CAMPUS PLACEMENT PREDICTION USING MACHINE LEARNING

TEAM_ID: NM2023TMID32101

	NAME OF THE STUDENT	NM ID
TEAM LEADER	M.SUBRAMANIAN	36ECE0BD52DB27187B41664DA08607
TEAM MEMBERS	T.DHARMARAJ	F9A26E9A7EE7D8818E4EF3463C5BA96
	S.DHANA VARADHA RAJAN	113546087F53FA09EF2401F47889E0D3
	R.SHANMUGA GANESH	48722CEEDEA2944B8444B08B4F4939E7
	S.MOHAMED SUBER KHAN	10F19EC2F10A7808CBF4EC87737C960EC

GOVERNMENT ARTS AND SCIENCE COLLEGE

KADAYANALLUR

INDEX

CAMPUS PLACEMENT PREDICTION

1. INDRODUCTION

OVERVIEW

Campus placement is a crucial aspect of the education system, as it connects students with job opportunities and helps them transition into their careers. By participating in campus placement programs, students can interact with potential employers, gain exposure to different industries, and secure job offers before they even graduate. For universities, campus placement is also important for attracting and retaining students. If a university has a strong track record of placing students in top companies and industries, it can be a major selling point for prospective students and their families. To facilitate campus placement, universities and companies often work together to organize job fairs, interviews, and other events. However, the process of matching students with job opportunities can be complex and time-consuming, which is where machine learning can come in. By using machine learning algorithms like the random forest classifier, we can analyze data on student performance, interests, and other factors to predict which students are most likely to be placed in jobs after graduation. This can help universities and companies optimize their campus placement strategies, leading to better outcomes for students and employers alike.

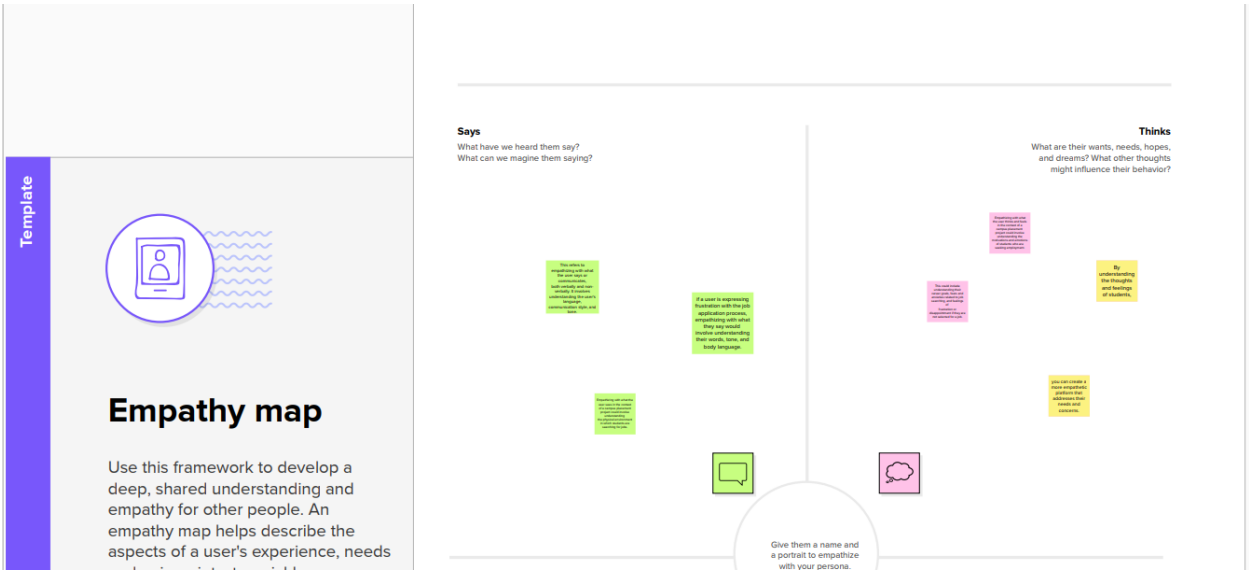
PURPOSE

The purpose of your campus placement prediction project is to develop a machine learning model that can accurately predict whether a student will be placed in a job after completing their studies. This model can be used to optimize campus placement strategies for universities and employers, leading to better outcomes for both parties. There are several benefits to developing an accurate campus placement prediction model. For universities, a better understanding of which students are most likely to be placed in jobs can help them tailor their programs and career services to better meet student needs. It can also help

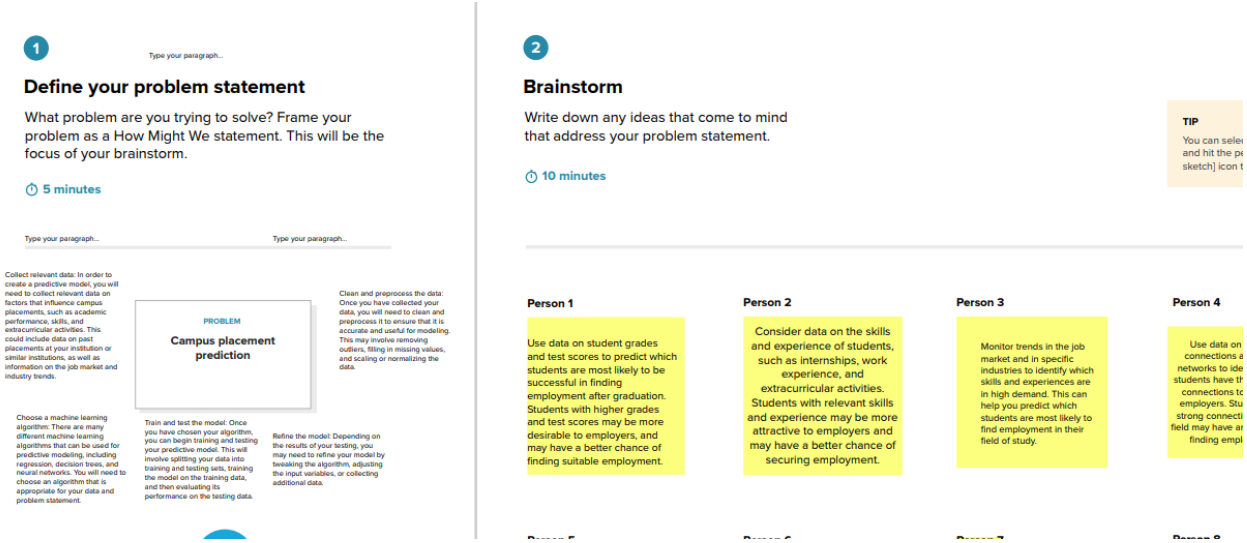
universities attract and retain students by demonstrating their commitment to providing practical career preparation. For employers, a more accurate campus placement model can help them identify top talent early in the recruitment process, reducing the time and resources required to fill open positions. It can also help employers build stronger relationships with universities by participating in campus placement programs and offering internships and other opportunities to students. Overall, the purpose of your project is to use machine learning to solve an important real-world problem and improve the campus placement process for both students and employers. By developing a high-quality prediction model, you can help optimize campus placement strategies and improve outcomes for everyone involved.

2.PROBLEM DEFINITION & DESIGN THINKING

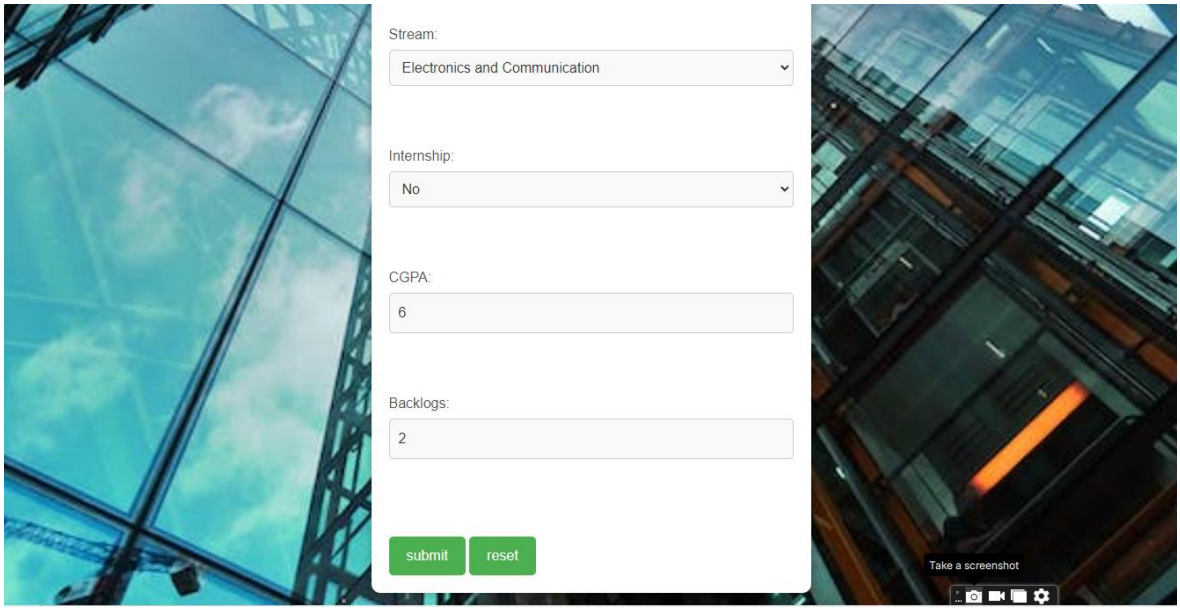
EMPATHY MAP



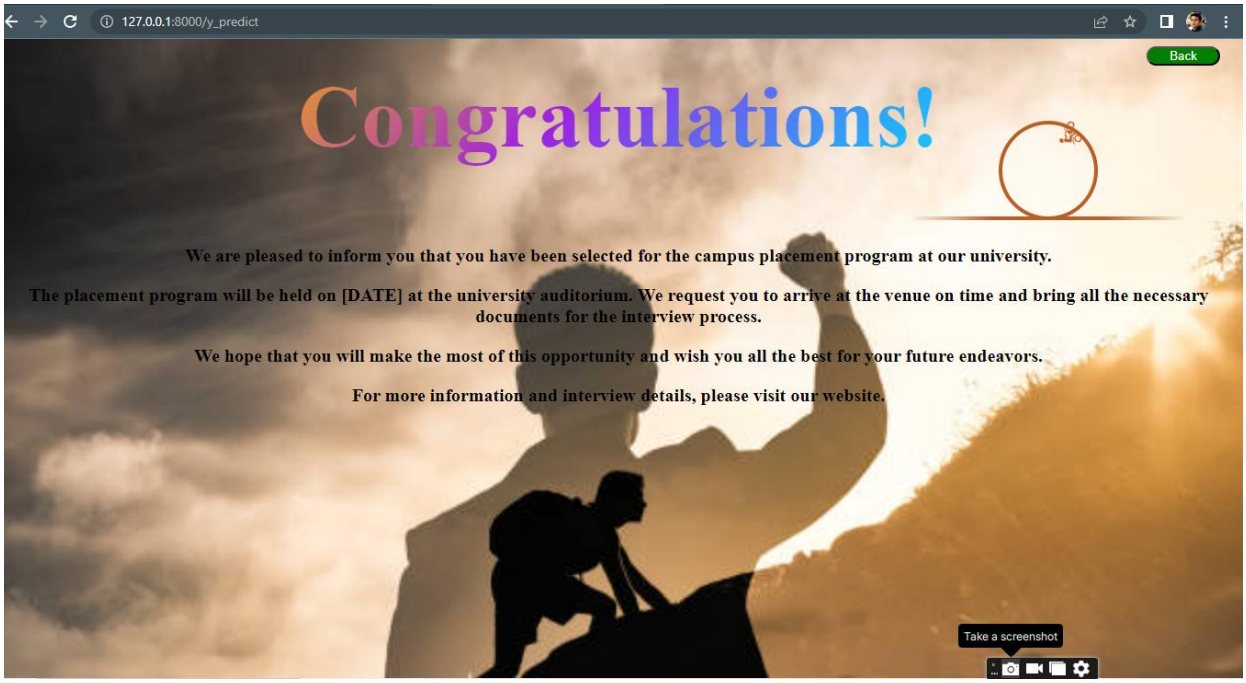
IDEATION & BRAINSTORMING MAP


1
Type your paragraph...
Define your problem statement
What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.
5 minutes
Type your paragraph...
Type your paragraph...
Collect relevant data: In order to create a predictive model, you will need to collect relevant data on factors that influence campus placements, such as academic performance, skills, and extracurricular activities. This could include data on past placements at your institution or similar institutions, as well as information on the job market and industry trends.
PROBLEM
Campus placement prediction
Clean and preprocess the data: Once you have collected your data, you will need to clean and preprocess it to ensure that it is accurate and useful for modeling. This may involve removing outliers, filling in missing values, and scaling or normalizing the data.
Choose a machine learning algorithm: There are many different machine learning algorithms that can be used for predictive modeling, including regression, decision trees, and neural networks. You will need to choose an algorithm that is appropriate for your data and problem statement.
Train and test the model: Once you have chosen your algorithm, you can begin training and testing your predictive model. This will involve splitting your data into training and testing sets, training the model on the training data, and then evaluating its performance on the testing data.
Refine the model: Depending on the results of your testing, you may need to refine your model by tweaking the algorithm, adjusting the input variables, or collecting additional data.
2
Brainstorm
Write down any ideas that come to mind that address your problem statement.
10 minutes
TIP
Person 1
Use data on student grades and test scores to predict which students are most likely to be successful in finding employment after graduation. Students with higher grades and test scores may be more desirable to employers, and may have a better chance of finding suitable employment.
Person 2
Consider data on the skills and experience of students, such as internships, work experience, and extracurricular activities. Students with relevant skills and experience may be more attractive to employers and may have a better chance of securing employment.
Person 3
Monitor trends in the job market and in specific industries to identify which skills and experiences are in high demand. This can help you predict which students are most likely to find employment in their field of study.
Person 4

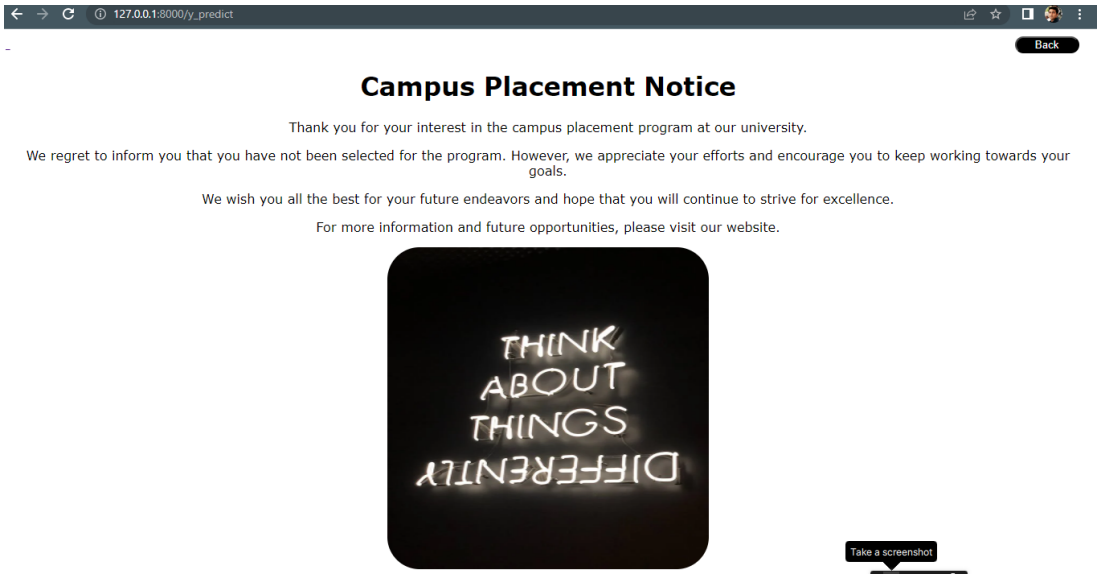
3.RESULT


Stream:
Electronics and Communication
Internship:
No
CGPA:
6
Backlogs:
2
submit
reset
Take a screenshot

STUDENT PLACED OUTPUT



STUDENT NOT PLACED OUTPUT



4.ADAVANTAGES AND DISADVANTAGES

ADVANTAGES

Increased accuracy: Machine learning algorithms like the random forest classifier can analyze large amounts of data and identify complex patterns that may not be apparent through traditional statistical methods. This can lead to more accurate predictions of which students are most likely to be placed in jobs after graduation**.** Optimization of resources: By accurately predicting which students are most likely to be placed in jobs, universities and employers can focus their resources on the students who are most likely to benefit from them. This can lead to more efficient use of time, money, and other resources. Improved decision-making: The insights provided by a machine learning model can help universities and employers make better decisions about which programs and strategies to pursue. For example, if the model shows that students with certain academic or extracurricular backgrounds are more likely to be placed in jobs, universities can focus on developing those programs and activities. Enhanced student outcomes: By improving the campus placement process, machine learning models can help more students secure jobs after graduation, leading to better outcomes for those students and their families. This can also help universities attract and retain students by demonstrating their commitment to providing practical career preparation.

DISADVANTAGES

Dependence on data quality and availability: The accuracy of the prediction model depends heavily on the quality and quantity of the data used to train it. If the data is incomplete or inaccurate, the model's predictions may not be reliable. Potential for bias in the data or model: If the data used to train the model contains bias or discriminatory factors, the model may perpetuate those biases and lead to unfair outcomes for certain groups of students. It's important to carefully evaluate and address any potential biases in the data and model. Complexity of the machine learning algorithms and potential difficulty in interpretation: Machine learning models can be complex and difficult to interpret, especially for stakeholders who are not familiar with the underlying algorithms and techniques. It's important to communicate the model's findings and limitations clearly and transparently.

Need for ongoing updates and maintenance of the model: Machine learning models require ongoing updates and maintenance to remain accurate and relevant. This can be time-consuming and resource-intensive. Possible resistance or skepticism from stakeholders who are unfamiliar with machine learning and predictive modeling techniques: Some stakeholders may be skeptical of the accuracy and validity of machine learning models, especially if they are not familiar with the underlying techniques. It's important to communicate the benefits and limitations of the model clearly and transparently to build trust and support among stakeholders.

5.APPLICATIONS

The model can be used to provide personalized career counseling to students, based on their individual strengths and weaknesses. This can help students make more informed decisions about their future career paths**.** Universities and job placement agencies can use the model to optimize their resources by focusing on the students who are most likely to be placed in jobs after graduation. This can lead to more efficient use of time, money, and other resources. The model can help universities plan their curricula to better align with the needs of the job market. By identifying the skills and experiences that are most valued by employers, universities can develop programs that prepare students for the most in-demand jobs. The model can be used by employers to identify the most promising candidates for their job openings. This can save time and resources by allowing employers to focus on the candidates who are most likely to succeed in their organizations. The insights provided by the model can be used by policymakers to make decisions about funding and support for education and job training programs. By identifying the factors that lead to successful job placement, policymakers can develop policies that better support students and job seekers.

6.CONCLUSION

In conclusion, the development of a campus placement prediction model using machine learning algorithms like random forest classifier has many potential benefits for students, universities, and employers. By analyzing data on student characteristics and job placement outcomes, the model can accurately predict which students are most likely to be placed in jobs after graduation. This can lead to more efficient use of resources, better decision-making, and improved career outcomes for students. While there are potential challenges and drawbacks to developing such a model, including data quality and bias concerns and the complexity of the underlying algorithms, these challenges can be mitigated through careful data selection and evaluation, transparency in the model's findings and limitations, and ongoing updates and maintenance of the model. Overall, the campus placement prediction model has a wide range of practical applications in the education and job sectors, from career counseling and curriculum planning to employer recruitment and policymaking. By leveraging the power of machine learning algorithms, we can improve the campus placement process and help students achieve their career goals. One additional point to consider is the potential for scalability of the campus placement prediction model. Once developed and validated, the model can be applied to a larger dataset of students and job placement outcomes, which can help to refine and improve its accuracy. This scalability can also allow for the model to be applied across different universities and geographic regions, which can help to identify broader trends and insights into the job market. Furthermore, the scalability of the model can enable universities and employers to make better-informed decisions about where to allocate their resources, leading to more efficient and effective campus placement processes.

7.FUTURE SCOPE

The development of a campus placement prediction model using machine learning algorithms like random forest classifier has a significant future scope in the education and job sectors. The campus placement prediction model can be integrated with other predictive models, such as models that predict student retention and graduation rates. This can provide a more comprehensive view of student outcomes and help universities make more informed decisions about resource allocation and program development. The model can be expanded to include new variables and data sources, such as social media data and job market trends. This can improve the accuracy and relevance of the model's predictions and provide new insights into the factors that influence job placement outcomes. Explainable AI techniques can be used to make the model more transparent and interpretable. This can help to build trust and support among stakeholders and enable better decision-making based on the model's findings. The machine learning techniques used in the campus placement prediction model can be applied to other domains, such as healthcare and finance, to predict outcomes and inform decision-making. Overall, the future scope of the campus placement prediction model is vast, and there are many potential areas for research and development. By leveraging the power of machine learning algorithms and integrating new data sources and techniques, we can improve the accuracy and relevance of the model's predictions and help students achieve their career goals.

8.APPENDIX

Ipynb file :

```
    #import libraries
import numpy as np
import pandas as pd
import os
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn import svm
from sklearn.metrics import accuracy_score
from sklearn.ensemble import RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn import metrics
from sklearn.model_selection import cross_val_score
from sklearn import preprocessing
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
import joblib
from sklearn.metrics import accuracy_score
#read the dataset
df=pd.read_excel("D:\my python\placement2.xlsx")
```

```
df.head()

#num of columns and rows display

df.shape


#check the null values

df.info()



df.isnull().sum()


def transformationplot(feature):
    plt.figure(figsize=(12,5))
    plt.subplot(1,2,1)
    sns.distplot(feature)
transformationplot(np.log(df["Age"]))


#handing categorical values

df=df.replace(["Male"],[0])

df=df.replace(["Female"],[1])

df=df.replace(["Civil","Computer Science","Electrical","Electronics And Communication","Information Technology","Mechanical"],
```

```
        [0,1,2,3,4,5])
df=df.drop(["Hostel"],axis=1)
df


#Data analysis,visual analysis,univariate analysis
plt.figure(figsize=(12,5))
plt.subplot(1,2,1)
sns.distplot(df["CGPA"],color="r")


plt.figure(figsize=(12,5))
plt.subplot(1,2,1)
sns.distplot(df["PlacedOrNot"],color="r")


#Bivariate analysis ,plotting the count plot
plt.figure(figsize=(18,4))
plt.subplot(1,4,1)
sns.countplot(x="Gender",data=df)
plt.subplot(1,4,2)
sns.countplot(x="HistoryOfBacklogs",data=df)
plt.show()
```

```
#multivariate analysis

plt.figure(figsize=(20,5))

plt.subplot(131)

sns.countplot(data=df,x="PlacedOrNot",hue="CGPA")

#spliting the data  into train and test

X=df.iloc[:,df.columns!="PlacedOrNot"]

Y=df.iloc[:,df.columns=="PlacedOrNot"]

xtrain,xtest,ytrain,ytest=train_test_split(X,Y,test_size=0.2)

xtrain.head


#model building

from sklearn.ensemble import RandomForestClassifier

model=RandomForestClassifier()

model1=model.fit(xtrain,ytrain.values.ravel())


#predict output

predict_output=model1.predict(xtest) #to test the alg

print(predict_output)


#to check accuracy

acc=accuracy_score(predict_output,ytest)
```

```
print("The accuracy score for Rf",acc)


#save the model
import pickle
pickle.dump(model,open("myplacementnow.pkl","wb"))
Model=pickle.load(open("myplacementnow.pkl","rb"))
```

Python flask file:

```
 from flask import Flask, request, render_template,redirect,url_for
import numpy as np
import joblib
import pickle
app = Flask(__name__)
model1 =pickle.load(open('myplacementnow.pkl',"rb"))
ct=joblib.load("myplacementnow.pkl")

@app.route('/')
def hel():
    return render_template("current.html")
@app.route('/login')
def log():
    return render_template("login.html")

@app.route('/sec')
```

```
def hello():
    return render_template("index.html")


@app.route('/guest', methods=['GET','POST'])
def Guest():
    if request.method == 'POST':
        age = request.form['age']
        gender = request.form['gender']
        stream = request.form['stream']
        internship = request.form['internship']
        cgpa = request.form['cgpa']
        backlogs = request.form['backlogs']
        internship=request.form['internship']
    return render_template("index2.html")



@app.route('/y_predict',methods=["POST"])
def y_predict():
    x_test=[yo for yo in request.form.values()]
    predict_output=model1.predict([x_test])
    if predict_output==0:
        return render_template("unsel.html")
    else:
       return render_template("sel.html")
if __name__=="__main__":
 app.run(debug=True,port=8000)
```