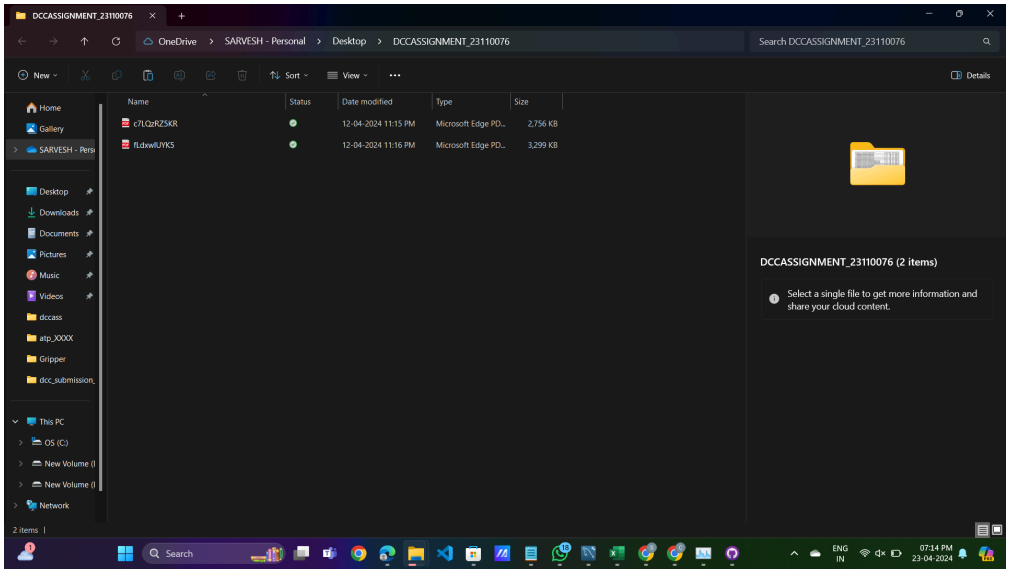
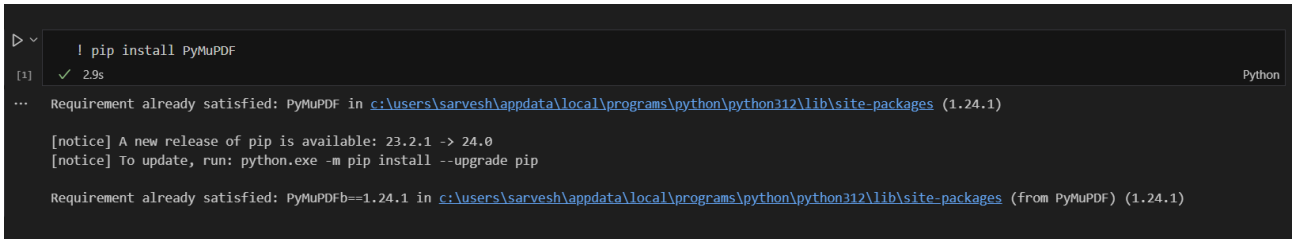
Problem Statement

1.b]

STEP] Downloading the pdf files from the given link and importing them in a new folder named “DCCASSIGNMENT_23110076”



STEP] Open The folder in the VS Code to convert the PDF files to csv.



Installing PyMuPDF module in VS code which will be used to convert PDF files to csv.

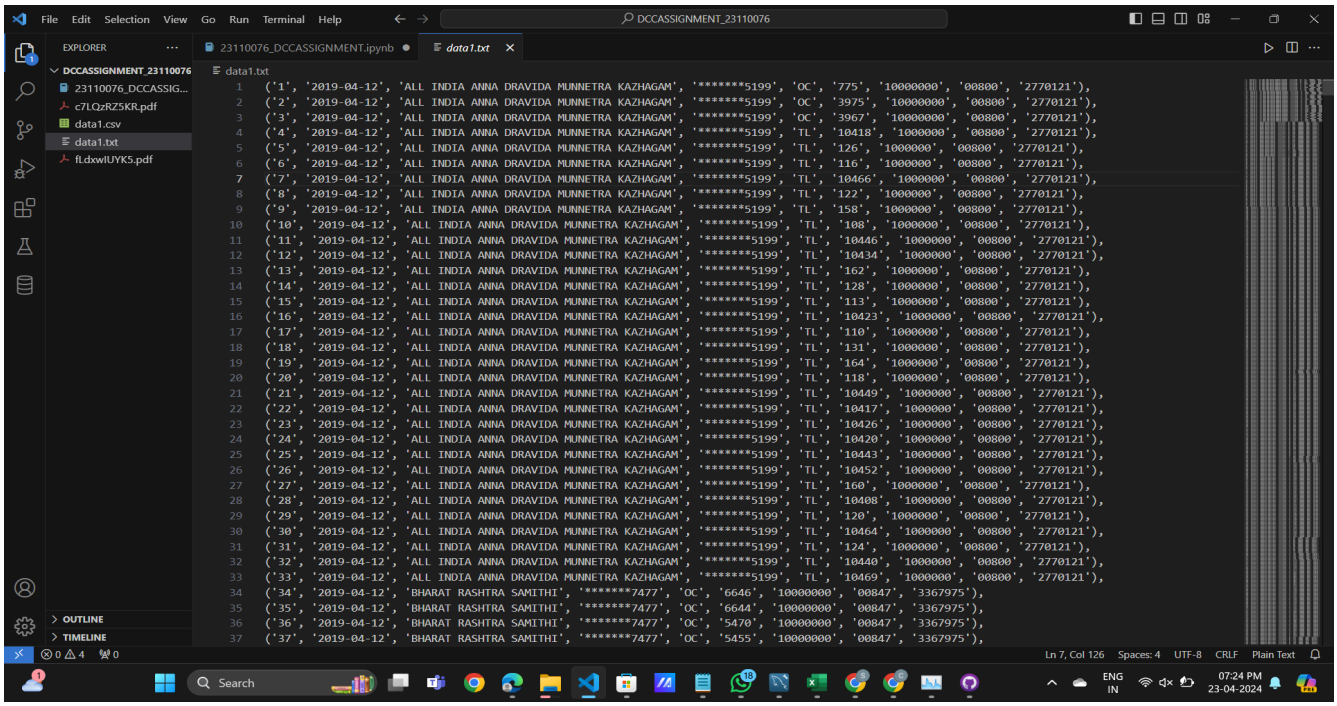
```
import fitz
import pandas as pd

document=fitz.open("c7LQzRZ5KR.pdf")

text=document[0].find_tables()[0].extract()
columns=text[0]

data=[[] for i in range(len(columns))]
i=0
while i<len(document):
  text = document[i].find_tables()[0].extract()
  for j in range(1,len(text)):
    for k in range(len(text[j])):
      data[k].append(text[j][k])
  i=i+1

f=[1]
for p in f:
  for i in range(len(data[p])):
    t=data[p][i].split("/")
    mon=["Jan","Feb","Mar","Apr","May","Jun","Jul","Aug","Sep","Oct","Nov","Dec"]
    monn=["01","02","03","04","05","06","07","08","09","10","11","12"]
    ing=mon.index(t[-2])
    t[-2]=monn[ing]
    dumb1=f"{t[-1]}-{t[-2]}-{t[-3]}"
    data[p][i]=dumb1

f=[6]
for j in f:
  for i in range(len(data[j])):
    t=data[j][i].split(",")
    l="".join(t)
    data[j][i]=l
```

Reading PDF files using the method “fitz.open()”. Iteration of tabular data of each page and getting access to which row of the table. Processing data such as changing the format of date from dd-mon-year to dd-mm-yyyy and changing the presentation of values in denomination ex. From 10,00,000 to 100000.

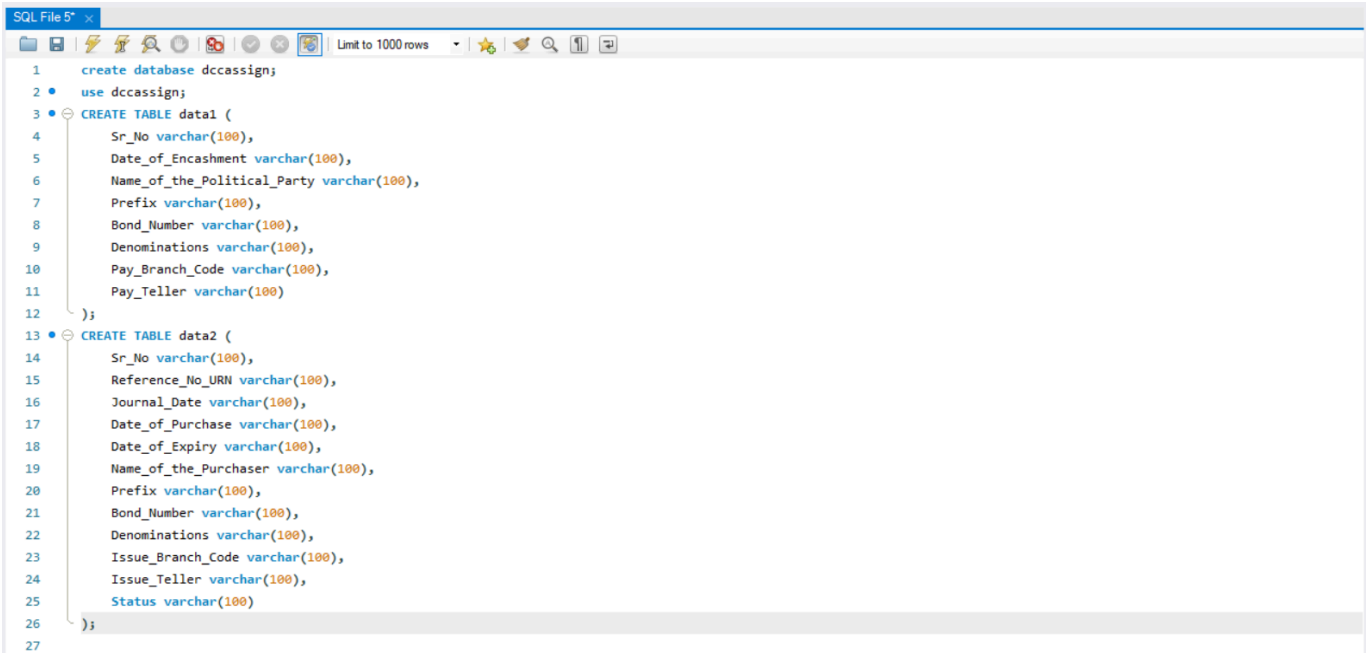
```
f=[6]
for j in f:
  for i in range(len(data[j])):
    t=data[j][i].split(",")
    l="".join(t)
    data[j][i]=l

dic={}
for i in range(len(data)):
  dic[columns[i]]=data[i]
display=pd.DataFrame(dic)

display.to_csv('data1.csv')
y=[]
for i in range(len(display)):
    x=list(display.iloc[i])
    x=str(x)
    x=x.replace('[','(')
    x=x.replace(']',')')
    y.append(x)
files=open("data1.txt","w")
for tuple in y:
    files.write(tuple+","+'\n')
files.close()
```

✓ 4m 40.7s

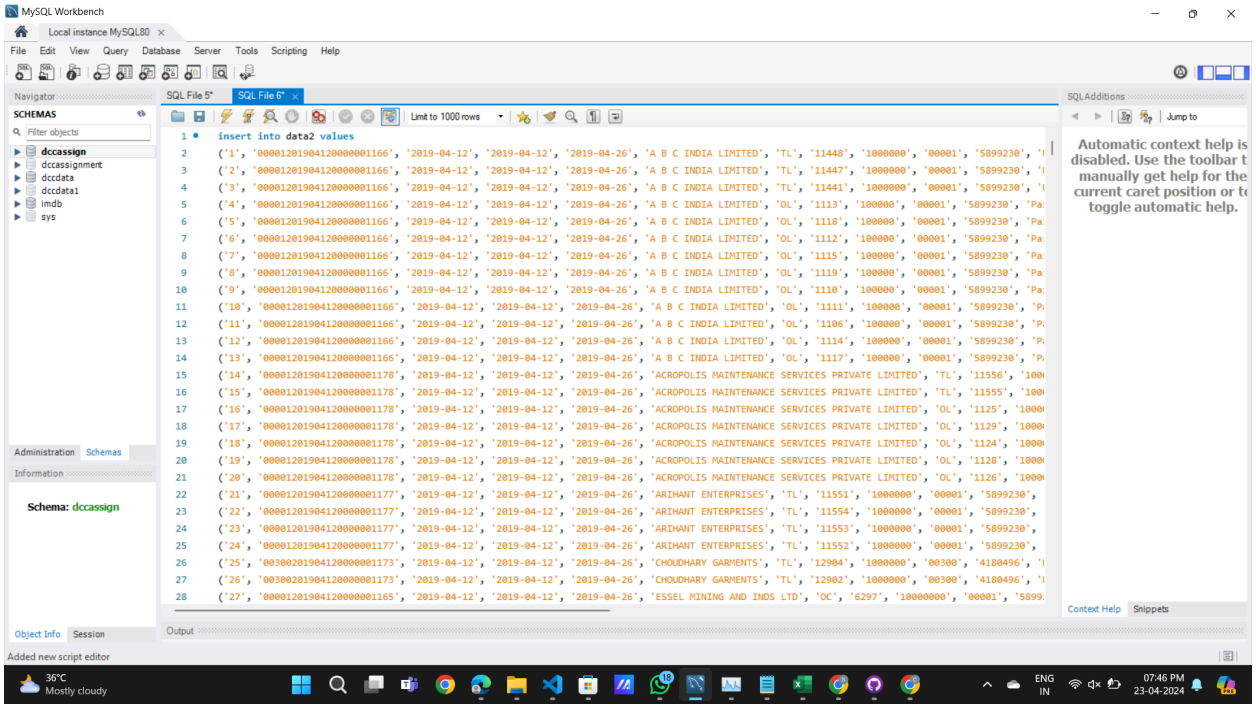
After processing data, creating a list for each column and then creating a dic and eventually creating a DataFrame with the help of pandas library. Creating a csv of given pdf files using the method pd.DataFrame.to_csv(“data1.csv”).
Inorder to create a table in database and inserting the data available ,creating a query in text form which becomes easier by simplifying copying entire text file to sql query.



Now,creating a database named “dccassign” and creating tables named as “data1” and “data2”



```sql
create database dccassign;
use dccassign;
CREATE TABLE data1 (
    Sr_No varchar(100),
    Date_of_Encashment varchar(100),
    Name_of_the_Political_Party varchar(100),
    Prefix varchar(100),
    Bond_Number varchar(100),
    Denominations varchar(100),
    Pay_Branch_Code varchar(100),
    Pay_Teller varchar(100)
);
CREATE TABLE data2 (
    Sr_No varchar(100),
    Reference_No_URN varchar(100),
    Journal_Date varchar(100),
    Date_of_Purchase varchar(100),
    Date_of_Expiry varchar(100),
    Name_of_the_Purchaser varchar(100),
    Prefix varchar(100),
    Bond_Number varchar(100),
    Denominations varchar(100),
    Issue_Branch_Code varchar(100),
    Issue_Teller varchar(100),
    Status varchar(100)
);
```



Inserting values in table via copying text from “data1.txt” to query file in mysql workbench.

Now connect by database “dccassign” to flask using the “flask” and “flask_mysqldb” modules.

Creating a flask application instance and specifying the folder in which the Flask should look for the html templates.

Then specifies the hostname ,user,password and database of the MySQL server.

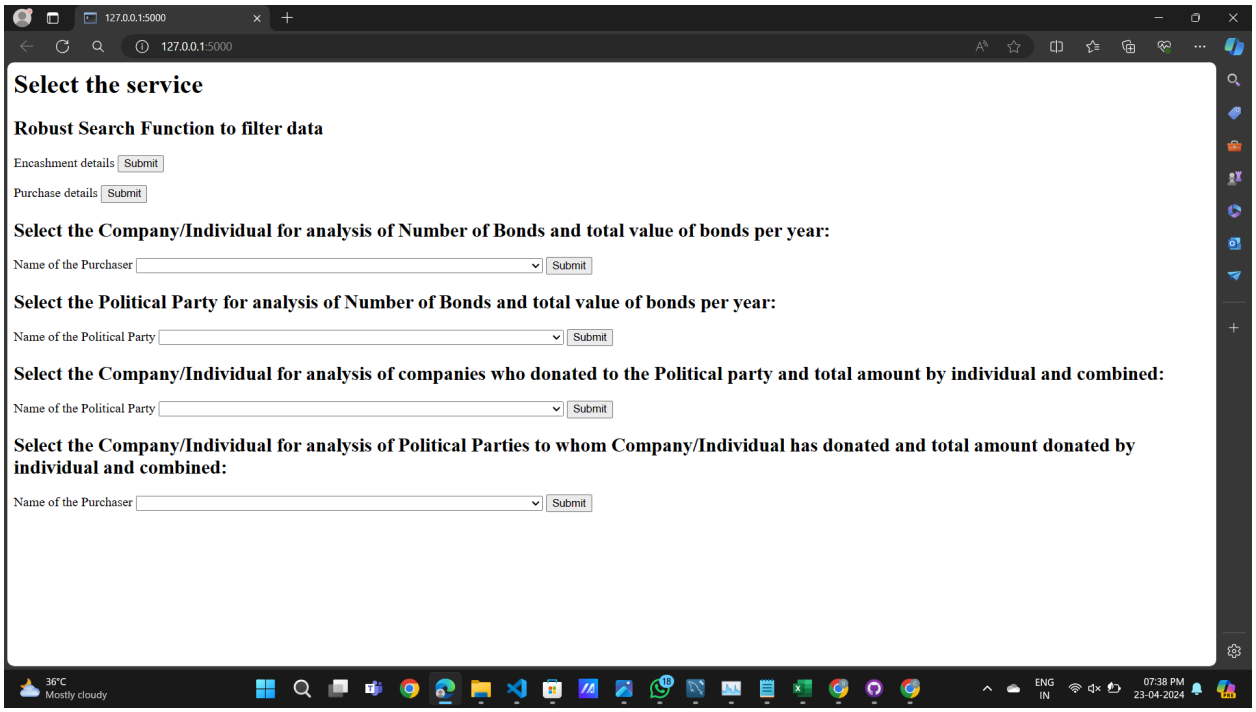
23110076_DCCASSIGNMENT.ipynb ● data2.txt app.py ×

app.py > ...

```python
from flask import Flask, redirect, url_for, request, Response, render_template
from flask_mysqldb import MySQL
import pandas as pd

app = Flask(__name__,template_folder="template")
app.config['MYSQL_HOST'] = 'localhost'
app.config['MYSQL_USER'] = 'root'
app.config['MYSQL_PASSWORD'] = 's@@@rvesh@@@123'
app.config['MYSQL_DB'] = 'dccassign'

mysql = MySQL(app)

def create(userdetails1,L):
    data={}
    for i in L:
        data[i]=[]
    for i in userdetails1:
        for j in range(len(i)):
            data[L[j]].append(i[j])
    data=pd.DataFrame(data)
    data=data.set_index("Sr_No")
    return data

@app.route('/')
def main_page():
    return render_template("index0.html")
```

```python
@app.route('/')
def main_page():
    return render_template("index0.html")

@app.route('/link1', methods = ["POST"])
def link1():
    return render_template("index.html")

@app.route('/link2', methods = ["POST"])
def link2():
    return render_template("index3.html")
```

1.e]
User Interface.

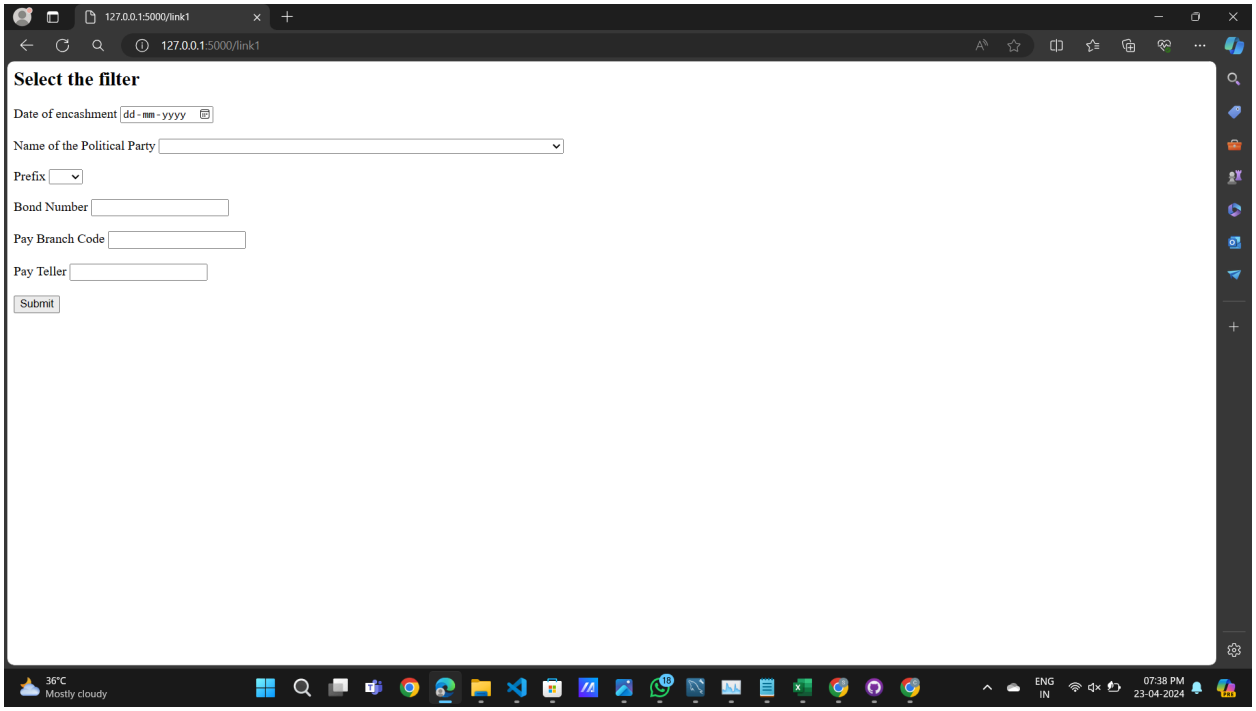


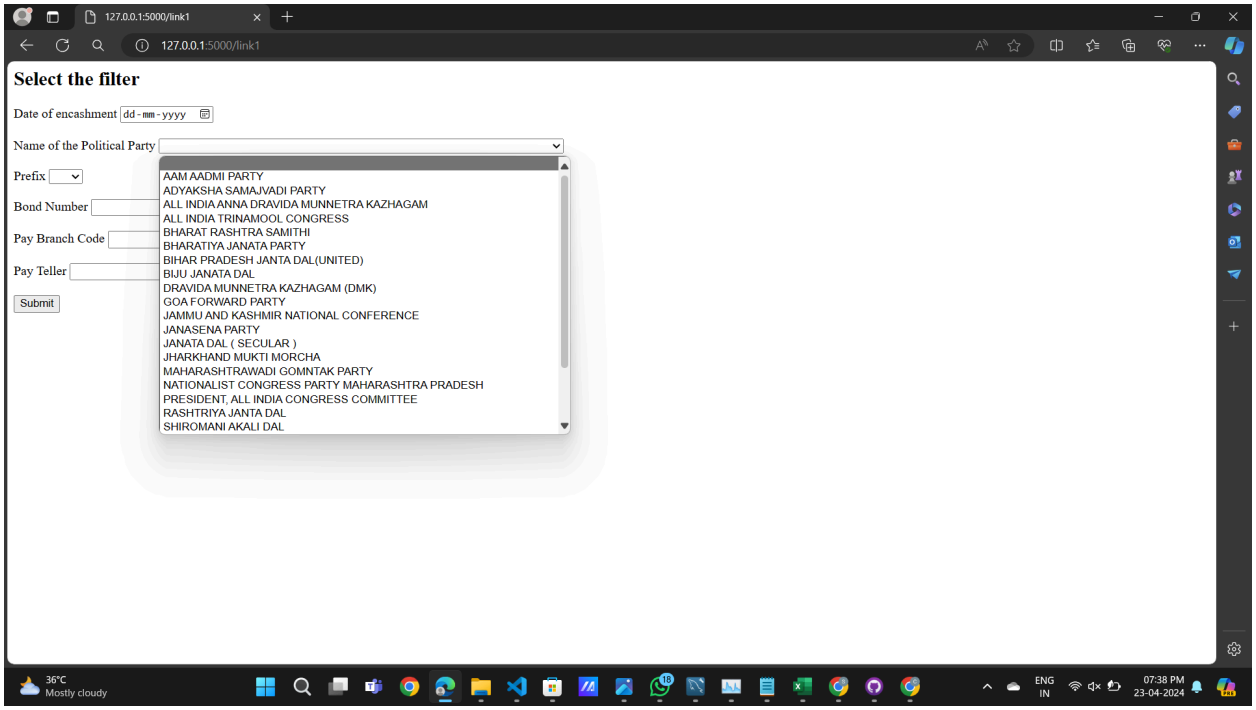
1]Implement robust search functionality that allows users to quickly search for specific records based on Bond Number or filter data based on any column in the table except Sr. No. and Status (e.g., date, political party, company name). The output should be displayed in the form of a table which is fetched from the database for a given query.

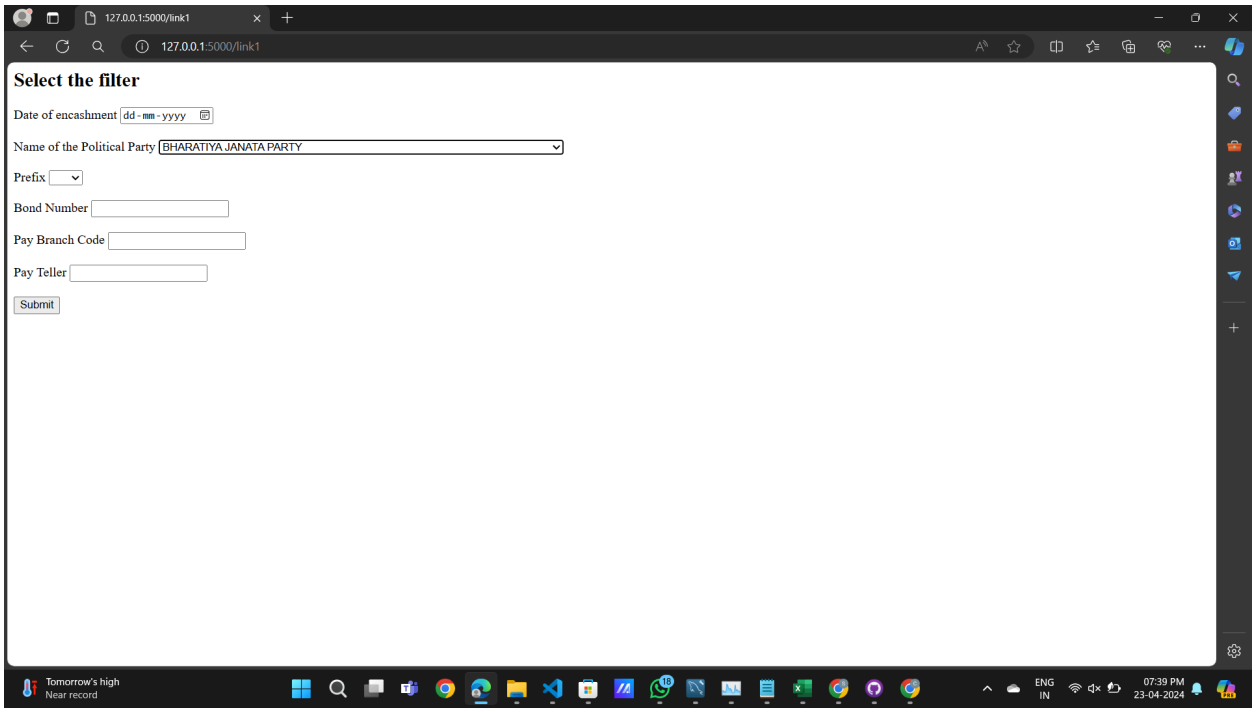
Function is used to filter data according to the filter selected from the user interface. Displaying the resultant data in the form of a table using pd.DataFrame.to_html().

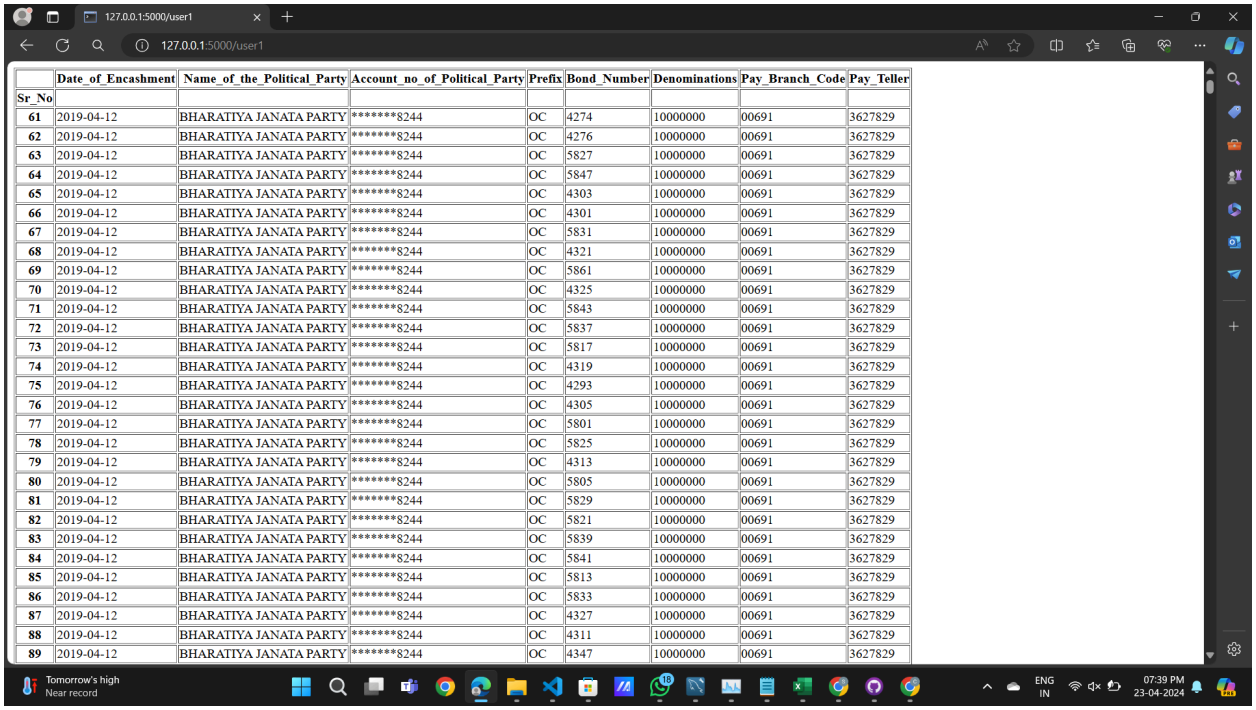
1]For analysis of data from the encashment table.

```python
@app.route('/user1', methods = ["POST"])
def user1():
    ps=["Date_of_Encashment","Name_of_the_Political_Party","Prefix","Bond_Number","Pay_Branch_Code","Pay_Teller"]
    date=request.form["date"]
    name=request.form["party"]
    prefix=request.form["prefix"]
    bnumber=request.form["bondnumber"]
    pcode=request.form["branchcode"]
    pteller=request.form["payteller"]
    check=[date,name,prefix,bnumber,pcode,pteller]
    cursor = mysql.connection.cursor()
    cursor.execute(f"select * from data1")
    userdetails=cursor.fetchall()
    L1=["Sr_No","Date_of_Encashment","Name_of_the_Political_Party","Account_no_of_Political_Party","Prefix","Bond_Number","Denominations","Pay_Branc
    data1=create(userdetails,L1)
    for i in range(len(ps)):
        if check[i]!="":
            if len(data1)!=0:
                data1=data1[data1[ps[i]]==check[i]]
    return data1.to_html()
```



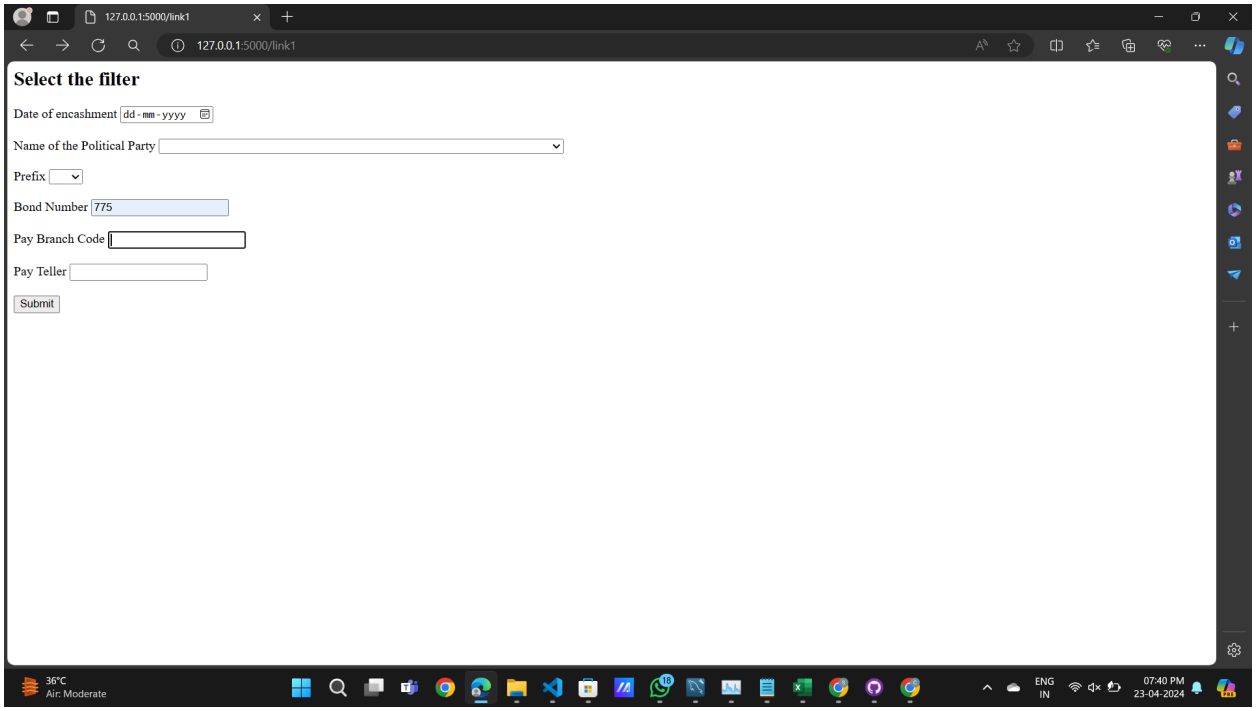

127.0.0.1:5000/link1
Select the filter
Date of encashment dd-mm-yyyy
Name of the Political Party
Prefix
Bond Number
Pay Branch Code
Pay Teller
Submit
AAM AADMI PARTY
ADYAKSHA SAMAJVADI PARTY
ALL INDIA ANNA DRAVIDA MUNNETRA KAZHAGAM
ALL INDIA TRINAMOOL CONGRESS
BHARAT RASHTRA SAMITHI
BHARATIYA JANATA PARTY
BIHAR PRADESH JANTA DAL(UNITED)
BIJU JANATA DAL
DRAVIDA MUNNETRA KAZHAGAM (DMK)
GOA FORWARD PARTY
JAMMU AND KASHMIR NATIONAL CONFERENCE
JANASENA PARTY
JANATA DAL (SECULAR)
JHARKHAND MUKTI MORCHA
MAHARASHTRAWADI GOMNTAK PARTY
NATIONALIST CONGRESS PARTY MAHARASHTRA PRADESH
PRESIDENT, ALL INDIA CONGRESS COMMITTEE
RASHTRIYA JANTA DAL
SHIROMANI AKALI DAL
36°C
Mostly cloudy
ENG
IN
07:38 PM
23-04-2024


127.0.0.1:5000/link1
Select the filter
Date of encashment dd-mm-yyyy
Name of the Political Party BHARATIYA JANATA PARTY
Prefix
Bond Number
Pay Branch Code
Pay Teller
Submit
Tomorrow's high
Near record
ENG
IN
07:39 PM
23-04-2024



127.0.0.1:5000/user1

Sr_No	Date_of_Encashment	Name_of_the_Political_Party	Account_no_of_Political_Party	Prefix	Bond_Number	Denominations	Pay_Branch_Code	Pay_Teller
61	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	4274	10000000	00691	3627829
62	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	4276	10000000	00691	3627829
63	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5827	10000000	00691	3627829
64	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5847	10000000	00691	3627829
65	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	4303	10000000	00691	3627829
66	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	4301	10000000	00691	3627829
67	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5831	10000000	00691	3627829
68	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	4321	10000000	00691	3627829
69	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5861	10000000	00691	3627829
70	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	4325	10000000	00691	3627829
71	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5843	10000000	00691	3627829
72	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5837	10000000	00691	3627829
73	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5817	10000000	00691	3627829
74	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	4319	10000000	00691	3627829
75	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	4293	10000000	00691	3627829
76	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	4305	10000000	00691	3627829
77	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5801	10000000	00691	3627829
78	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5825	10000000	00691	3627829
79	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	4313	10000000	00691	3627829
80	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5805	10000000	00691	3627829
81	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5829	10000000	00691	3627829
82	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5821	10000000	00691	3627829
83	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5839	10000000	00691	3627829
84	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5841	10000000	00691	3627829
85	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5813	10000000	00691	3627829
86	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	5833	10000000	00691	3627829
87	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	4327	10000000	00691	3627829
88	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	4311	10000000	00691	3627829
89	2019-04-12	BHARATIYA JANATA PARTY	*******8244	OC	4347	10000000	00691	3627829



127.0.0.1:5000/link1

Select the filter

Date of encashment dd-mm-yyyy

Name of the Political Party

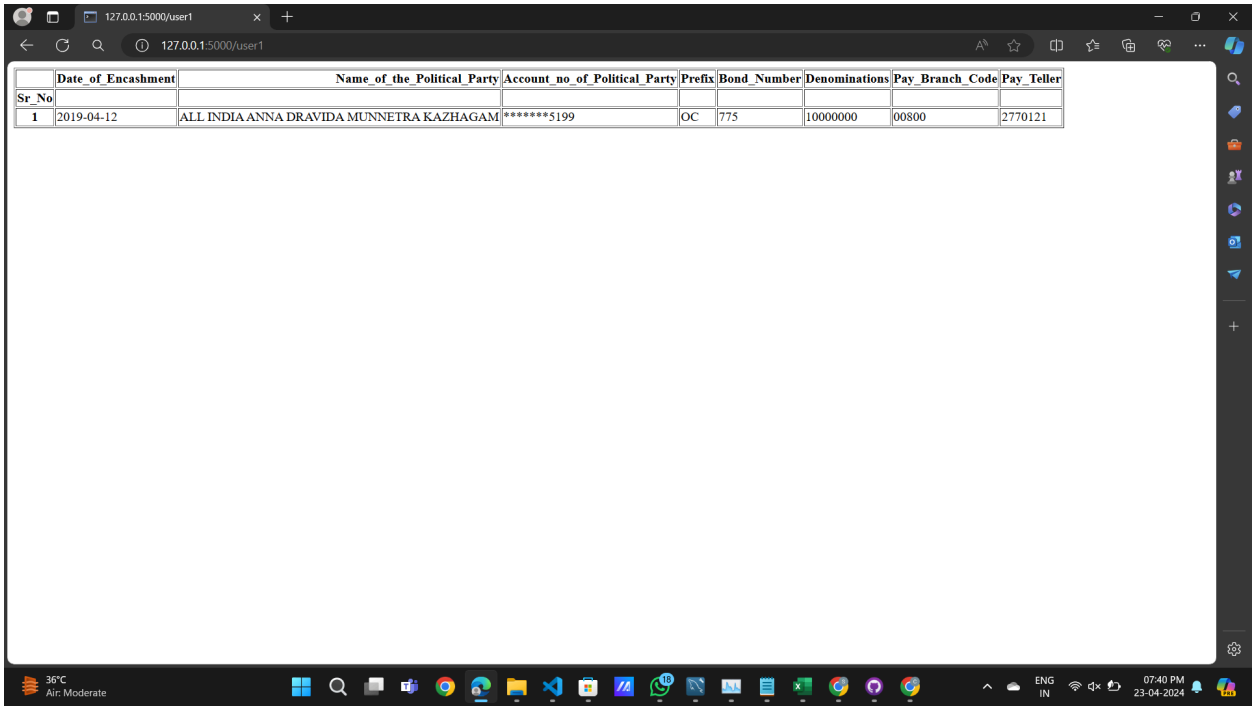
Prefix

Bond Number 775

Pay Branch Code

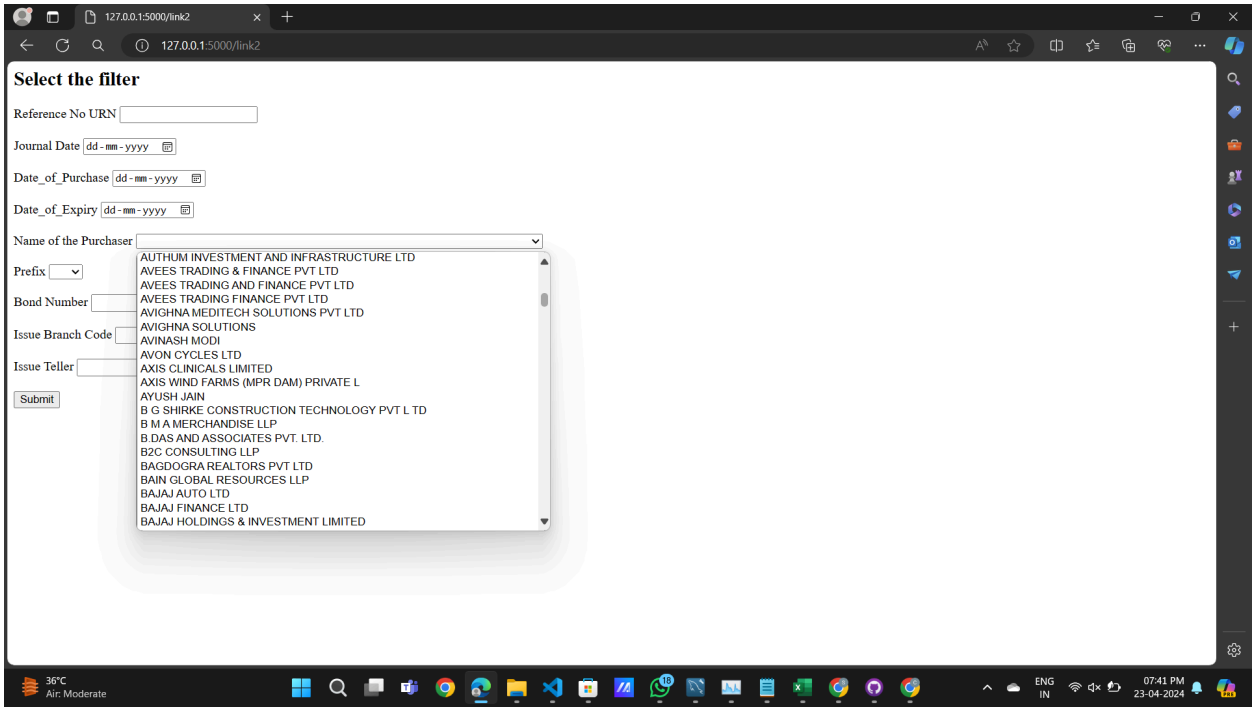
Pay Teller

Submit



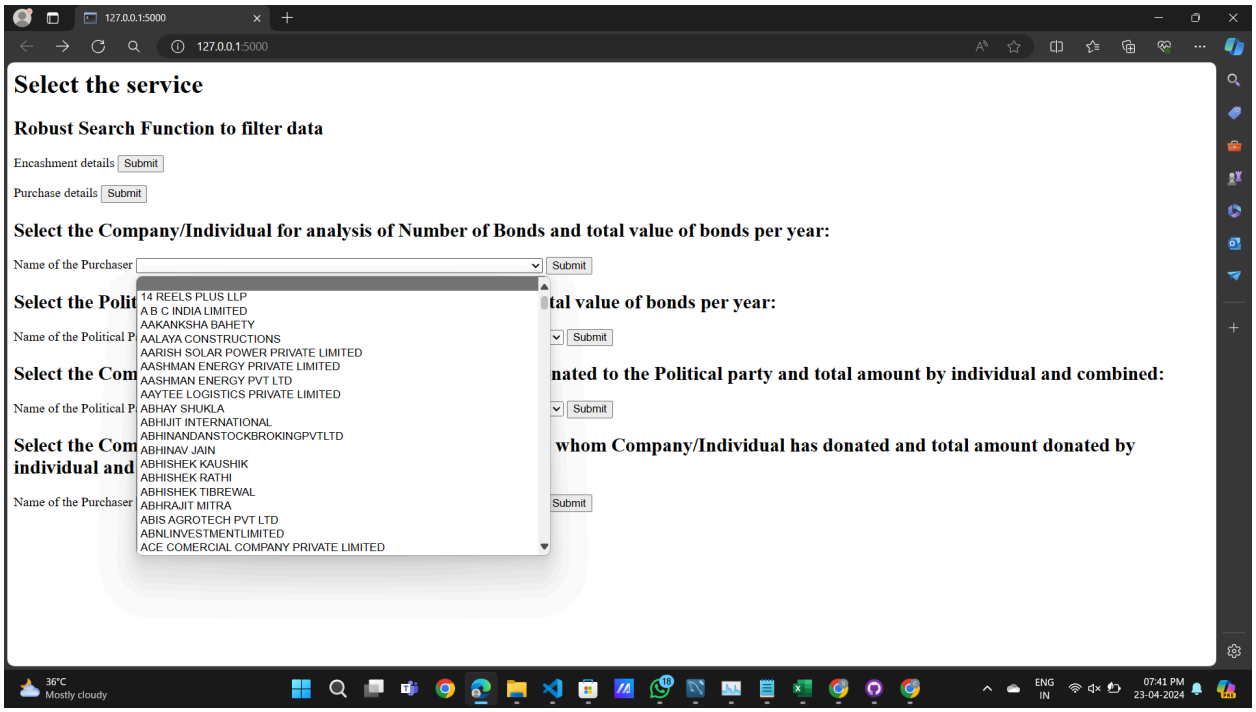
	Date_of_Encashment	Name_of_the_Political_Party	Account_no_of_Political_Party	Prefix	Bond_Number	Denominations	Pay_Branch_Code	Pay_Teller
Sr_No								
1	2019-04-12	ALL INDIA ANNA DRAVIDA MUNNETRA KAZHAGAM	*******5199	OC	775	10000000	00800	2770121

Similarly, for analysis of purchasing table

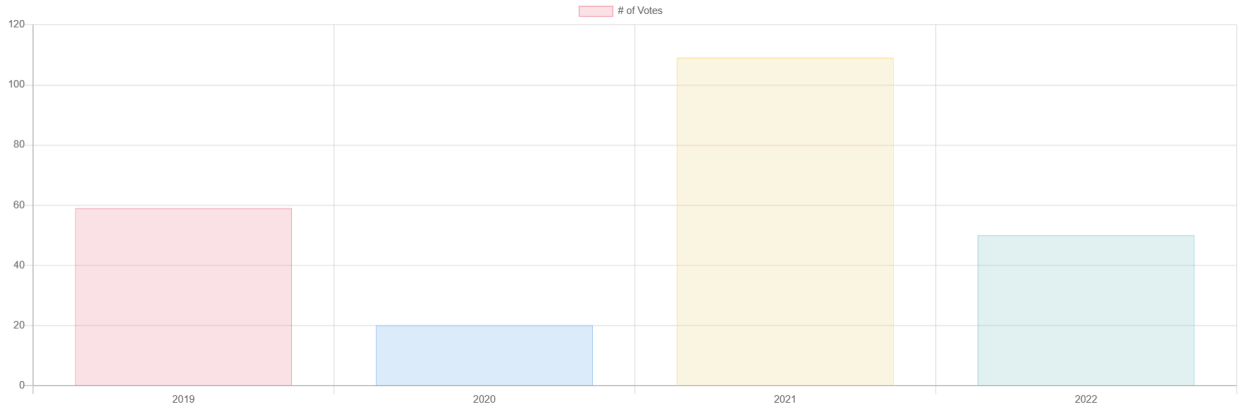


2]The option to select a Company/Individual from a drop-down/search, and show how many bonds and the total value of bonds purchased per year. You can present a bar plot depicting your results.

```python
@app.route('/user2', methods = ["POST"])
def user2():
    ps=["Reference_No_URN","Journal_Date","Date_of_Purchase","Date_of_Expiry","Name_of_the_Purchaser","Prefix","Bond_Number","Issue_Branch_Code","Is
    refer=request.form["refer"]
    date1=request.form["date1"]
    date2=request.form["date2"]
    date3=request.form["date3"]
    name=request.form["company"]
    prefix=request.form["prefix"]
    bnumber=request.form["bondnumber"]
    pcode=request.form["branchcode"]
    pteller=request.form["teller"]
    check=[f"{refer}",date1,date2,date3,name,prefix,bnumber,pcode,pteller]
    cursor = mysql.connection.cursor()
    cursor.execute(f"select * from data2")
    userdetails=cursor.fetchall()
    L1=["Sr_No","Reference_No_URN","Journal_Date","Date_of_Purchase","Date_of_Expiry","Name_of_the_Purchaser","Prefix","Bond_Number","Denominations"
    data1=create(userdetails,L1)
    for i in range(len(ps)):
        if check[i]!="":
            if len(data1)!=0:
                data1=data1[data1[ps[i]]==check[i]]
    return data1.to_html(escape=True)
```



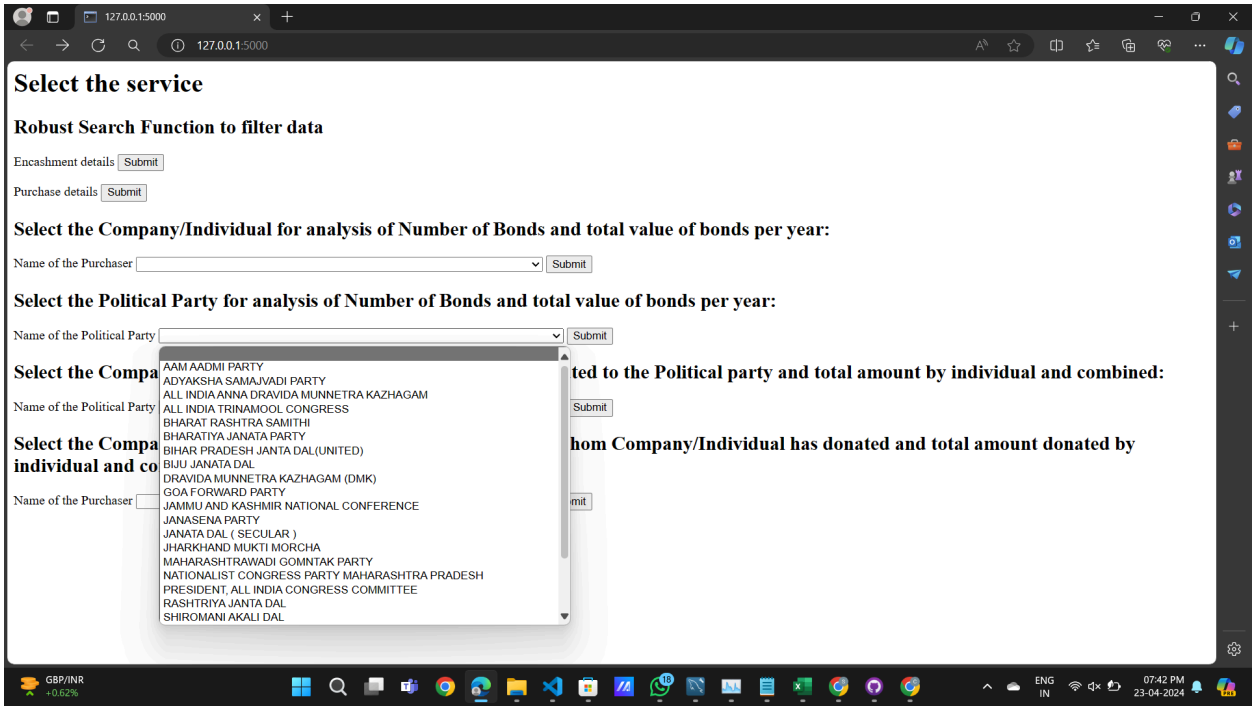
	Year	Number of bonds	Total value of bonds
Name of Purchaser			
ESSEL MINING AND INDS LTD	2019	59	590000000
ESSEL MINING AND INDS LTD	2020	20	200000000
ESSEL MINING AND INDS LTD	2021	109	1090000000
ESSEL MINING AND INDS LTD	2022	50	500000000

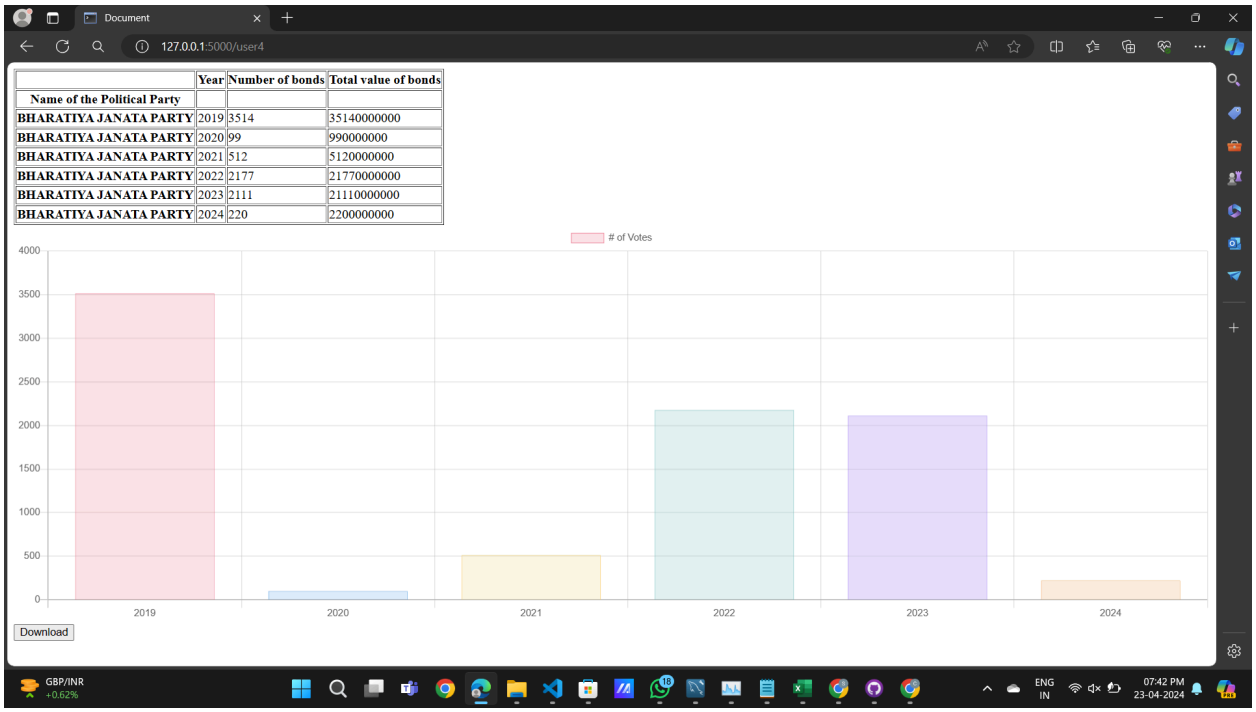


Download

3]The option to select a political party from a drop-down/search, and show how many bonds and total value per year are in the timeline. You can present a bar plot depicting your results.

```python
@app.route('/user3', methods = ["POST"])
def user3():
    name=request.form["company"]
    cursor = mysql.connection.cursor()
    cursor.execute(f"select * from data2")
    userdetails=cursor.fetchall()
    L1=["Sr_No","Reference_No_URN","Journal_Date","Date_of_Purchase","Date_of_Expiry","Name_of_the_Purchaser","Prefix","Bond_Number","Denominations"
    data1=create(userdetails,L1)
    year=list(data1[data1["Name_of_the_Purchaser"]==f"{name}"]["Date_of_Purchase"])
    don=list(data1[data1["Name_of_the_Purchaser"]==f"{name}"]["Denominations"])
    for i in range(len(year)):
        year[i]=year[i][:4]
    years=list(pd.Series(year).unique())
    donate=[]
    count=[]
    for i in range(len(years)):
        don1=[]
        count1=0
        for j in range(len(year)):
            if years[i]==year[j]:
                count1=count1+1
                don1.append(don[i])
        sum=0
        for i in range(len(don1)):
            don1[i]=int(don1[i])
            sum=sum+don1[i]
        donate.append(sum)
        count.append(count1)
    dic={}
    dic["Name of Purchaser"]=[name for i in range(len(count))]
    dic["Year"]=years
    dic["Number of bonds"]=count
    dic["Total value of bonds"]=donate
    dic=pd.DataFrame(dic)
    dic=dic.set_index("Name of Purchaser")
    return render_template('new1.html', page_data=dic.to_html().split("\n"), key=dic["Year"], val=dic["Number of bonds"])
```



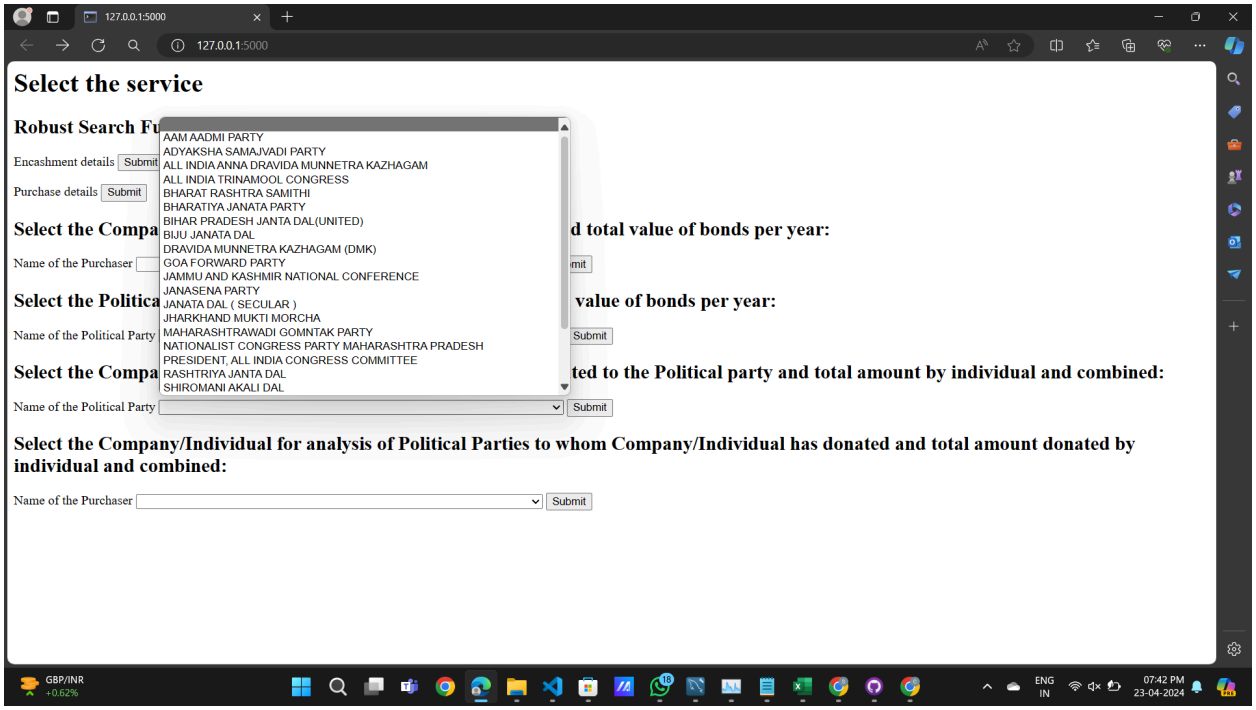


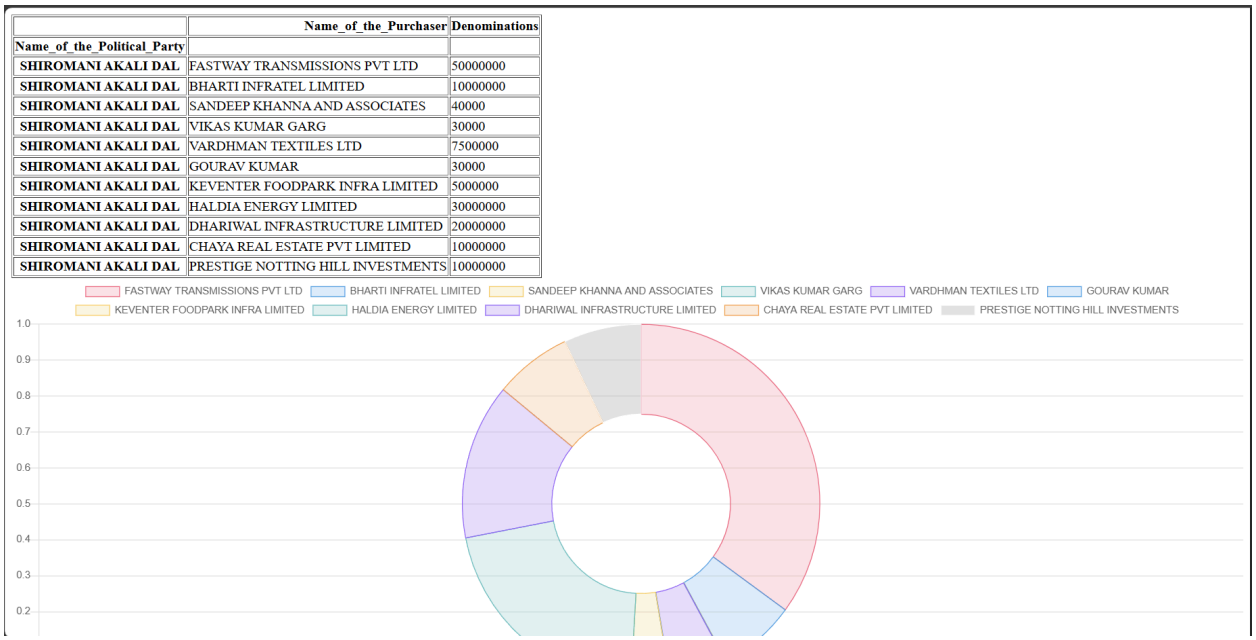
	Year	Number of bonds	Total value of bonds
Name of the Political Party			
BHARATIYA JANATA PARTY	2019	3514	35140000000
BHARATIYA JANATA PARTY	2020	99	990000000
BHARATIYA JANATA PARTY	2021	512	5120000000
BHARATIYA JANATA PARTY	2022	2177	21770000000
BHARATIYA JANATA PARTY	2023	2111	21110000000
BHARATIYA JANATA PARTY	2024	220	2200000000

4]The option to select a political party from a drop-down/search, shows which companies have donated to it and what amount individually and combined.

```python
@app.route('/user5', methods = ["POST"])
def user5():
    name=request.form["party"]
    cursor = mysql.connection.cursor()
    cursor.execute(f"select * from data1")
    userdetails=cursor.fetchall()
    L1=["Sr_No","Date_of_Encashment","Name_of_the_Political_Party","Account_no_of_Political_Party","Prefix","Bond_Number","Denominations","Pay_Branc
    data1=create(userdetails,L1)
    cursor.execute(f"select * from data2")
    userdetails=cursor.fetchall()
    L1=["Sr_No","Reference_No_URN","Journal_Date","Date_of_Purchase","Date_of_Expiry","Name_of_the_Purchaser","Prefix","Bond_Number","Denominations"
    data2=create(userdetails,L1)
    data1=data1[data1["Name_of_the_Political_Party"]==name]
    bond=list(data1["Bond_Number"])
    f=[]
    g=[]
    for i in range(len(bond)):
        data1=list(data2[data2["Bond_Number"]==bond[i]]["Name_of_the_Purchaser"])
        data11=list(data2[data2["Bond_Number"]==bond[i]]["Denominations"])
        f.append(data1)
        g.append(data11)
    newbond,newd,newf=[],[],[]
    for i in range(len(f)):
        if f[i]!=[]:
            for j in range(len(f[i])):
                newbond.append(bond[i])
                newf.append(f[i][j])
                newd.append(g[i][j])

    k=[name for i in range(len(newbond))]
    dic={}
    dic["Name_of_the_Political_Party"]=k
    dic["Name_of_the_Purchaser"]=newf
    dic["Bond_Number"]=newbond
    dic["Denominations"]=newd
    dic=pd.DataFrame(dic)
```

```python
    purchaser=dic["Name_of_the_Purchaser"].unique()
    donate=[]
    for i in range(len(purchaser)):
        dic1=list(dic[dic["Name_of_the_Purchaser"]==purchaser[i]]["Denominations"])
        sum=0
        for i in range(len(dic1)):
            dic1[i]=int(dic1[i])
            sum=sum+dic1[i]
        donate.append(sum)
    display={}
    display["Name_of_the_Political_Party"]=[name for i in range(len(purchaser))]
    display["Name_of_the_Purchaser"]=purchaser
    display["Denominations"]=donate
    display=pd.DataFrame(display)
    display=display.set_index("Name_of_the_Political_Party")
    return render_template('new2.html', page_data=display.to_html().split("\n"), key=display["Name_of_the_Purchaser"], val=display["Denominations"])
```



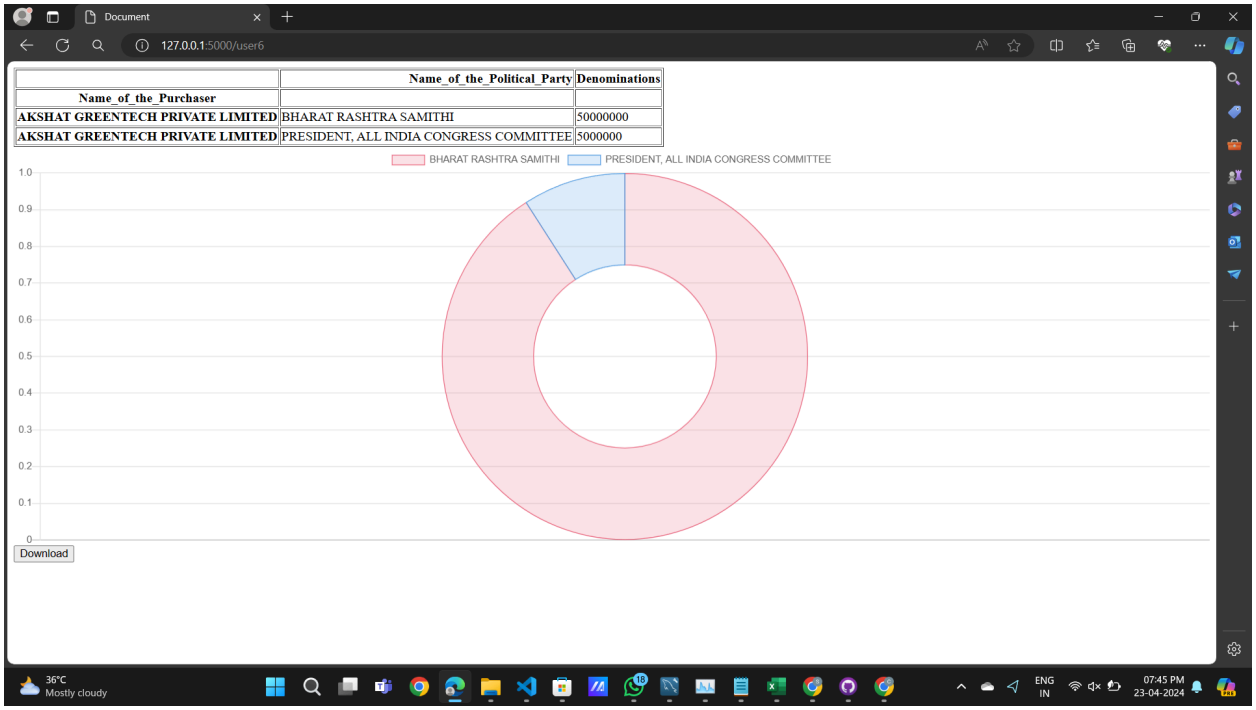


	Name_of_the_Purchaser	Denominations
Name_of_the_Political_Party		
SHIROMANI AKALI DAL	FASTWAY TRANSMISSIONS PVT LTD	50000000
SHIROMANI AKALI DAL	BHARTI INFRATEL LIMITED	10000000
SHIROMANI AKALI DAL	SANDEEP KHANNA AND ASSOCIATES	40000
SHIROMANI AKALI DAL	VIKAS KUMAR GARG	30000
SHIROMANI AKALI DAL	VARDHMAN TEXTILES LTD	7500000
SHIROMANI AKALI DAL	GOURAV KUMAR	30000
SHIROMANI AKALI DAL	KEVENTER FOODPARK INFRA LIMITED	5000000
SHIROMANI AKALI DAL	HALDIA ENERGY LIMITED	30000000
SHIROMANI AKALI DAL	DHARIWAL INFRASTRUCTURE LIMITED	20000000
SHIROMANI AKALI DAL	CHAYA REAL ESTATE PVT LIMITED	10000000
SHIROMANI AKALI DAL	PRESTIGE NOTTING HILL INVESTMENTS	10000000

5]Similarly, provide an option to select a company from a drop-down/search, showcasing which parties they have donated and what amount individually and combined.

```python
@app.route('/user6', methods = ["POST"])
def user6():
    name=request.form["company"]
    cursor = mysql.connection.cursor()
    cursor.execute(f"select * from data1")
    userdetails=cursor.fetchall()
    L1=["Sr_No","Date_of_Encashment","Name_of_the_Political_Party","Account_no_of_Political_Party","Prefix","Bond_Number","Denominations","Pay_Branc
    data1=create(userdetails,L1)
    cursor.execute(f"select * from data2")
    userdetails=cursor.fetchall()
    L1=["Sr_No","Reference_No_URN","Journal_Date","Date_of_Purchase","Date_of_Expiry","Name_of_the_Purchaser","Prefix","Bond_Number","Denominations"
    data2=create(userdetails,L1)
    data2=data2[data2["Name_of_the_Purchaser"]==name]
    bond=list(data2["Bond_Number"])
    f=[]
    g=[]
    for i in range(len(bond)):
        data2=list(data1[data1["Bond_Number"]==bond[i]]["Name_of_the_Political_Party"])
        data11=list(data1[data1["Bond_Number"]==bond[i]]["Denominations"])
        f.append(data2)
        g.append(data11)
    newbond,newd,newf=[],[],[]
    for i in range(len(f)):
        if f[i]!=[]:
            for j in range(len(f[i])):
                newbond.append(bond[i])
                newf.append(f[i][j])
                newd.append(g[i][j])
    k=[name for i in range(len(newbond))]
    dic={}
    dic["Name_of_the_Purchaser"]=k
    dic["Name_of_the_Political_Party"]=newf
    dic["Bond_Number"]=newbond
    dic["Denominations"]=newd
    dic=pd.DataFrame(dic)
```

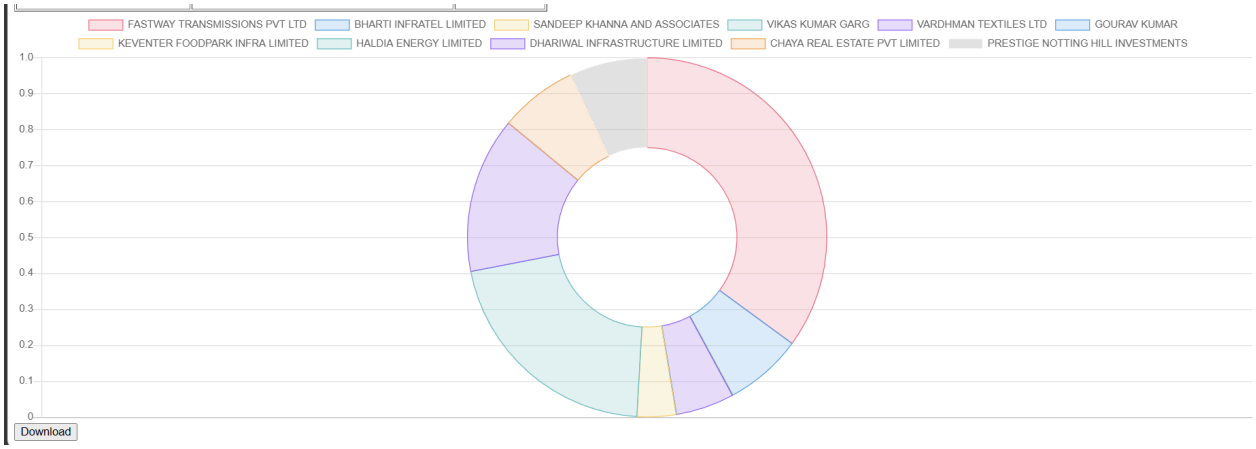
```python
                newd.append(g[i][j])
    k=[name for i in range(len(newbond))]
    dic={}
    dic["Name_of_the_Purchaser"]=k
    dic["Name_of_the_Political_Party"]=newf
    dic["Bond_Number"]=newbond
    dic["Denominations"]=newd
    dic=pd.DataFrame(dic)
    purchaser=dic["Name_of_the_Political_Party"].unique()
    donate=[]
    for i in range(len(purchaser)):
        dic1=list(dic[dic["Name_of_the_Political_Party"]==purchaser[i]]["Denominations"])
        sum=0
        for i in range(len(dic1)):
            dic1[i]=int(dic1[i])
            sum=sum+dic1[i]
        donate.append(sum)
    display={}
    display["Name_of_the_Purchaser"]=[name for i in range(len(purchaser))]
    display["Name_of_the_Political_Party"]=purchaser
    display["Denominations"]=donate
    display=pd.DataFrame(display)
    display=display.set_index("Name_of_the_Purchaser")
    return render_template('new2.html', page_data=display.to_html().split("\n"), key=display["Name_of_the_Political_Party"], val=display["Denominati

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

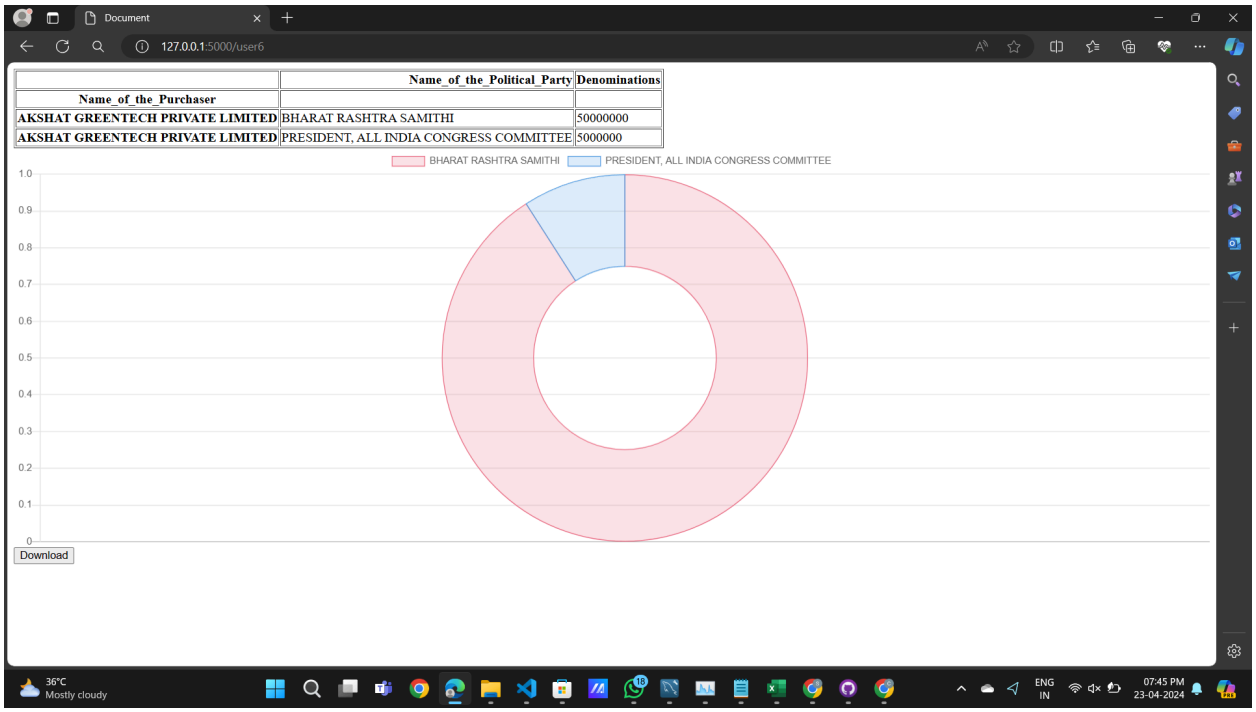


6]Apart from 1e4 and 1e5, you can also display the Pie chart depicting the total amount of donations to all the parties.

Select the Company/Individual for analysis of companies who donated to the Political party and total amount by individual and combined:

Name of the Political Party SHIROMANI AKALI DAL Submit





	Name_of_the_Political_Party	Denominations
Name_of_the_Purchaser		
AKSHAT GREENTECH PRIVATE LIMITED	BHARAT RASHTRA SAMITHI	50000000
AKSHAT GREENTECH PRIVATE LIMITED	PRESIDENT, ALL INDIA CONGRESS COMMITTEE	5000000