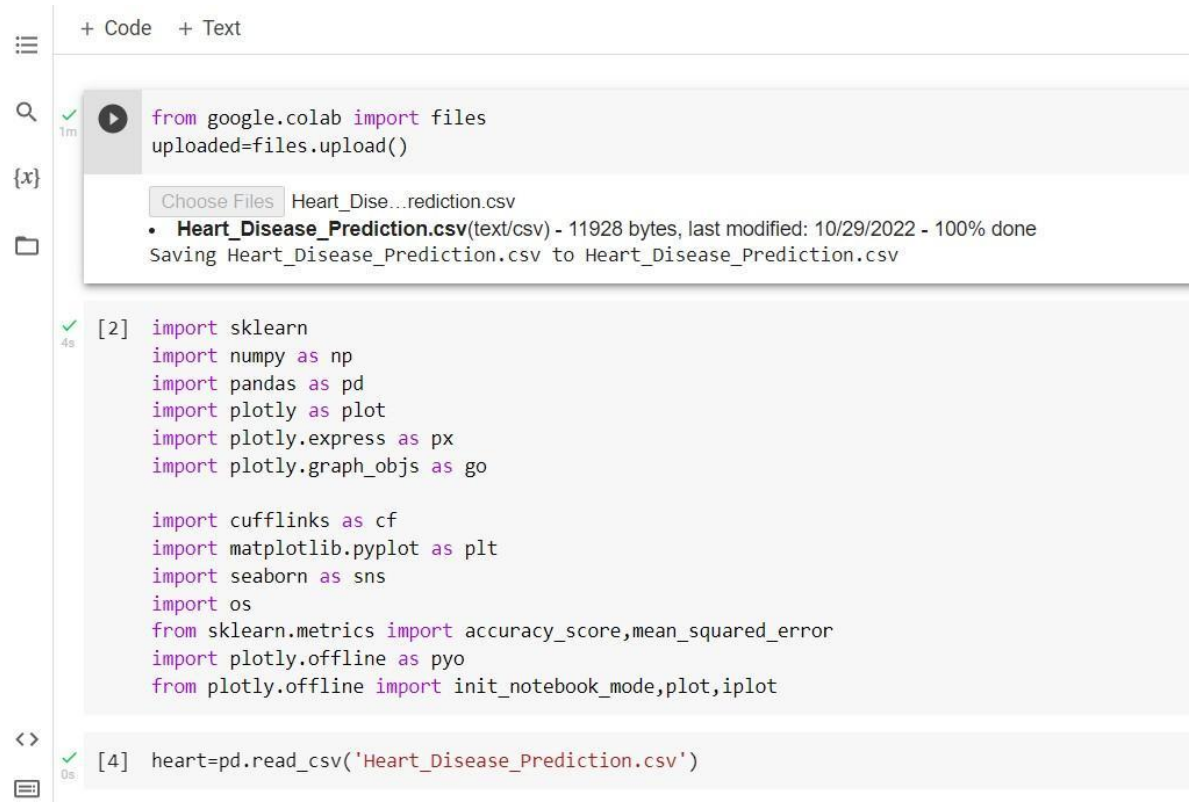
SPRINT-2

Date	06 November 2022
Team ID	PNT2022TMID21605
Project Name	Visualizing and Predicting Heart Diseases with an Interactive Dash Board

Visualizing the data to get more understanding



+ Code + Text

```
from google.colab import files
uploaded=files.upload()
```

Choose Files Heart_Dise...rediction.csv

- **Heart_Disease_Prediction.csv**(text/csv) - 11928 bytes, last modified: 10/29/2022 - 100% done

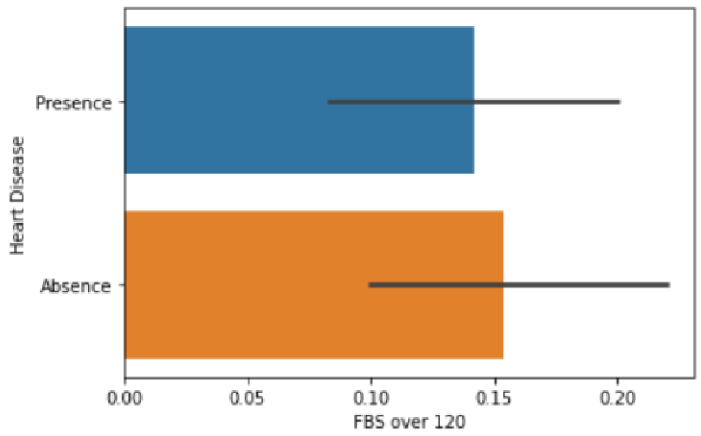
```
Saving Heart_Disease_Prediction.csv to Heart_Disease_Prediction.csv
```

```
[2] import sklearn
    import numpy as np
    import pandas as pd
    import plotly as plot
    import plotly.express as px
    import plotly.graph_objs as go

    import cufflinks as cf
    import matplotlib.pyplot as plt
    import seaborn as sns
    import os
    from sklearn.metrics import accuracy_score,mean_squared_error
    import plotly.offline as pyo
    from plotly.offline import init_notebook_mode,plot,iplot
```

```
[4] heart=pd.read_csv('Heart_Disease_Prediction.csv')
```

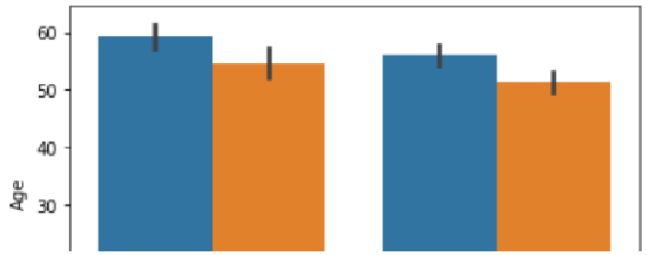
+ Code + Text

```
sns.barplot(x="FBS over 120", y="Heart Disease", data=heart)
plt.show()
```



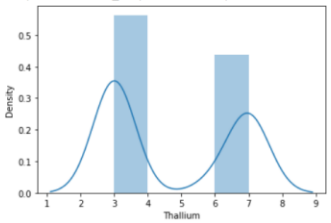

```
[9] sns.barplot(x=heart['Sex'],y=heart['Age'],hue=heart['Heart Disease'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fc23f85c410>
```



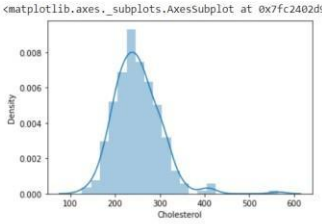

```
sns.distplot(heart["Thallium"])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning:

`distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibilit

<matplotlib.axes._subplots.AxesSubplot at 0x7fc23f923350>
```



```
sns.distplot(heart["Cholesterol"])
```

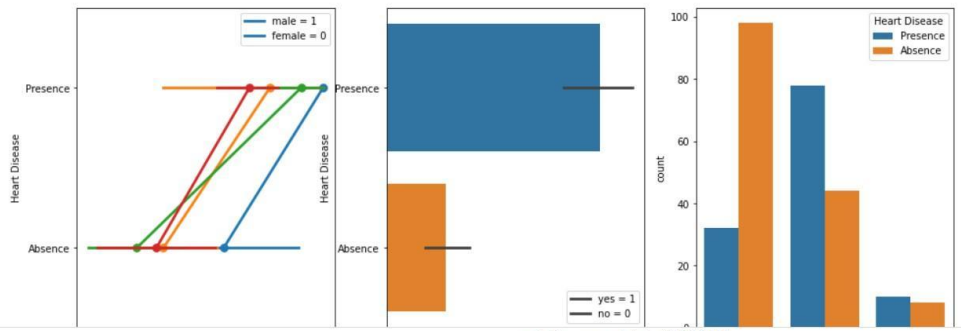
```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning:

`distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histp

<matplotlib.axes._subplots.AxesSubplot at 0x7fc2402d9290>
```




```
#  correlation heatmap
sns.heatmap(heart[numeric_columns].corr(),annot=True, cmap='terrain', linewidths=0.1)
fig=plt.gcf()
fig.set_size_inches(8,6)
plt.show()
```

+ Code + Text

```
# create pairplot and two barplots
plt.figure(figsize=(16,6))
plt.subplot(131)
sns.pointplot(x="Sex", y="Heart Disease", hue='Chest pain type', data=heart)
plt.legend(['male = 1', 'female = 0'])
plt.subplot(132)
sns.barplot(x="Exercise angina", y="Heart Disease", data=heart)
plt.legend(['yes = 1', 'no = 0'])
plt.subplot(133)
sns.countplot(x="Slope of ST", hue='Heart Disease', data=heart)
plt.show()
```



0s completed at 10:34 AM

[4] `heart=pd.read_csv('Heart_Disease_Prediction.csv')`

```
plt.bar(x=heart['Sex'],height=heart['Age'])
plt.show()
```