```
In [ ]: import pandas as pd
        data = pd.read_csv(r"C:\Users\SURYA MANOJ\OneDrive\Desktop\dataset\Student Depression Dataset
        data
```

DATA PREPROCESSING

```
In [ ]: data.isnull()
        data['Financial Stress'] = data['Financial Stress'].fillna(data['Financial Stress'].mean())
```

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

```
In [ ]: data = data.drop(['id','Profession','Work Pressure','Job Satisfaction','City'],axis =1)
        data
```

```
In [ ]: from sklearn.preprocessing import LabelEncoder
        le = LabelEncoder()
        data['Gender'] = le.fit_transform(data['Gender'])
        data['Sleep Duration'] = le.fit_transform(data['Sleep Duration'])
        data['Dietary Habits'] = le.fit_transform(data['Dietary Habits'])
        data['Degree']= le.fit_transform(data['Degree'])
        data['Have you ever had suicidal thoughts ?'] = le.fit_transform(data['Have you ever had suic
        data['Family History of Mental Illness'] = le.fit_transform(data['Family History of Mental Il
        data
```

```
In [ ]: x = data.drop(['Depression'],axis = 1)
        x
```

```
In [ ]: y = data['Depression']
        y
```

```
In [ ]: 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 [ ]: print(x_train)
```

```
In [ ]: print(x_test)
```

```
In [ ]: print(y_train)
```

```
In [ ]: print(y_test)
```

```
In [ ]:
```

```
In [ ]: from sklearn.neural_network import MLPClassifier
        clf = MLPClassifier()
        clf_model = clf.fit(x_train,y_train)
        clf_model
```

```
In [ ]: mlp_p = clf_model.predict(x_test)
        mlp_p
```

```
In [ ]: from sklearn.metrics import accuracy_score
        acs_mlp = accuracy_score(y_test,mlp_p)
        acs_mlp
```

In [ ]:

```python
# Taking input from the user with corrected syntax and formatting
Gender = int(input("Enter the Gender \n 'Female': 0 \n 'Male': 1\n"))
Age = float(input("Enter the Age: "))
Academic_Pressure = float(input("Enter the Academic Pressure: "))
CGPA = float(input("Enter the CGPA: "))
Study_Satisfaction = float(input("Enter the Study Satisfaction: "))
Sleep_Duration = int(input("\nEnter the Sleep Duration\n '5-6 hours': 0 \n '7-8 hours': 1 \n
Dietary_Habits = int(input("\nEnter the Dietary Habits \n 'Healthy': 0 \n 'Moderate': 1 \n 'U
Degree = int(input("Enter Your Degree \n 'B.Pharm': 0 \n 'BA': 1 \n 'BCA': 2 \n 'BSc': 3 \n '
Suicidal_Thoughts = int(input("Enter if you have ever had suicidal thoughts \n 'No': 0 \n 'Ye
Work_Study_Hours = float(input("Enter Work/Study Hours: "))
Financial_Stress = float(input("Enter the Financial Stress: "))
Family_History_Mental_Illness = int(input("Enter Family History of Mental Illness \n 'No': 0

# Displaying the collected data
print("\nCollected Data:")
print(f"Gender: {Gender}")
print(f"Age: {Age}")
print(f"Academic Pressure: {Academic_Pressure}")
print(f"CGPA: {CGPA}")
print(f"Study Satisfaction: {Study_Satisfaction}")
print(f"Sleep Duration: {Sleep_Duration}")
print(f"Dietary Habits: {Dietary_Habits}")
print(f"Degree: {Degree}")
print(f"Suicidal Thoughts: {Suicidal_Thoughts}")
print(f"Work/Study Hours: {Work_Study_Hours}")
print(f"Financial Stress: {Financial_Stress}")
print(f"Family History of Mental Illness: {Family_History_Mental_Illness}")

'''
  539   Female  24      Kolkata Student 5       0       8.96    1       0       5-6 hours
'''
```

In [ ]:

```python
import numpy as np
input_data = np.array([[
    Gender,
    Age,
    Academic_Pressure,
    CGPA,
    Study_Satisfaction,
    Sleep_Duration,
    Dietary_Habits,
    Degree,
    Suicidal_Thoughts,
    Work_Study_Hours,
    Financial_Stress,
    Family_History_Mental_Illness
]])

# Make a prediction
prediction = clf_model.predict(input_data)
prediction_prob = clf_model.predict_proba(input_data)

# Display the results
print("\nPrediction Result:")
if prediction[0] == 1:
    print("The model predicts: The Students Mental Health is not good")
else:
    print("The model predicts: The Students Mental Health is  good")

print("\nPrediction Probability:")
print(f"Probability of Having Good Mental Health : {prediction_prob[0][0]:.2f}")
print(f"Probability of Not Having the Good Mental Health: {prediction_prob[0][1]:.2f}")
```

In [ ]:

```
```