IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING

BANU PRIYA
MARIAANANTHI
RAMAJEYANTHI
SURYA
MANISANKAR

IDENTIFYING PATTERNS AND TRENDS IN CAMPUS PLACEMENT DATA USING MACHINE LEARNING

1. INTRODUCTION

❖ OVERVIEW:

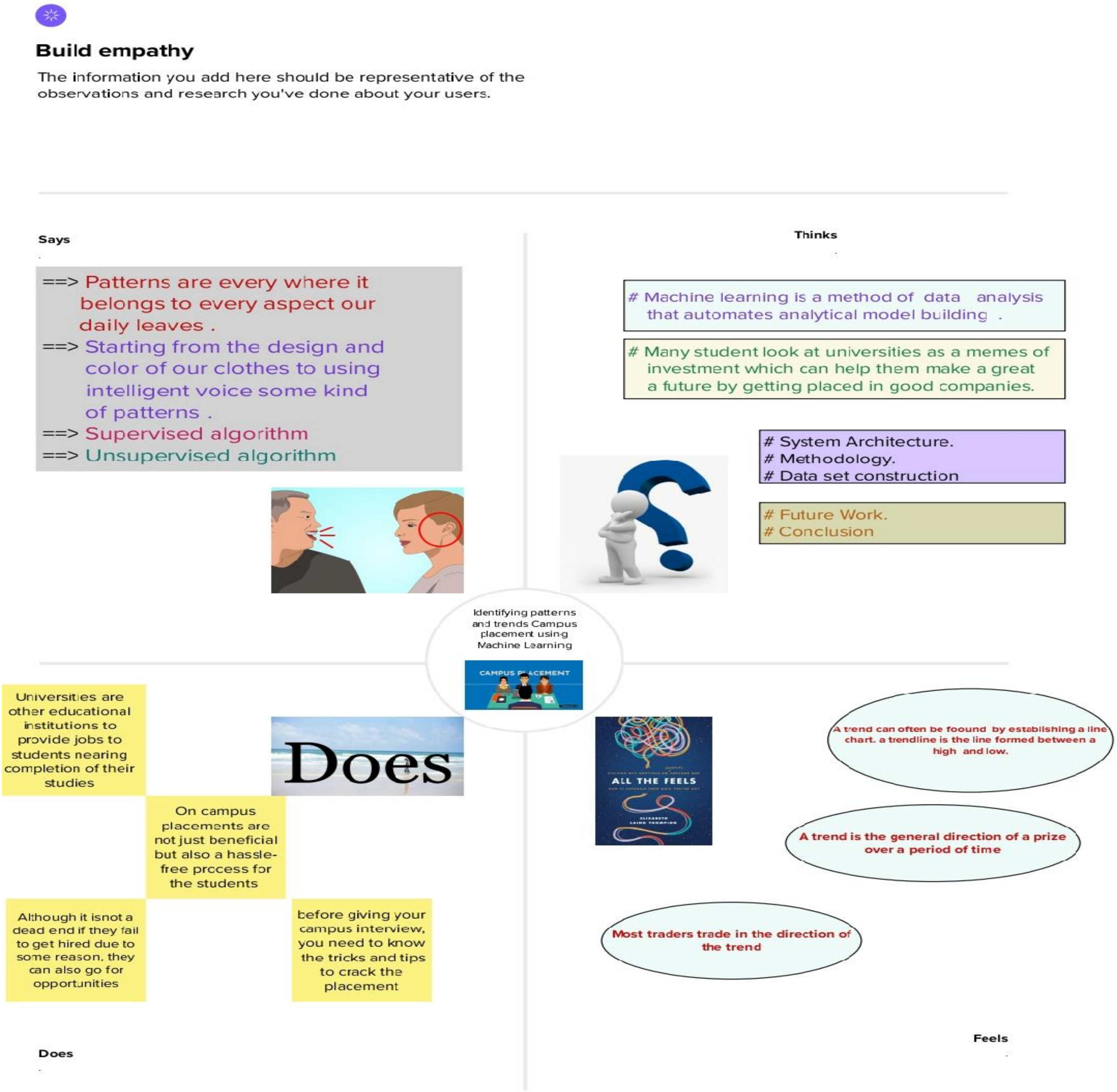
Analyzing campus placement data using machine learning techniques involves collecting data on campus placements, preprocessing the data, exploring the data to identify patterns and trends, selecting the most relevant features, choosing appropriate machine learning models, training the models, evaluating the performance of the models, and deploying the best performing model to make predictions about future placement outcomes. This analysis can provide valuable insights into the factors that influence job placements, leading to more successful placements and a better understanding of the recruitment process.

❖ PURPOSE:

- **Data Collection: Gather data on campus placements from various sources such as placement cells, colleges, job portals. Company, job profile, salary offered, location, education qualification, etc.**
- **Data Preparation: Clean the data and remove any errors, duplicates, or missing values.**
- **Exploratory Data Analysis: Use statistical methods to analyze the data and visualize the trends in the placement data.**
- **Machine Learning Model Development: Build a machine learning model to predict the placement trends to predict the placement trends and patterns.**
- **Model Evaluation: Evaluate the accuracy of the model and compare it with other models to select the best one.**

2.PROBLEM DEFINITION & DESIGN THINKING

❖ EMPATHY MAP :



Build empathy

The information you add here should be representative of the observations and research you've done about your users.

Says

==> Patterns are every where it belongs to every aspect our daily leaves .

==> Starting from the design and color of our clothes to using intelligent voice some kind of patterns .

==> Supervised algorithm

==> Unsupervised algorithm

Thinks

Machine learning is a method of data analysis that automates analytical model building .

Many student look at universities as a memes of investment which can help them make a great a future by getting placed in good companies.

System Architecture.
Methodology.
Data set construction

Future Work.
Conclusion

Identifying patterns and trends Campus placement using Machine Learning

Universities are other educational institutions to provide jobs to students nearing completion of their studies

On campus placements are not just beneficial but also a hassle-free process for the students

Although it isnot a dead end if they fail to get hired due to some reason, they can also go for opportunities

before giving your campus interview, you need to know the tricks and tips to crack the placement

A trend can often be foound by establishing a line chart. a trendline is the line formed between a high and low.

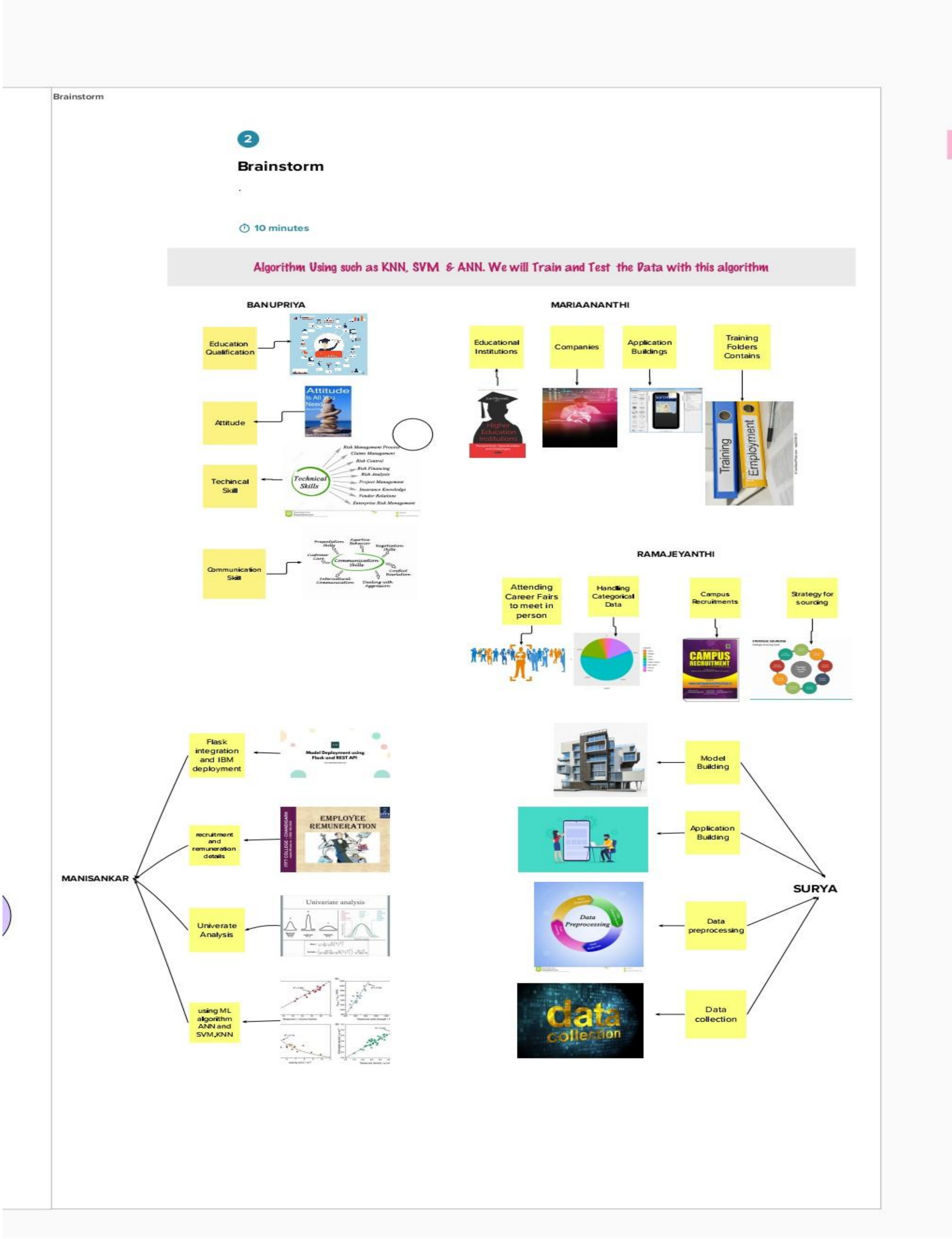
A trend is the general direction of a prize over a period of time

Most traders trade in the direction of the trend

Does

Feels

❖ BRAIN STORM :



Brainstorm

2

Brainstorm

10 minutes

Algorithm Using such as KNN, SVM & ANN. We will Train and Test the Data with this algorithm

BANUPRIYA

Education Qualification

Attitude

Techincal Skill

Communication Skill

MARIAANANTHI

Educational Institutions

Companies

Application Buildings

Training Folders Contains

RAMAJEYANTHI

Attending Career Fairs to meet in person

Handling Categorical Data

Campus Recruitments

Strategy for sourcing

MANISANKAR

Flask integration and IBM deployment

recruitment and remuneration details

Univerate Analysis

using ML algorithm ANN and SVM,KNN

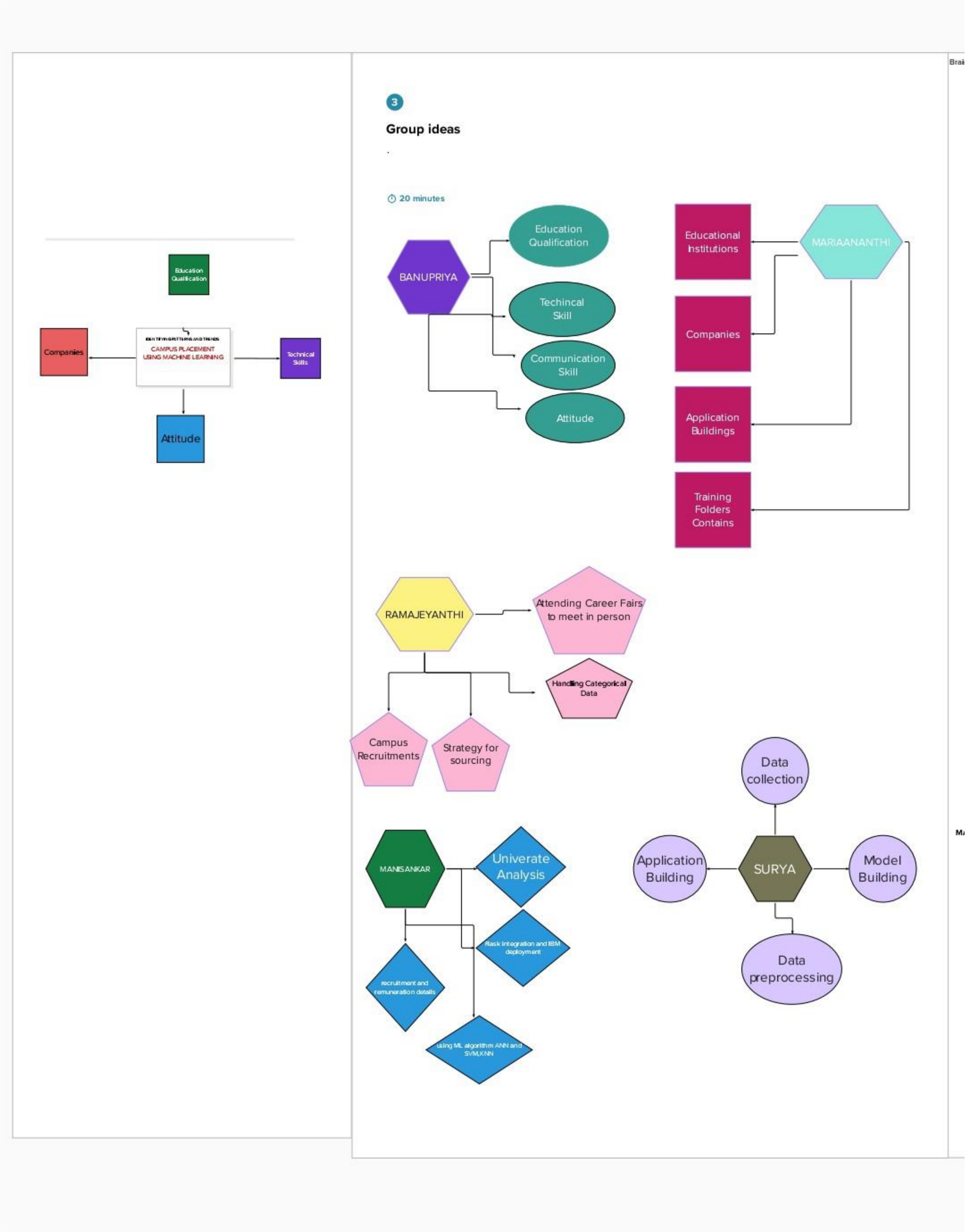
SURYA

Model Building

Application Building

Data preprocessing

Data collection



Brai

IDENTIFYING PATTERNS AND TRENDS

CAMPUS PLACEMENT USING MACHINE LEARNING

Education Qualification

Companies

Technical Skills

Attitude

3

Group ideas

.

20 minutes

BANUPRIYA

Education Qualification

Techincal Skill

Communication Skill

Attitude

MARIAANANTHI

Educational Institutions

Companies

Application Buildings

Training Folders Contains

RAMAJEYANTHI

Attending Career Fairs to meet in person

Handling Categorical Data

Campus Recruitments

Strategy for sourcing

MANISANKAR

Univerate Analysis

Flask integration and IBM deployment

recruitment and remuneration details

using ML algorithm ANN and SVM,KNN

SURYA

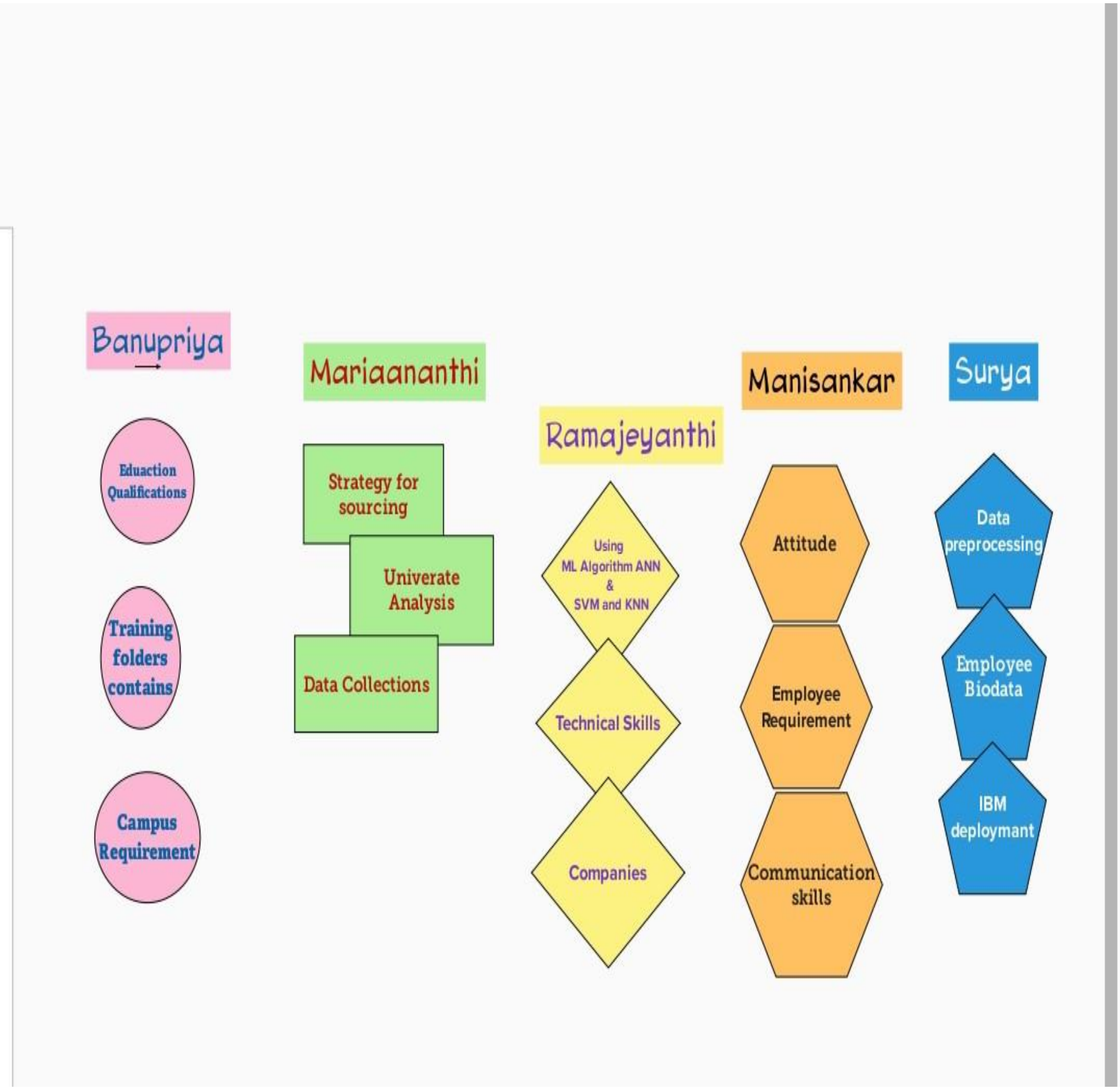
Data collection

Application Building

Model Building

Data preprocessing

M



Banupriya

Eduaction Qualifications

Training folders contains

Campus Requirement

Mariaananthi

Strategy for sourcing

Univerate Analysis

Data Collections

Ramajeyanthi

Using ML Algorithm ANN & SVM and KNN

Technical Skills

Companies

Manisankar

Attitude

Employee Requirement

Communication skills

Surya

Data preprocessing

Employee Biodata

IBM deploymant

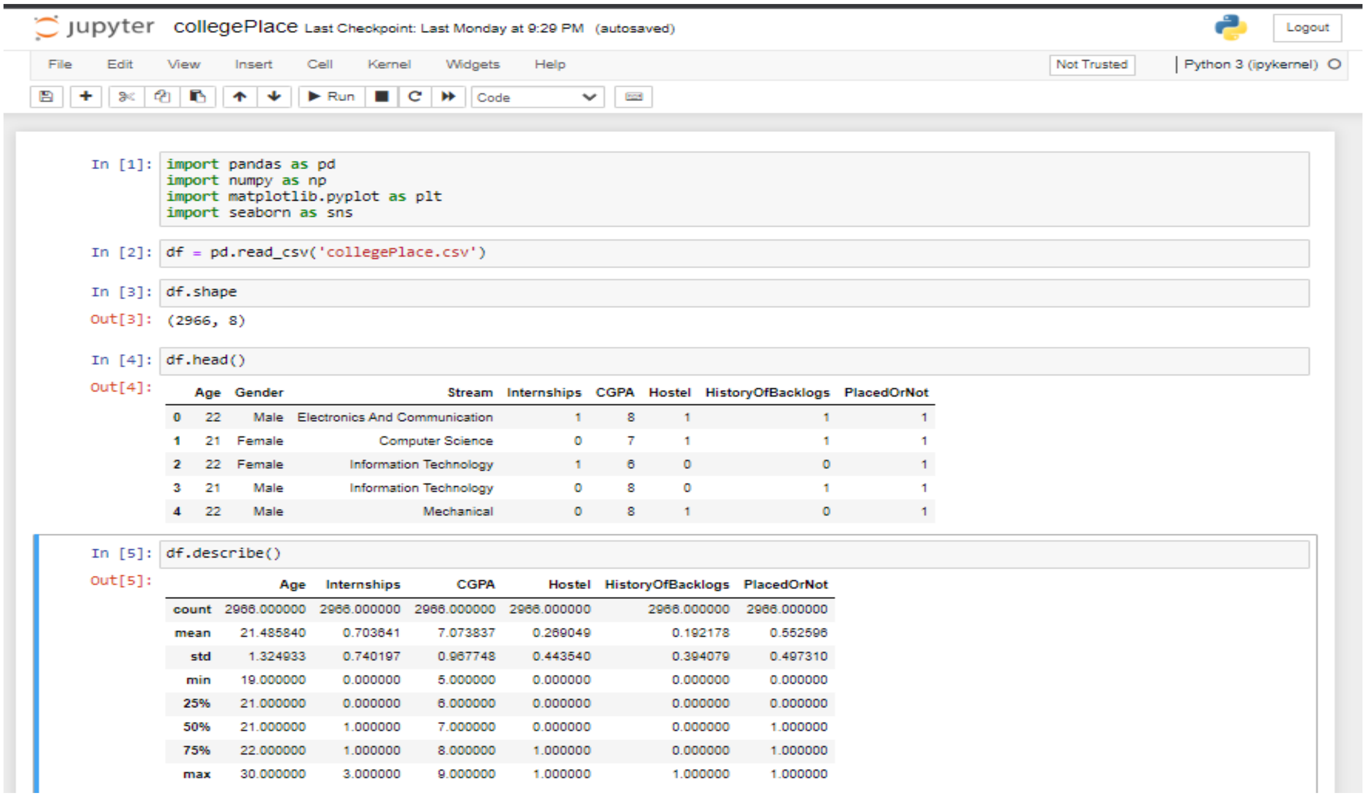
3.RESULT

Data Model :

<table>
<tr><th>Object name</th><th>Fields in the Object</th></tr>
<tr><td>Prediction</td><td><table><tr><th>Field label</th><th>Data type</th></tr><tr><td>Age</td><td>integer</td></tr><tr><td>Gender</td><td>integer</td></tr></table></td></tr>
<tr><td>SVM</td><td><table><tr><th>Field label</th><th>Data type</th></tr><tr><td>stream</td><td>float</td></tr><tr><td></td><td></td></tr></table></td></tr>
</table>

Activity & Screenshot :

Read the Dataset


jupyter collegePlace Last Checkpoint: Last Monday at 9:29 PM (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel) ○
Run
Code

```
In [1]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        import seaborn as sns

In [2]: df = pd.read_csv('collegePlace.csv')

In [3]: df.shape
Out[3]: (2966, 8)

In [4]: df.head()
Out[4]:
```

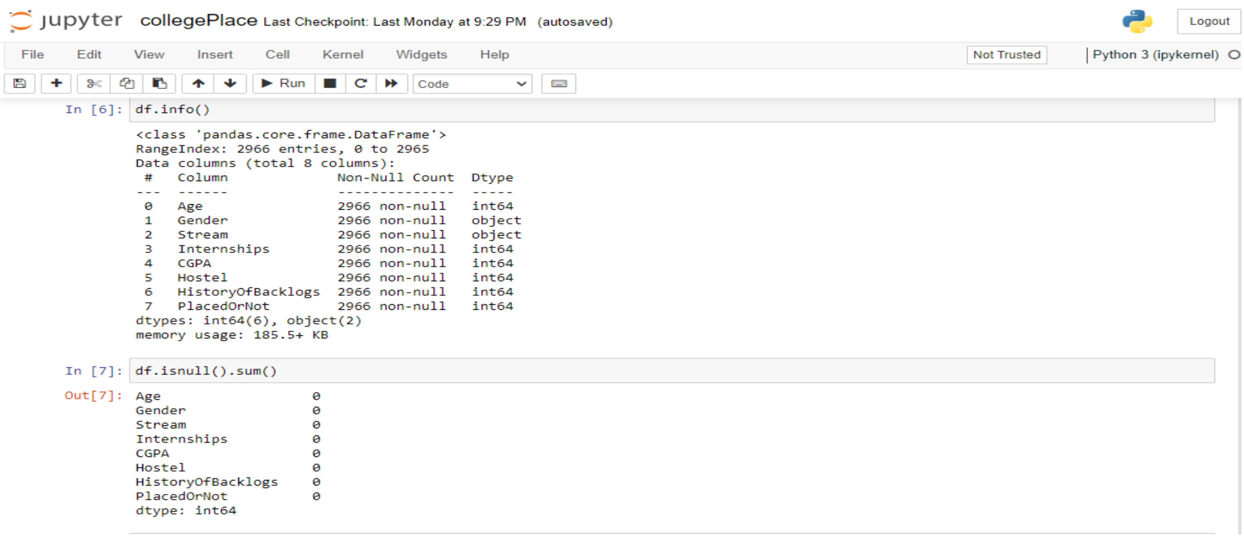
	Age	Gender	Stream	Internships	CGPA	Hostel	HistoryOfBacklogs	PlacedOrNot
0	22	Male	Electronics And Communication	1	8	1	1	1
1	21	Female	Computer Science	0	7	1	1	1
2	22	Female	Information Technology	1	6	0	0	1
3	21	Male	Information Technology	0	8	0	1	1
4	22	Male	Mechanical	0	8	1	0	1

```
In [5]: df.describe()
Out[5]:
```

	Age	Internships	CGPA	Hostel	HistoryOfBacklogs	PlacedOrNot
count	2966.000000	2966.000000	2966.000000	2966.000000	2966.000000	2966.000000
mean	21.485840	0.703641	7.073837	0.269049	0.192178	0.552596
std	1.324933	0.740197	0.967748	0.443540	0.394079	0.497310
min	19.000000	0.000000	5.000000	0.000000	0.000000	0.000000
25%	21.000000	0.000000	6.000000	0.000000	0.000000	0.000000
50%	21.000000	1.000000	7.000000	0.000000	0.000000	1.000000
75%	22.000000	1.000000	8.000000	1.000000	0.000000	1.000000
max	30.000000	3.000000	9.000000	1.000000	1.000000	1.000000

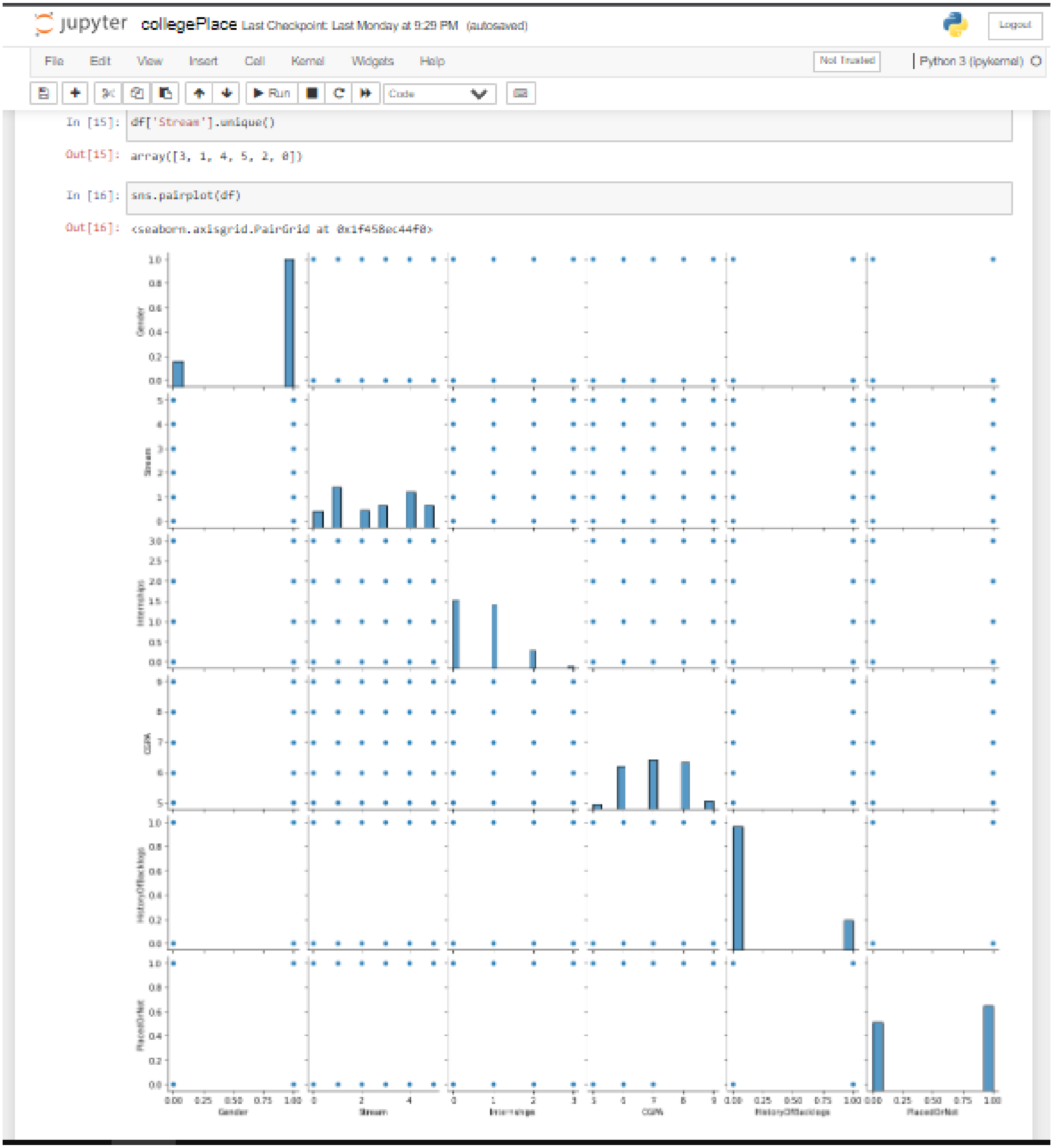
Data Preparation

Handling missing values


jupyter collegePlace Last Checkpoint: Last Monday at 9:29 PM (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel) ○
Run
Code

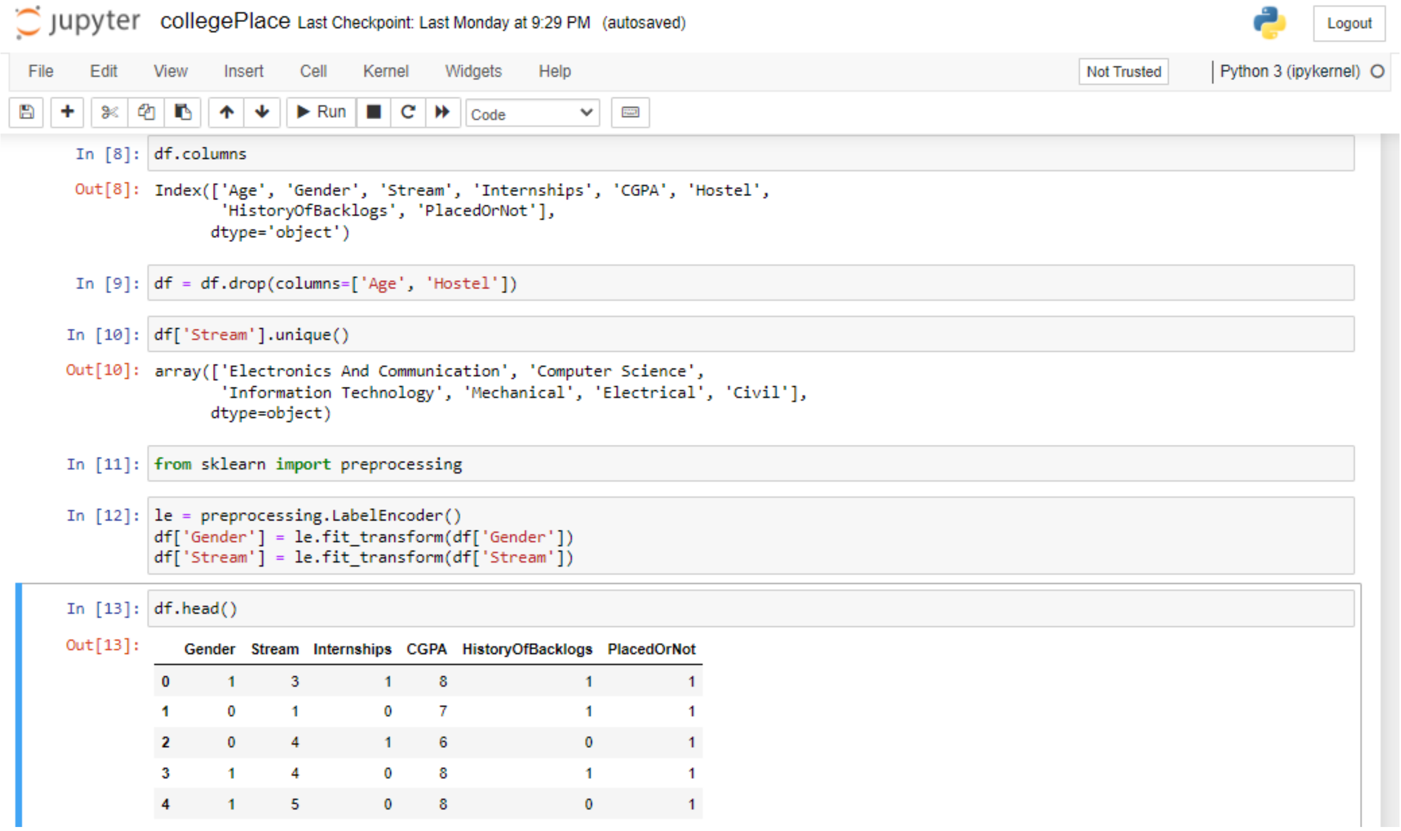
```
In [6]: df.info()

        <class 'pandas.core.frame.DataFrame'>
        RangeIndex: 2966 entries, 0 to 2965
        Data columns (total 8 columns):
         #   Column             Non-Null Count  Dtype
        ---  ------             --------------  -----
         0   Age                2966 non-null   int64
         1   Gender             2966 non-null   object
         2   Stream             2966 non-null   object
         3   Internships        2966 non-null   int64
         4   CGPA               2966 non-null   int64
         5   Hostel             2966 non-null   int64
         6   HistoryOfBacklogs  2966 non-null   int64
         7   PlacedOrNot        2966 non-null   int64
        dtypes: int64(6), object(2)
        memory usage: 185.5+ KB

In [7]: df.isnull().sum()

Out[7]: Age                  0
        Gender               0
        Stream               0
        Internships          0
        CGPA                 0
        Hostel               0
        HistoryOfBacklogs    0
        PlacedOrNot          0
        dtype: int64
```

Handling outliers



```
In [15]: df['Stream'].unique()

Out[15]: array([3, 1, 4, 5, 2, 0])

In [16]: sns.pairplot(df)

Out[16]: <seaborn.axisgrid.PairGrid at 0x1f458ec44f0>
```

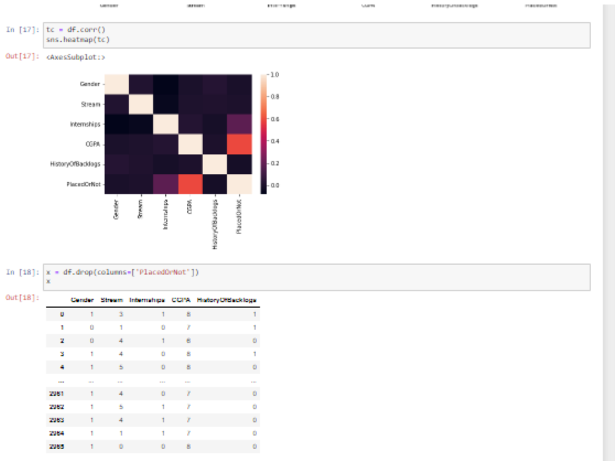
Handling Categorical Values



```
In [8]: df.columns
Out[8]: Index(['Age', 'Gender', 'Stream', 'Internships', 'CGPA', 'Hostel',
               'HistoryOfBacklogs', 'PlacedOrNot'],
              dtype='object')

In [9]: df = df.drop(columns=['Age', 'Hostel'])

In [10]: df['Stream'].unique()
Out[10]: array(['Electronics And Communication', 'Computer Science',
                'Information Technology', 'Mechanical', 'Electrical', 'Civil'],
               dtype=object)

In [11]: from sklearn import preprocessing

In [12]: le = preprocessing.LabelEncoder()
         df['Gender'] = le.fit_transform(df['Gender'])
         df['Stream'] = le.fit_transform(df['Stream'])

In [13]: df.head()
Out[13]:
```

	Gender	Stream	Internships	CGPA	HistoryOfBacklogs	PlacedOrNot
0	1	3	1	8	1	1
1	0	1	0	7	1	1
2	0	4	1	6	0	1
3	1	4	0	8	1	1
4	1	5	0	8	0	1

Multivariate analysis



```
In [17]: tc = df.corr()
         sns.heatmap(tc)
Out[17]: <AxesSubplot:>
```

```
In [18]: x = df.drop(columns=['PlacedOrNot'])
         x
Out[18]:
```

	Gender	Stream	Internships	CGPA	HistoryOfBacklogs
0	1	3	1	8	1
1	0	1	0	7	1
2	0	4	1	6	0
3	1	4	0	8	1
4	1	5	0	8	0
...	...	...	...	...	...
2961	1	4	0	7	0
2962	1	5	1	7	0
2963	1	4	1	7	0
2964	1	1	1	7	0
2965	1	0	0	8	0

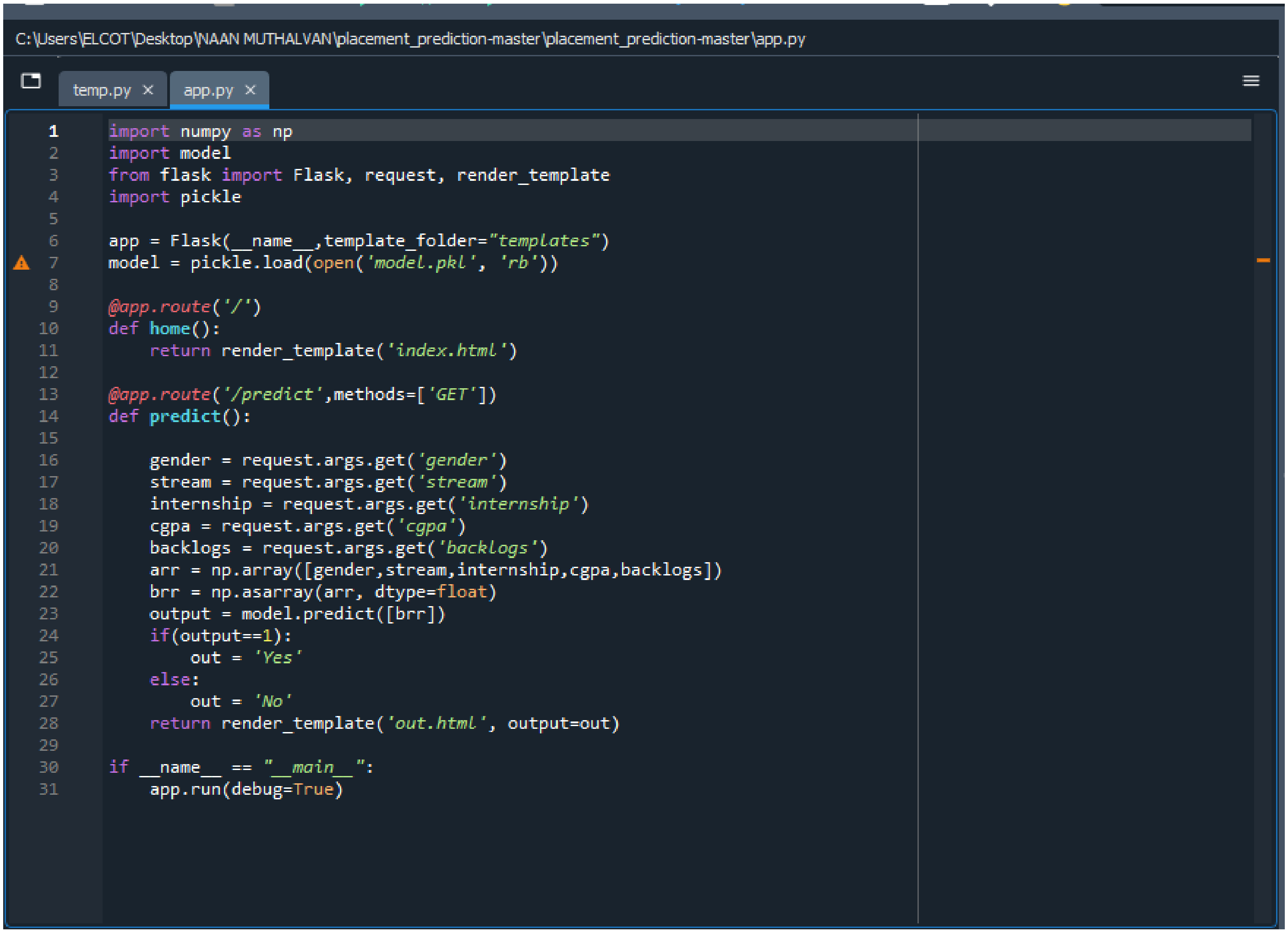
Model Deployment

Integrate with Web Framework

- Building HTML Pages
- Building server side script
- Run the web application

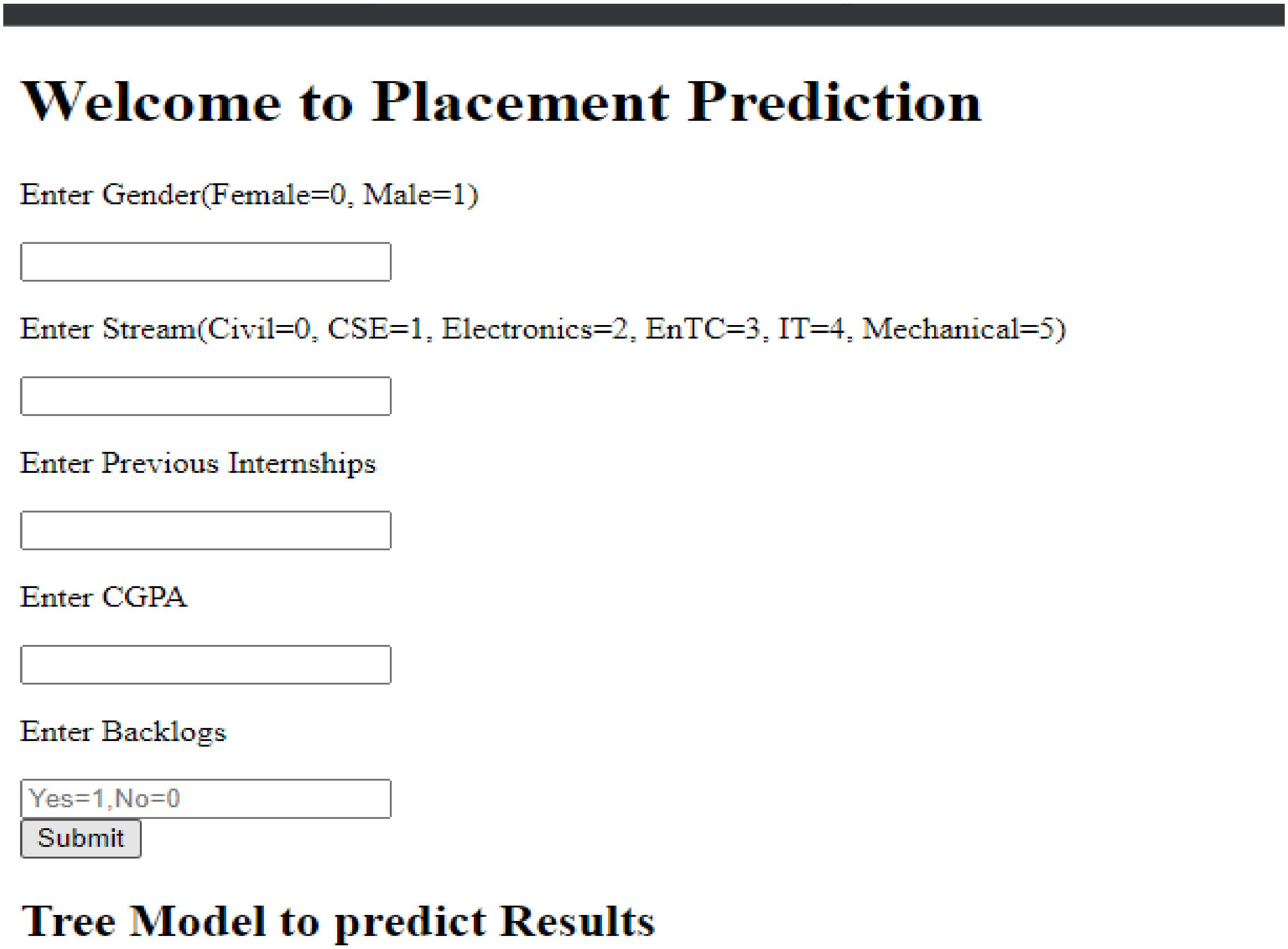
Building Html Pages

1) app.html
2) index.html
3) out.html



C:\Users\ELCOT\Desktop\NAAN MUTHALVAN\placement_prediction-master\placement_prediction-master\app.py
temp.py ×
app.py ×

```
import numpy as np
import model
from flask import Flask, request, render_template
import pickle

app = Flask(__name__,template_folder="templates")
model = pickle.load(open('model.pkl', 'rb'))

@app.route('/')
def home():
    return render_template('index.html')

@app.route('/predict',methods=['GET'])
def predict():

    gender = request.args.get('gender')
    stream = request.args.get('stream')
    internship = request.args.get('internship')
    cgpa = request.args.get('cgpa')
    backlogs = request.args.get('backlogs')
    arr = np.array([gender,stream,internship,cgpa,backlogs])
    brr = np.asarray(arr, dtype=float)
    output = model.predict([brr])
    if(output==1):
        out = 'Yes'
    else:
        out = 'No'
    return render_template('out.html', output=out)

if __name__ == "__main__":
    app.run(debug=True)
```



Welcome to Placement Prediction

Enter Gender(Female=0, Male=1)

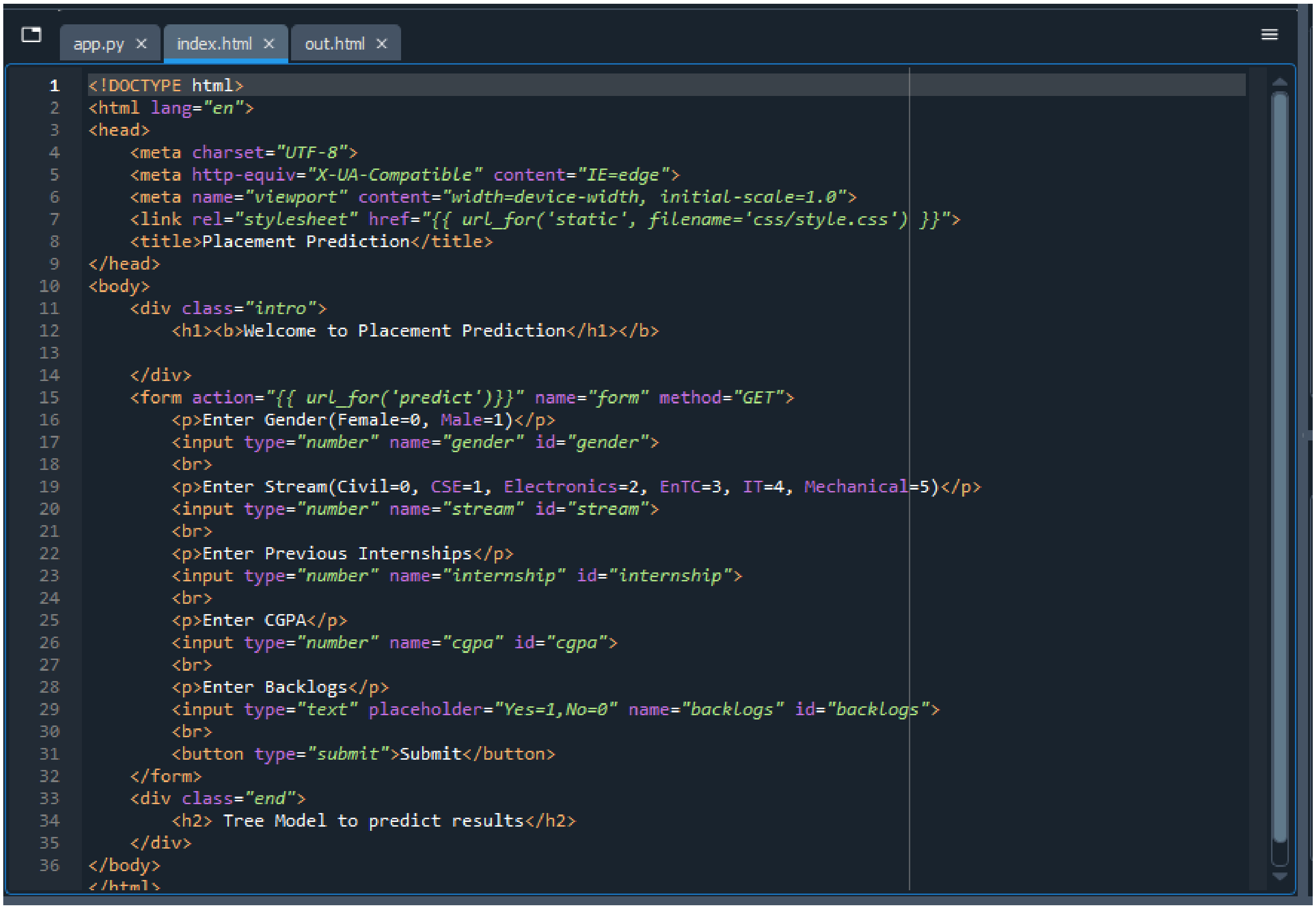
Enter Stream(Civil=0, CSE=1, Electronics=2, EnTC=3, IT=4, Mechanical=5)

Enter Previous Internships

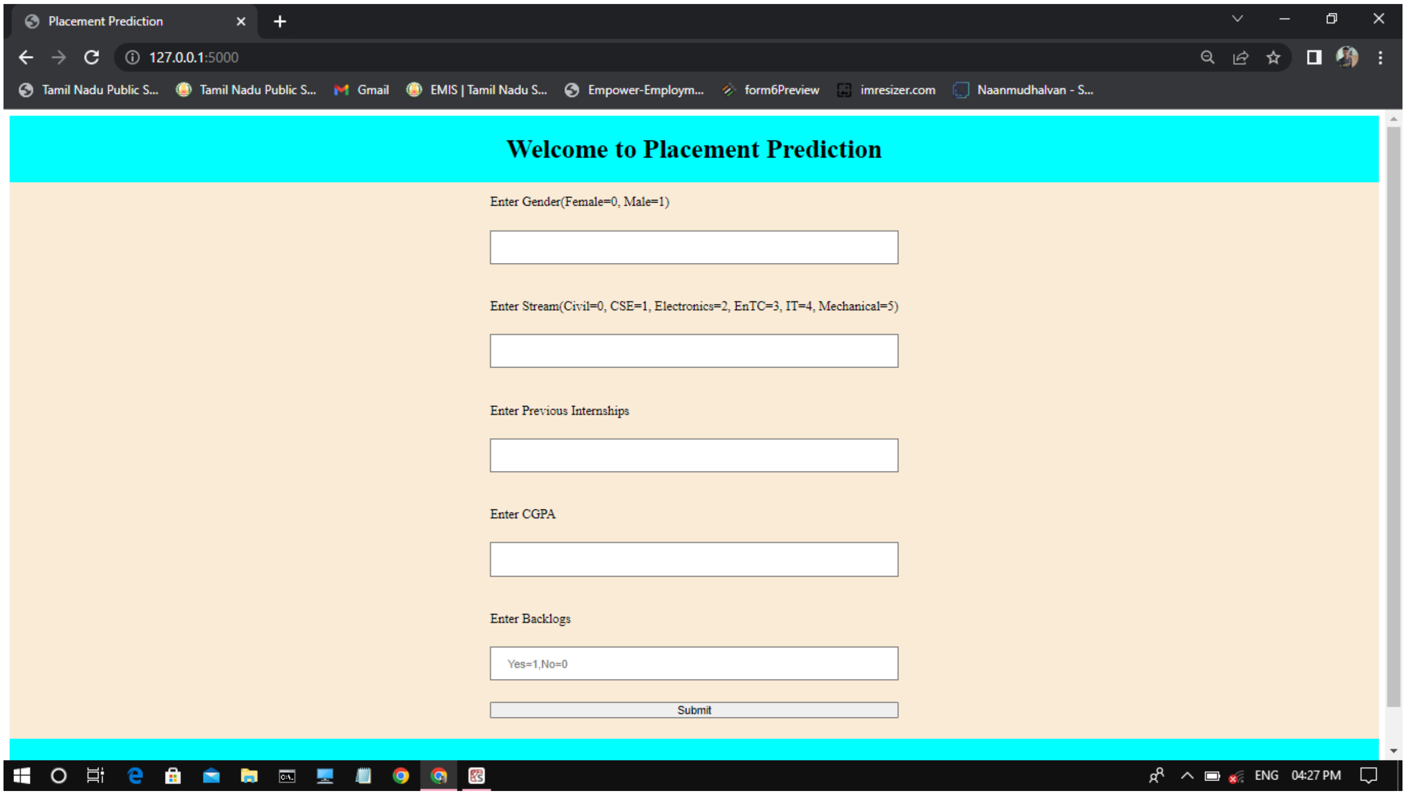
Enter CGPA

Enter Backlogs

Tree Model to predict Results



```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">
    <title>Placement Prediction</title>
</head>
<body>
    <div class="intro">
        <h1><b>Welcome to Placement Prediction</h1></b>

    </div>
    <form action="{{ url_for('predict')}}" name="form" method="GET">
        <p>Enter Gender(Female=0, Male=1)</p>
        <input type="number" name="gender" id="gender">
        <br>
        <p>Enter Stream(Civil=0, CSE=1, Electronics=2, EnTC=3, IT=4, Mechanical=5)</p>
        <input type="number" name="stream" id="stream">
        <br>
        <p>Enter Previous Internships</p>
        <input type="number" name="internship" id="internship">
        <br>
        <p>Enter CGPA</p>
        <input type="number" name="cgpa" id="cgpa">
        <br>
        <p>Enter Backlogs</p>
        <input type="text" placeholder="Yes=1,No=0" name="backlogs" id="backlogs">
        <br>
        <button type="submit">Submit</button>
    </form>
    <div class="end">
        <h2> Tree Model to predict results</h2>
    </div>
</body>
</html>
```



Welcome to Placement Prediction

Enter Gender(Female=0, Male=1)

Enter Stream(Civil=0, CSE=1, Electronics=2, EnTC=3, IT=4, Mechanical=5)

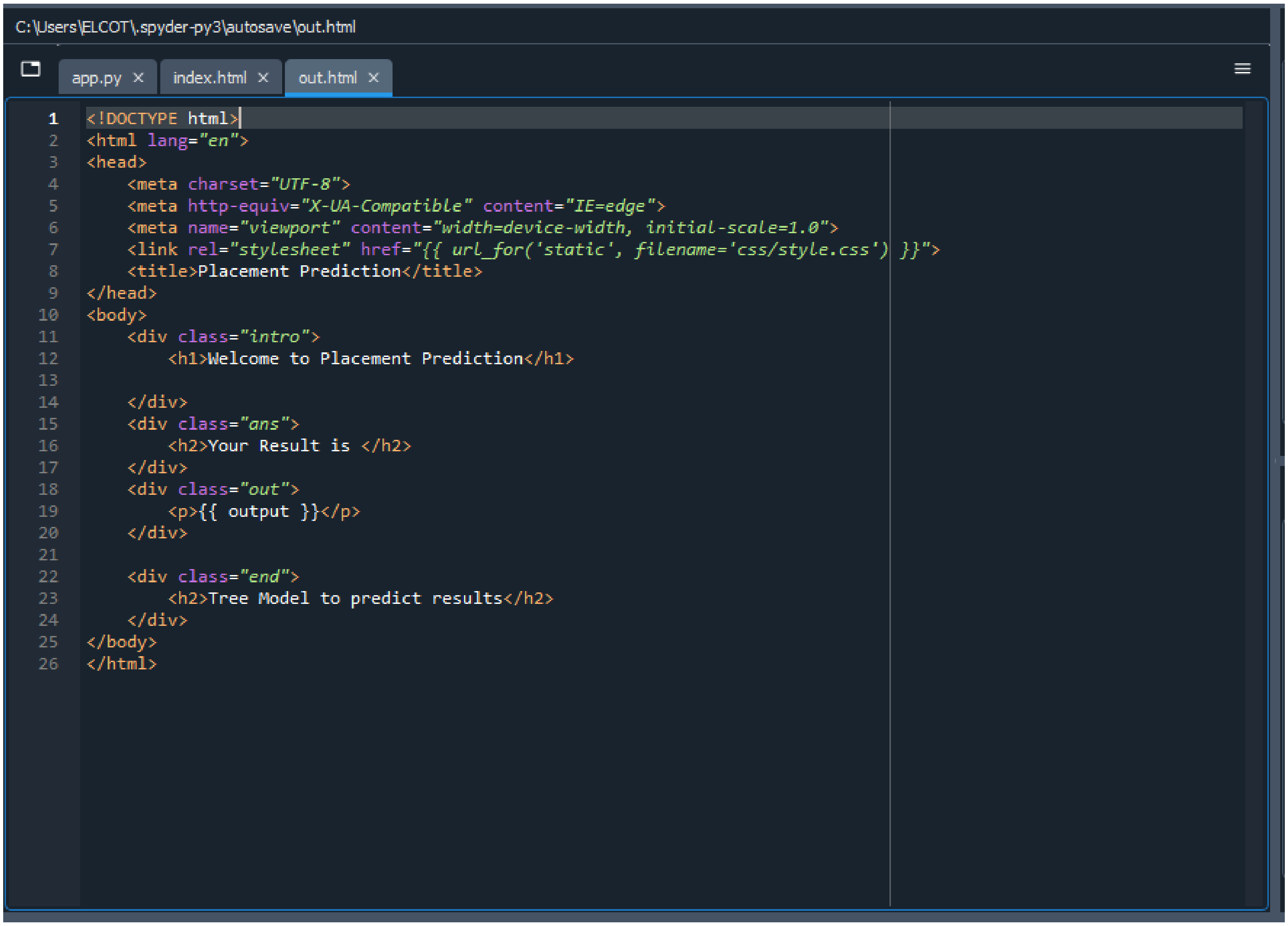
Enter Previous Internships

Enter CGPA

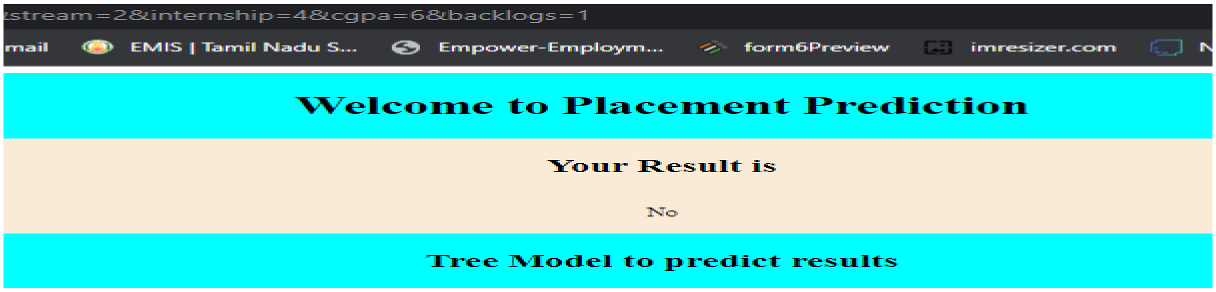
Enter Backlogs

Yes=1,No=0

Submit



```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">
    <title>Placement Prediction</title>
</head>
<body>
    <div class="intro">
        <h1>Welcome to Placement Prediction</h1>

    </div>
    <div class="ans">
        <h2>Your Result is </h2>
    </div>
    <div class="out">
        <p>{{ output }}</p>
    </div>

    <div class="end">
        <h2>Tree Model to predict results</h2>
    </div>
</body>
</html>
```



Welcome to Placement Prediction

Your Result is

No

Tree Model to predict results

4.Trailhead Profile Public URL :

Team Lead - https://trailblazer.me/id/banupriya2003

Team Member 1 - https://trailblazer.me/id/ramajeyanthi832003

Team Member 2 – http://trailblazer.me/id/mariaananthi2002

Team Member 3 – https://trailblazer.me/id/msurya2003

Team Member 4 – https://trailblazer.me/id/manisankar2003

5.ADVANTAGES & DISADVANTAGE

ADVANTAGES :

- **Efficient data processing :** Machine learning algorithms can Process large amounts of data in a relatively short amount of time Making it easier to analyze and identify patterns in campus placement data.

- **Improved decision- making :** With the help of machine learning, universities and colleges can make data-driven decisions to improve Their placement rates and identify areas of improvement.

Overall , machine learning can help universities and colleges better Understand their campus placement data and make informed decisions to improve the placement rates of their students.

DISADVANTAGE :

- **Bias** : Machine learning algorithms are only as unbiased as the data they are trained on. If there is bias in the historical campus Placement data, the machine learning.
- **Limited Data Availability :** Machine learning algorithms require Large amounts of high- quality data to be effective. If the campus Placement data is limited in scope or quality the machine learning Algorithm may not be able to identify meaningful patterns or trend.

6.APPLICATIONS :

- Train the model using a suitable machine learning algorithm such as SVM.
- The test set contains already predicted values.
- Its used for validating the predictions made by the training set.

7. CONCLUSION:

Training and Test data

Now the next step is to split our dataset into two. Training set and a Test set. We will train our machine learning models on our training set i.e our machine learning models.

8. FEATURE SCOPE :

Machine learning uses statistical patterns to make accurate predictions. Its sourcing the banking sector