

INDEX

<u>S.No</u>	<u>Topic</u>	<u>Page No.</u>
1	Project Statement	1
2	Tables Used	2
3	Source Code	5
4	Project Screenshots	20
5	Limitations	31
6	References	31
7	Conclusion	31

ELECTRICITY BILLING AND ANALYSIS SYSTEM

Project Statement :-

Objective :-

The main objective of our project is to create an admin-user interface for managing electricity consumption data and billing system , where admins can manage data and the users can obtain information about their electricity consumption .

Portals available :

- Common portal :-
Provides option to access admin or user interface
- Admin portal :-
Admins must log in using their credentials and they get access to various options
 - Viewing data
 - Inserting data
 - Updating data
 - Deleting data
 - Visualise the data (bar graphs)
- User portal :-
Users must log in with their ID Number and the password assigned to them and can access various options
 - Viewing Consumption Data
 - Viewing Bill amount along with due date for payment
 - Generating a bill

Tables Used :-

1. Table customer:

- Contains information about users
- Database: electricity
- Creation Code :-

```
CREATE TABLE CUSTOMER(  
  
ID_NUMBER INT(4) ,  
  
CUSTOMER_NAME VARCHAR(50) ,  
  
ADDRESS VARCHAR(75) ,  
  
PHONE_NUMBER CHAR(10)) ;
```

- Table Data

id_number	CUSTOMER_NAME	ADDRESS	PHONE_NUMBER
1,001	SHIVA	ARUNDALE STREET,MYLAPORE	7905668276
1,002	SURYA	MUTHU STREET,MYLAPORE	6383256879
1,003	SHANKAR	SILVA ROAD	9894855667
1,004	SUBRAMANIAM	KARPAGAM AVENUE,MYLAPORE	9940181165
1,005	SANTHOSH	ADAM STREET,MYLAPORE	9924567895
1,007	STELLA	SANTHOME CHURCH,SANTHOME	7805863276
1,008	SKANDA	T.S.K ROAD,MYLAPORE	9947171175
1,009	SATHYA	ROHINI GARDENS,MYLAPORE	9894855544
1,006	SAM	NACHIYAPPAN STREET,MYLAPORE	9840057501

2. Table current_consumption:

- Contains information about consumption of electricity by customers
- Database: electricity
- Creation Code :-

```
CREATE TABLE CURRENT_CONSUMPTION(  
    ID_NUMBER CHAR(4),  
    PREVIOUS_MONTH_READING CHAR(5),  
    PRESENT_MONTH_READING CHAR(5),  
    UNITS_CONSUMED CHAR(5));
```

- Table Data

ID_NUMBER	PREVIOUS_MONTH_READING	PRESENT_MONTH_READING	UNITS_CONSUMED
1,001	1,000	1,500	500
1,002	1,500	2,500	1,000
1,003	2,000	2,700	700
1,004	2,500	2,850	350
1,007	1,000	1,500	500
1,008	2,500	2,750	250
1,005	3,000	4,000	1,000
1,006	2,000	3,000	1,000
1,009	3,000	4,000	1,000

3. Table electricity_bill:

- Contains information about electricity bill to be paid and the due date for payment
- Database: electricity
- Creation Code :-

```
CREATE TABLE ELECTRICITY_BILL(
ID_NUMBER CHAR(4),
UNITS_CONSUMED CHAR(5),
RATE_PER_UNIT INT(5),
BILL_AMOUNT INT(8),
PERCENTAGE_OF_ADDITIONAL_TAXES FLOAT,
TOTAL_AMOUNT INT(10),
DUE_DATE DATE);
```

- Table Data

id_number	UNITS_CONSUMED	RATE_PER_UNIT	BILL_AMOUNT	PERCENTAGE_OF_ADDITIONAL_TAXES	TOTAL_AMOUNT	DUE_DATE
1,001	500	9	4,500	10	4,950	2022-01-14
1,002	1,000	9	9,000	10	9,900	2022-01-14
1,003	200	9	1,800	10	1,980	2022-01-17
1,004	350	9	3,150	10	3,465	2022-01-17
1,005	1,000	9	9,000	10	9,900	2022-01-21
1,006	1,000	9	9,000	10	9,900	2022-01-17
1,007	500	9	4,500	10	4,950	2022-01-25
1,008	250	9	2,250	10	6,000	2022-02-04
1,009	1,000	9	9,000	10	9,900	2022-02-03

Source Code :-

```
import tkinter as tk
import pandas as pd
import mysql.connector as ms
import matplotlib.pyplot as plt
import datetime
from datetime import date
today=date.today()
#storing username and password
det_df=pd.DataFrame({'Username':['rithesh','vignesh','sankrant','1001','1002','1003','1004','1005','1006','1007','1008','1009'],
'Password':['RIT_adm_1','VIG_adm_2','SAN_adm_3','u_1','u_2','u_3','u_4','u_5','u_6','u_7','u_8','u_9']})
l_adm=[]
l_user=[]
for i in range(3):

l_adm.append([det_df.loc[i,'Username'],det_df.loc[i,'Password']])
for j in range(3,12):

l_user.append([det_df.loc[j,'Username'],det_df.loc[j,'Password']])

#defining admin commands
def adm():
    #creating admin portal
    ad_portal=tk.Tk()
    ad_portal.title('Admin portal')
    wina_width=600
    wina_height=700
    sca_width=ad_portal.winfo_screenwidth()
```

```

sca_height=ad_portal.winfo_screenheight()
center_xa=int(sca_width/2-wina_width/2)
center_ya=int(sca_height/2-wina_height/2)

ad_portal.geometry(f'{wina_width}x{wina_height}+{center_xa}
+{center_ya}')
ad_portal.configure(bg='blue')

#adding widgets-admin portal

tk.Label(ad_portal,text='welcome',bg='blue',fg='white',height=1,font=2).pack()
tk.Label(ad_portal,text='enter
username',bg='blue',fg='white',height=1,font=1).pack()
lp1=tk.Entry(ad_portal,width=20)
lp1.pack()
tk.Label(ad_portal,text='enter
password',bg='blue',fg='white',height=1,font=1).pack()
lp2=tk.Entry(ad_portal,show='*',width=20)
lp2.pack()
tk.Label(ad_portal,text='  ',bg='blue').pack()
def store_ad():
    usn=lp1.get()
    pss=lp2.get()
    list_1=[usn,pss]

con=ms.connect(host='localhost',user='root',passwd='rithesh27')
if list_1 in l_adm :
    tk.Label(ad_portal,text='welcome '+usn+' \nPlease select
one of the options below',bg='blue',fg='white',font=1).pack()
    tk.Label(ad_portal,text='To view the tables click
here',bg='blue',fg='white',font=1).pack()
    #view module

```

```

def view():

con=ms.connect(host='localhost',user='root',passwd='rithesh27',d
atabase='electricity')
    mycursor=con.cursor()
    mycursor.execute('SHOW TABLES')
    print('Tables available')
    for k in mycursor:
        print(k)
    a=str(input('enter table to display '))
    b='SELECT * FROM '+a
    c=pd.read_sql(b,con)
    print(c)
    print('\n\t')
tk.Button(ad_portal,text='View',command=view).pack()
tk.Label(ad_portal,text='  ',bg='blue').pack()
tk.Label(ad_portal,text='To insert new data click
here',bg='blue',fg='white',font=1).pack()
#create module
def insert():

```

```

con=ms.connect(host='localhost',user='root',passwd='rithesh27',d
atabase='electricity')
    mycursor=con.cursor()
    mycursor.execute('SHOW TABLES')
    print('Tables available')
    for k in mycursor:
        print(k)

```

```

con=ms.connect(host='localhost',user='root',passwd='rithesh27',d
atabase='electricity')
    mycursor=con.cursor()

```



```

t=str(input('enter name of table which has to be
changed '))
tg=pd.read_sql('SELECT * FROM '+t,con)
print(tg)
if t=='customer':
    print('enter ',tg.columns[0])
    var1=input()
    print('enter ',tg.columns[1])
    var2=input()
    print('enter ',tg.columns[2])
    var3=input()
    print('enter ',tg.columns[3])
    var4=input()
    if len(var4)!=10:
        print('please enter valid phone number')
        var4=input()
    s=( ' '+var1+',\'"+var2+\'"+var3+\'"+var4+\'" )'
    mycursor.execute('INSERT INTO '+t+' VALUES '+s)
    con.commit()
    print(pd.read_sql('SELECT * FROM '+t,con))
elif t=='current_consumption':
    print('enter ',tg.columns[0])
    var1=input()
    print('enter ',tg.columns[1])
    var2=input()
    print('enter ',tg.columns[2])
    var3=input()
    var4=str(int(var3)-int(var2))
    s=( ' '+var1+', '+var2+', '+var3+', '+var4+' )'
    mycursor.execute('INSERT INTO '+t+' VALUES '+s)
    con.commit()
    print(pd.read_sql('SELECT * FROM '+t,con))
elif t=='electricity_bill':

```

```

        print('enter ',tg.columns[0])
        var1=input()
        print('enter ',tg.columns[1])
        var2=input()
        var3='9'
        var4=str(int(var3)*int(var2))
        var5='10'
        var6=str(int(var4)+(int(var4)*int(var5)/100))
        print('enter ',tg.columns[6],' in the format of
yyyy-mm-dd ')
        var7=input()
        s='(
'+var1+', '+var2+', '+var3+', '+var4+', '+var5+', '+var6+', '"+var7+'\"')
        mycursor.execute('INSERT INTO '+t+' VALUES '+s)
        con.commit()
        print('\n\t')
        tk.Button(ad_portal,text='Insert',command=insert).pack()
        tk.Label(ad_portal,text='  ',bg='blue').pack()
        tk.Label(ad_portal,text='To update the data click
here',bg='blue',fg='white',font=1).pack()
        #update module
        def update():

con=ms.connect(host='localhost',user='root',passwd='rithesh27',d
atabase='electricity')
        mycursor=con.cursor()
        mycursor.execute('SHOW TABLES')
        print('Tables available')
        for k in mycursor:
            print(k)
            t=input('enter table which has to be updated ')
            dt=pd.read_sql('SELECT * FROM '+t,con)
            print(dt)

```

```

        uvar1=input('enter field name to be updated ')
        if t=='electricity_bill' and (uvar1=='due_date' or
uvar1=='DUE_DATE'):
            v1=input('enter ID_NUMBER ')
            y_d=pd.read_sql('SELECT YEAR(DUE_DATE)
FROM ELECTRICITY_BILL WHERE ID_NUMBER='+v1,con)
            y=int(y_d.loc[0,'YEAR(DUE_DATE)'])
            m_d=pd.read_sql('SELECT MONTH(DUE_DATE)
FROM ELECTRICITY_BILL WHERE ID_NUMBER='+v1,con)
            m=int(m_d.loc[0,'MONTH(DUE_DATE)'])
            d_d=pd.read_sql('SELECT DAY(DUE_DATE)
FROM ELECTRICITY_BILL WHERE ID_NUMBER='+v1,con)
            d=int(d_d.loc[0,'DAY(DUE_DATE)'])
            v2=input('enter new due_date ')
            while True:
                c_y=int(v2[0:4])
                c_m=int(v2[5:7])
                c_d=int(v2[8:])
                if (c_y>=y):
                    if c_m>=m:
                        if c_d>=d or c_m>m:
                            mycursor.execute('UPDATE '+t+' SET
'+uvar1+'= \''+v2+'\' WHERE ID_NUMBER='+v1)
                            con.commit()
                            print(pd.read_sql('SELECT * FROM
ELECTRICITY_BILL ',con))
                            break
                        else:
                            print('invalid due_date please enter again')
                            v2=input()
                    else:
                        print('invalid due_date please enter again')
                        v2=input()

```

```

        else:
            print('invalid due_date please enter again')
            v2=input()
    else:
        uvar2=input('enter new data ')
        print('any conditions ? type yes or no')
        opt=input()
        if opt=='yes'or opt=='YES':
            uvar3=input('enter field to apply conditions ')
            uvar4=input('enter condition ')
            mycursor.execute('UPDATE '+t+' SET '+uvar1+'=
\"+uvar2+\" WHERE '+uvar3+'= \"+uvar4+\"')
            con.commit()
        elif opt=='no'or opt=='NO':
            mycursor.execute('UPDATE '+t+' SET '+uvar1+'=
\"+uvar2+\"')
            con.commit()
        print(pd.read_sql('SELECT * FROM '+t,con))
        print('\n\t')

tk.Button(ad_portal,text='Update',command=update).pack()
tk.Label(ad_portal,text='  ',bg='blue').pack()
tk.Label(ad_portal,text='To remove data click
here',bg='blue',fg='white',font=1).pack()
#delete module
def delete():

con=ms.connect(host='localhost',user='root',passwd='rithesh27',d
atabase='electricity')
    mycursor=con.cursor()
    mycursor.execute('SHOW TABLES')
    print('Tables available')
    for k in mycursor:

```

```

        print(k)
        t=input('enter table from which data is to be deleted ')
        dt=pd.read_sql('SELECT * FROM '+t,con)
        print(dt)
        dvar1=input('enter field in which condition is applied ')
        dvar2=input('enter condition ')
        mycursor.execute('DELETE FROM '+t+' WHERE
'+dvar1+'= \''+dvar2+'\'')
        con.commit()
        print(pd.read_sql('SELECT * FROM '+t,con))
        print('\n\t')
        tk.Button(ad_portal,text='Delete',command=delete).pack()
        tk.Label(ad_portal,text='  ',bg='blue').pack()
        tk.Label(ad_portal,text='To plot a graph of the data click
here',bg='blue',fg='white',font=1).pack()
        #graph module
        def graph():

con=ms.connect(host='localhost',user='root',passwd='rithesh27',d
atabase='electricity')
        mycursor=con.cursor()
        mycursor.execute('SHOW TABLES')
        print('Tables available')
        for k in mycursor:
            print(k)
            opt='yes'
            while opt=='yes':
                a=input('enter table name ')
                if a=='current_consumption':
                    b='SELECT
ID_NUMBER,PREVIOUS_MONTH_READING,PRESENT_M
ONTH_READING FROM CURRENT_CONSUMPTION'
                    c=pd.read_sql(b,con)

```

```

        print(c)
        c.plot(kind='bar',x='ID_NUMBER')
        plt.ylabel('Units in kWh')
        plt.title('Current consumption data')
        plt.show()
    elif a=='electricity_bill':
        b='SELECT
ID_NUMBER,UNITS_CONSUMED,TOTAL_AMOUNT
FROM ELECTRICITY_BILL'
        c=pd.read_sql(b,con)
        print(c)

c.plot(kind='bar',x='ID_NUMBER',y='UNITS_CONSUMED',color='red')

        plt.ylabel('Units in kWh')
        plt.title('Units of current consumed by users')
        plt.show()

c.plot(kind='bar',x='ID_NUMBER',y='TOTAL_AMOUNT',color='blue')

        plt.ylabel('Rupees')
        plt.title('Amount paid by users')
        plt.show()
    else:
        print('Please enter valid table name')
        opt=input('do you want to plot other table? ')
        if opt=='no':
            break
    tk.Button(ad_portal,text='Graph',command=graph).pack()
    tk.Label(ad_portal,text=' ',bg='blue').pack()
    def close_ad():
        ad_portal.destroy()

```

```

        tk.Label(ad_portal,text='To exit click
here',bg='blue',fg='white',font=1).pack()

tk.Button(ad_portal,text='Logout',command=close_ad).pack()
        tk.Label(ad_portal,text='  ',bg='blue').pack()
    else:
        tk.Label(ad_portal,text='Incorrect username or
password.\nPlease try again',bg='blue',fg='white',font=1).pack()
        tk.Button(ad_portal,text='Login',command=store_ad).pack()
    ad_portal.mainloop()

```

#defining user commands

```
def usr():
```

```
    #creating user portal
```

```
    us_portal=tk.Tk()
```

```
    us_portal.title('User portal')
```

```
    winu_width=600
```

```
    winu_height=600
```

```
    scu_width=us_portal.winfo_screenwidth()
```

```
    scu_height=us_portal.winfo_screenheight()
```

```
    center_xu=int(scu_width/2-winu_width/2)
```

```
    center_yu=int(scu_height/2-winu_height/2)
```

```
us_portal.geometry(f'{winu_width}x{winu_height}+{center_xu}
+{center_yu}')

```

```
    us_portal.configure(bg='blue')
```

```
    #adding widgets-user portal
```

```
tk.Label(us_portal,text='Welcome',bg='blue',fg='white',font=1).pa
ck()

```

```
    tk.Label(us_portal,text='enter
username',bg='blue',fg='white',font=1).pack()

```

```

lu1=tk.Entry(us_portal,width=20)
lu1.pack()
tk.Label(us_portal,text='enter
password',bg='blue',fg='white',font=1).pack()
lu2=tk.Entry(us_portal,show='*',width=20)
lu2.pack()
tk.Label(us_portal,text='  ',bg='blue').pack()
def store_us():
    usn=lu1.get()
    pss=lu2.get()
    list_1=[usn,pss]

con=ms.connect(host='localhost',user='root',passwd='rithesh27')
    if list_1 in l_user :
        tk.Label(us_portal,text='Please select one of the options
below',bg='blue',fg='white',font=1).pack()
        tk.Label(us_portal,text='To view your electricity
consumption data click here',bg='blue',fg='white',font=1).pack()
        #consumption module
        def consume():

con=ms.connect(host='localhost',user='root',passwd='rithesh27',d
atabase='electricity')
        mycursor=con.cursor()
        b='SELECT * FROM current_consumption WHERE
ID_NUMBER='+usn
        c=pd.read_sql(b,con)
        for i in c.columns:
            print(i,':',c.loc[0,i])
        print('\n\t')

tk.Button(us_portal,text='CNSM',command=consume).pack()

```



```
tk.Label(us_portal,text='To view your bill amount and due  
date click here',bg='blue',fg='white',font=1).pack()
```

```
#amount module
```

```
def bill_am():
```

```
con=ms.connect(host='localhost',user='root',passwd='rithesh27',d  
atabase='electricity')
```

```
mycursor=con.cursor()
```

```
b='SELECT
```

```
BILL_AMOUNT,PERCENTAGE_OF_ADDITIONAL_TAXES,  
TOTAL_AMOUNT,DUE_DATE FROM electricity_bill WHERE  
ID_NUMBER='+usn
```

```
c=pd.read_sql(b,con)
```

```
for i in c.columns:
```

```
    print(i,':',c.loc[0,i])
```

```
print('\n\t')
```

```
tk.Button(us_portal,text='AMT',command=bill_am).pack()
```

```
tk.Label(us_portal,text='To generate a bill click  
here',bg='blue',fg='white',font=1).pack()
```

```
#bill module
```

```
def bill():
```

```
con=ms.connect(host='localhost',user='root',passwd='rithesh27',d  
atabase='electricity')
```

```
mycursor=con.cursor()
```

```
q1='SELECT CUSTOMER_NAME FROM  
CUSTOMER WHERE ID_NUMBER='+usn
```

```
q2='SELECT
```

```
PREVIOUS_MONTH_READING,PRESENT_MONTH_READI  
NG FROM CURRENT_CONSUMPTION WHERE  
ID_NUMBER='+usn
```

```

q3='SELECT TOTAL_AMOUNT,DUE_DATE FROM
ELECTRICITY_BILL WHERE ID_NUMBER='+usn
q1_d=pd.read_sql(q1,con)
q2_d=pd.read_sql(q2,con)
q3_d=pd.read_sql(q3,con)
print('*'*50)
print('\n\t{}\n'.format('TNPYElectricityBoard'))
print('\n\t{}\n'.format('ELECTRICITY BILL'))
print('='*50)
print('\t{}\t\t{}'.format('ID_Number',usn))

print('\t{}\t\t\t{}'.format('Name',q1_d.loc[0,'CUSTOMER_NAME']))

print('\t{}\t{}'.format('Previous
reading',str(q2_d.loc[0,'PREVIOUS_MONTH_READING'])+'
kWh'))

print('\t{}\t\t{}'.format('Current
reading',str(q2_d.loc[0,'PRESENT_MONTH_READING'])+'
kWh'))

print('='*50)
print('\t{}\t\t\t{}'.format('Bill
Amount',str(q3_d.loc[0,'TOTAL_AMOUNT'])))
print('\t{}\t{}'.format('Due date for
payment',q3_d.loc[0,'DUE_DATE']))
print('='*50)
print('\n\t{}\n'.format('Thank you, transaction was
succesful'))
print('\n\t{}\n'.format('Save Electricity for a better
future!!'))
print('*'*50)
print('\n\t')
tk.Button(us_portal,text='BILL',command=bill).pack()

```

```

        tk.Label(us_portal,text='To exit click
here',bg='blue',fg='white',font=1).pack()
        def close_us():
            us_portal.destroy()

tk.Button(us_portal,text='Logout',command=close_us).pack()
    else :
        tk.Label(us_portal,text='Incorrect username or
password.\nPlease try again',bg='blue',fg='white',font=1).pack()
        tk.Button(us_portal,text='Login',command=store_us).pack()
        tk.Label(us_portal,text='  ',bg='blue').pack()
        us_portal.mainloop()

#creating welcome portal
wel=tk.Tk()
wel.title('Welcome portal')
win_width=600
win_height=400
sc_width=wel.winfo_screenwidth()
sc_height=wel.winfo_screenheight()
center_x=int(sc_width/2-win_width/2)
center_y=int(sc_height/2-win_height/2)
wel.geometry(f'{win_width}x{win_height}+{center_x}+{center
_y}')
wel.configure(bg='blue')

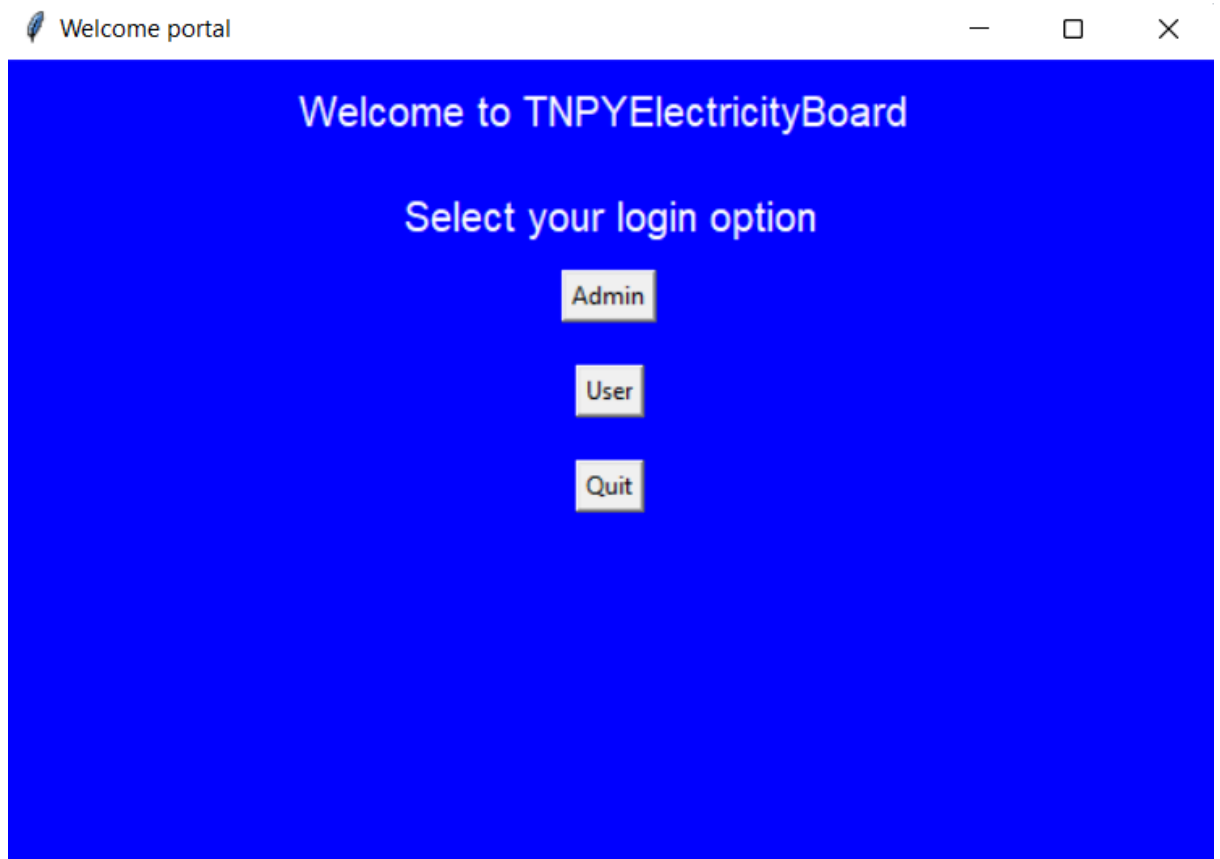
#adding widgets-welcome portal
l1=tk.Label(text='Welcome to TNPYElectricityBoard
',width=30,height=2,font=15,bg='blue',fg='white')
l1.pack()
l2=tk.Label(text='Select your login
option',width=30,height=2,font=10,bg='blue',fg='white')

```

```
l2.pack()
b1=tk.Button(text='Admin',command=adm)
b1.pack()
l3=tk.Label(text='    ',bg='blue')
l3.pack()
b2=tk.Button(text='User',command=usr)
b2.pack()
tk.Label(wel,text='    ',bg='blue').pack()
def close():
    wel.destroy()
b3=tk.Button(text='Quit',command=close)
b3.pack()
wel.mainloop()
```

Project Screenshots :-

1. Welcome Portal



2. Admin Portal

Admin portal

welcome

enter username

enter password

Login

welcome rithesh

Please select one of the options below

To view the tables click here

View

To insert new data click here

Insert

To update the data click here

Update

To remove data click here

Delete

To plot a graph of the data click here

Graph

To exit click here

Logout

2.Admin Portal :- View Module

```
Tables available
('current_consumption',)
('customer',)
('electricity_bill',)
enter table to display current_consumption
```

	ID_NUMBER	PREVIOUS_MONTH_READING	PRESENT_MONTH_READING	UNITS_CONSUMED
0	1001	1000	1500	500
1	1002	1500	2500	1000
2	1003	2000	2700	700
3	1004	2500	2850	350
4	1007	1000	1500	500
5	1008	2500	2750	250
6	1005	3000	4000	1000
7	1006	2000	3000	1000
8	1009	3000	4000	1000

```
Tables available
('current_consumption',)
('customer',)
('electricity_bill',)
enter table to display customer
```

	id_number	CUSTOMER_NAME	ADDRESS	PHONE_NUMBER
0	1001	SHIVA	ARUNDALE STREET,MYLAPORE	7905668276
1	1002	SURYA	MUTHU STREET,MYLAPORE	6383256879
2	1003	SHANKAR	SILVA ROAD	9894855667
3	1004	SUBRAMANIAM	KARPAGAM AVENUE,MYLAPORE	9940181165
4	1005	SANTHOSH	ADAM STREET,MYLPAORE	9924567895
5	1007	STELLA	SANTHOME CHURCH,SANTHOME	7805863276
6	1008	SKANDA	T.S.K ROAD,MYLAPORE	9947171175
7	1009	SATHYA	ROHINI GARDENS,MYLAPORE	9894855544
8	1006	SAM	NACHIYAPPAN STREET,MYLAPORE	9840057501

2.Admin Portal :- Insert Module

```
Tables available
('current_consumption',)
('customer',)
('electricity_bill',)
enter name of table which has to be changed customer
  id_number CUSTOMER_NAME ADDRESS PHONE_NUMBER
0      1001      SHIVA ARUNDALE STREET,MYLAPORE 7905668276
1      1002      SURYA MUTHU STREET,MYLAPORE 6383256879
2      1003      SHANKAR SILVA ROAD 9894855667
3      1004 SUBRAMANIAM KARPAGAM AVENUE,MYLAPORE 9940181165
4      1005      SANTHOSH ADAM STREET,MYLPAORE 9924567895
5      1007      STELLA SANTHOME CHURCH,SANTHOME 7805863276
6      1008      SKANDA T.S.K ROAD,MYLAPORE 9947171175
7      1009      SATHYA ROHINI GARDENS,MYLAPORE 9894855544
8      1006      SAM NACHIYAPPAN STREET,MYLAPORE 9840057501
enter id_number
1010
enter CUSTOMER_NAME
Raghav
enter ADDRESS
Blacksmith Street
enter PHONE_NUMBER
9120873547
  id_number CUSTOMER_NAME ADDRESS PHONE_NUMBER
0      1001      SHIVA ARUNDALE STREET,MYLAPORE 7905668276
1      1002      SURYA MUTHU STREET,MYLAPORE 6383256879
2      1003      SHANKAR SILVA ROAD 9894855667
3      1004 SUBRAMANIAM KARPAGAM AVENUE,MYLAPORE 9940181165
4      1005      SANTHOSH ADAM STREET,MYLPAORE 9924567895
5      1007      STELLA SANTHOME CHURCH,SANTHOME 7805863276
6      1008      SKANDA T.S.K ROAD,MYLAPORE 9947171175
7      1009      SATHYA ROHINI GARDENS,MYLAPORE 9894855544
8      1006      SAM NACHIYAPPAN STREET,MYLAPORE 9840057501
9      1010      Raghav Blacksmith Street 9120873547
```


2.Admin Portal :- Update Module

Tables available

('current_consumption',)

('customer',)

('electricity_bill',)

enter table which has to be updated current_consumption

	ID_NUMBER	PREVIOUS_MONTH_READING	PRESENT_MONTH_READING	UNITS_CONSUMED
0	1001	1000	1500	500
1	1002	1500	2500	1000
2	1003	2000	2700	700
3	1004	2500	2850	350
4	1007	1000	1500	500
5	1008	2500	2750	250
6	1005	3000	4000	1000
7	1006	2000	3000	1000
8	1009	3000	4000	1000

enter field name to be updated PRESENT_MONTH_READING

enter new data 2900

any conditions ? type yes or no

yes

enter field to apply conditions ID_NUMBER

enter condition 1006

	ID_NUMBER	PREVIOUS_MONTH_READING	PRESENT_MONTH_READING	UNITS_CONSUMED
0	1001	1000	1500	500
1	1002	1500	2500	1000
2	1003	2000	2700	700
3	1004	2500	2850	350
4	1007	1000	1500	500
5	1008	2500	2750	250
6	1005	3000	4000	1000
7	1006	2000	2900	1000
8	1009	3000	4000	1000

2.Admin Portal :- Delete Module

```
Tables available
('current_consumption',)
('customer',)
('electricity_bill',)
enter table from which data is to be deleted electricity_bill
  id_number  UNITS_CONSUMED  ...  TOTAL_AMOUNT  DUE_DATE
0      1001           500  ...      4950  2022-01-14
1      1002          1000  ...      9900  2022-01-14
2      1003           200  ...      1980  2022-01-17
3      1004           350  ...      3465  2022-01-17
4      1005          1000  ...      9900  2022-01-21
5      1006          1000  ...      9900  2022-01-17
6      1007           500  ...      4950  2022-01-25
7      1008           250  ...      6000  2022-02-04
8      1009          1000  ...      9900  2022-02-03

[9 rows x 7 columns]
enter field in which condition is applied ID_NUMBER
enter condition 1008
  id_number  UNITS_CONSUMED  ...  TOTAL_AMOUNT  DUE_DATE
0      1001           500  ...      4950  2022-01-14
1      1002          1000  ...      9900  2022-01-14
2      1003           200  ...      1980  2022-01-17
3      1004           350  ...      3465  2022-01-17
4      1005          1000  ...      9900  2022-01-21
5      1006          1000  ...      9900  2022-01-17
6      1007           500  ...      4950  2022-01-25
7      1009          1000  ...      9900  2022-02-03

[8 rows x 7 columns]
```

2. Admin Portal :- Graphs

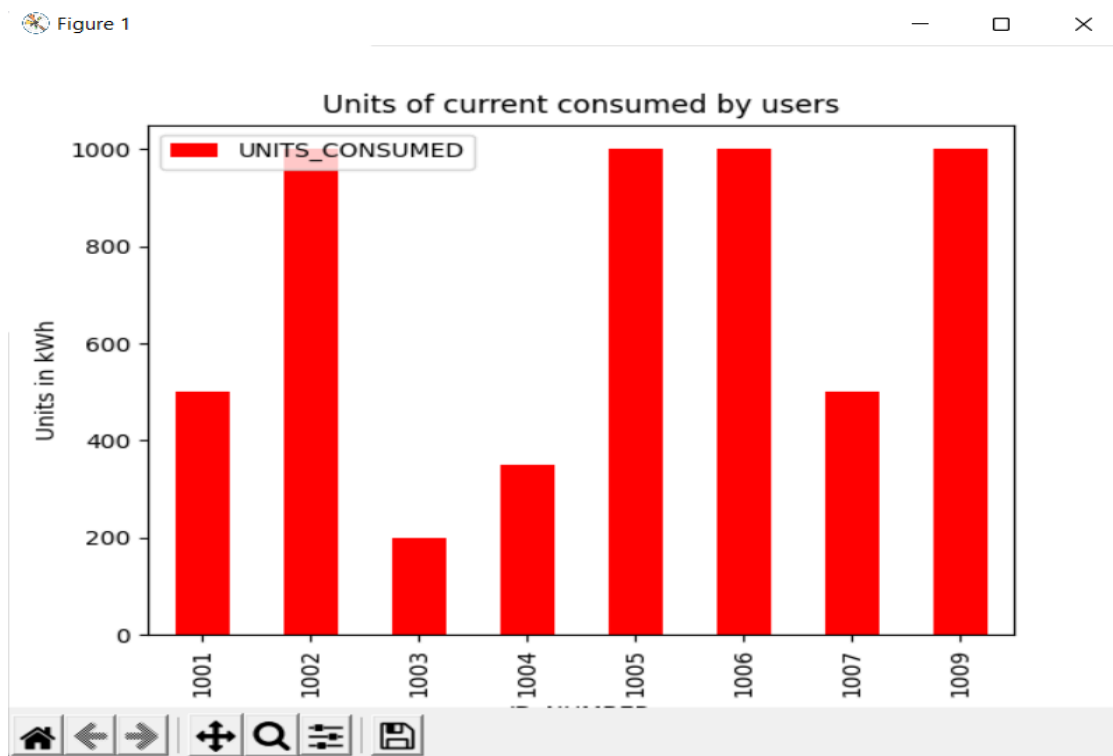
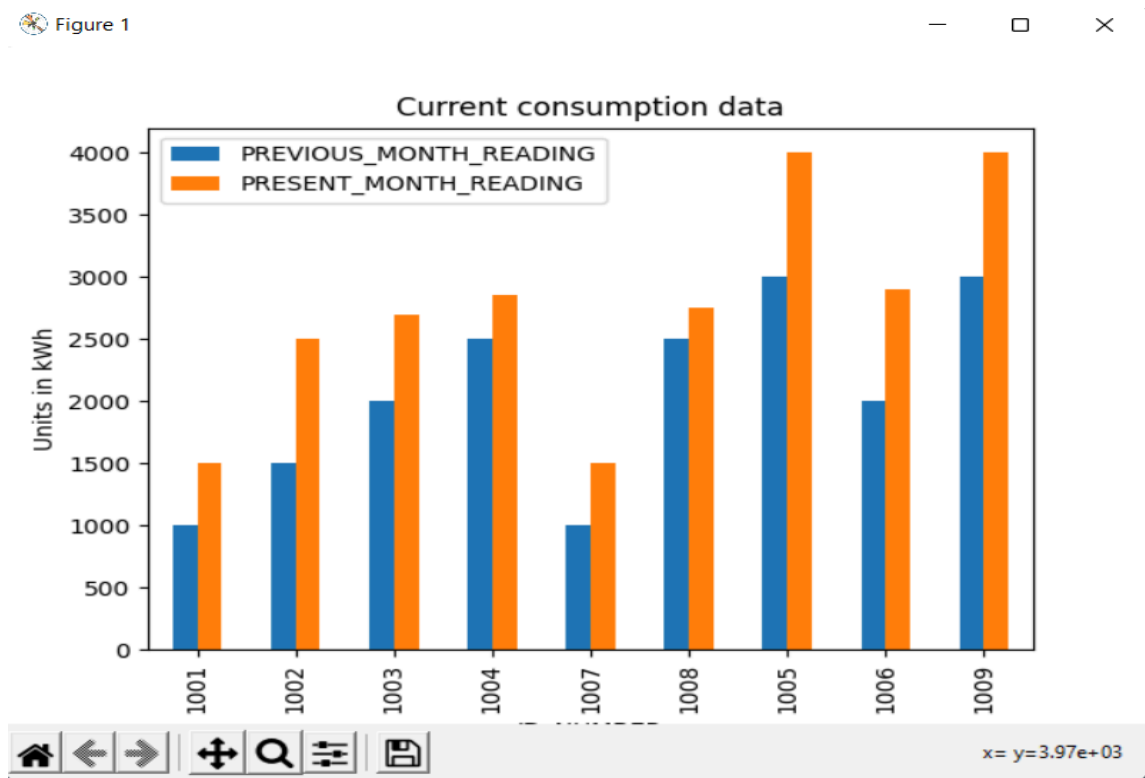
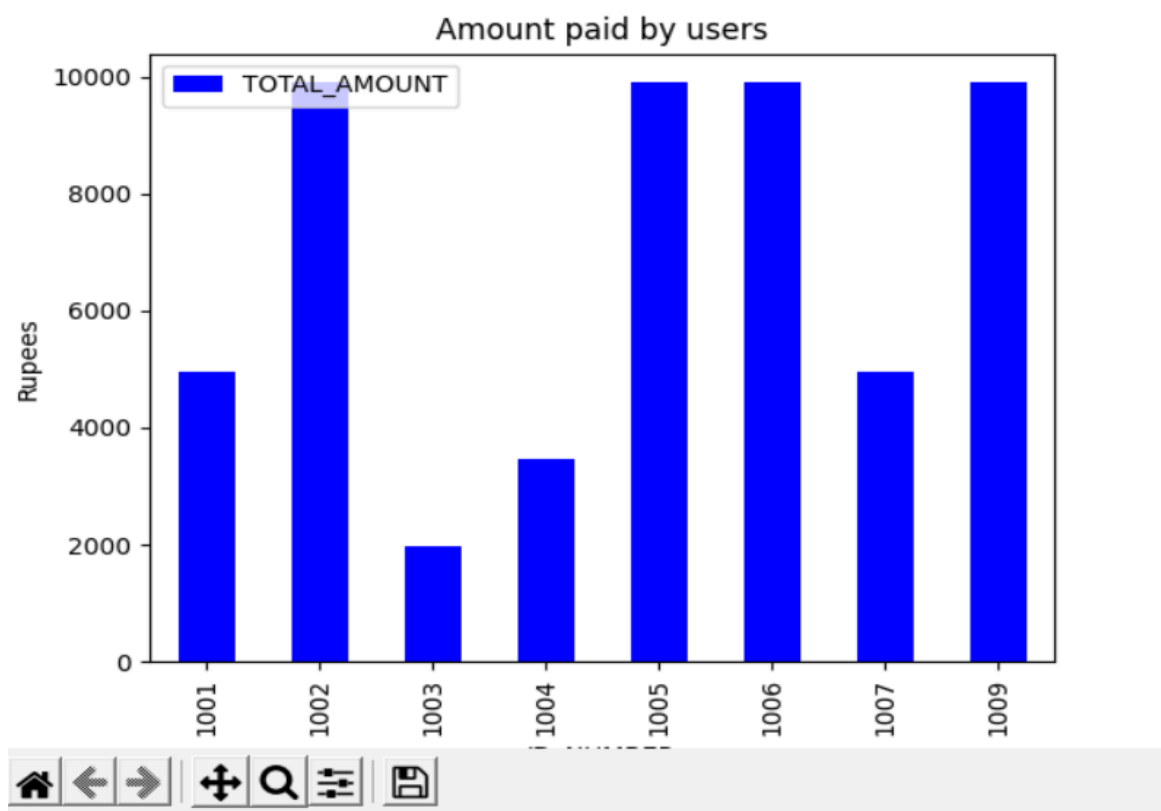


Figure 1



3. User Portal

 User portal—□×

Welcome

enter username

1002

enter password

Login

Please select one of the options below

To view your electricity consumption data click here

CNSM

To view your bill amount and due date click here

AMT

To generate a bill click here

BILL

To exit click here

Logout

3.User Portal :- Consumption Data

```
ID_NUMBER : 1002  
PREVIOUS_MONTH_READING : 1500  
PRESENT_MONTH_READING : 2500  
UNITS_CONSUMED : 1000
```

3.User Portal :- Electricity bill Data

```
BILL_AMOUNT : 9000  
PERCENTAGE_OF_ADDITIONAL_TAXES : 10.0  
TOTAL_AMOUNT : 9900  
DUE_DATE : 2022-01-14
```

3.User Portal :- Generating Bill

```
*****
TNPYElectricityBoard

ELECTRICITY BILL

=====
ID_Number          1002
Name               SURYA
Previous reading    1500 kWh
Current reading     2500 kWh
=====
Bill Amount         ₹9900
Due date for payment 2022-01-14
=====

Thank you, transaction was succesful

Save Electricity for a better future!!

*****
```

Limitations :-

- The Program does not provide any ways for transactions as it is not connected to any payment APIs
- Datasets containing reliable information on electricity billing could not be found , hence data had to be created .
- Only simple methods of calculation of electricity bills were used for the project using units consumed and rate per unit

References :-

- Class 12 IP Textbook - NCERT
- www.Stackoverflow.com
- www.pythontutorial.net
- www.tutorialspoint.com

Conclusion :-

The project titled Electricity Billing and Management System done by S K Rithesh Akash for the academic year 2021-2022 , has been completed and compiled , tested and executed successfully