```
#code for console
#Group 3.15
#DBMS Title: Municipal Corporation


import psycopg2
from texttable import Texttable

try:
   connection=psycopg2.connect(
                     host="10.100.71.21",
                     port="5432",
                     user="201701198",
                     password="201701198",
                     database="201701198"
                     )
except psycopg2.OperationalError as e:
   print('Unable to connect!\n{0}').format(e)
   exit(1)
else:
#connect to db
   print("You are connected to 201701198")


#Cursor
cursor=connection.cursor()

#query execute
cursor.execute("SET SEARCH_PATH TO Municipal_Corporation")
#cursor.execute("select * from team")
#if you are make any changes in data base then you have to do commit following:
#connection.commit()
#take data from query
#rows=cursor.fetchall()
#for r in rows:
 #   print(f"Team_Name: {r[0]} | Manager: {r[1]} | Country: {r[2]}")

def q1():
   q = """SET SEARCH_PATH TO Municipal_Corporation; select citizen.f_name, citizen.house_no, citizen.
society from citizen inner join
(select peoplehavecomplains.citizenid
from peoplehavecomplains
inner join
(select complaintid
from complains
where serviceid='1'
and cstatus='unsolved') as r1
on r1.complaintid=peoplehavecomplains.complaintid) as r2
on r2.citizenid=citizen.citizenid
where citizen.pincode='398965';"""
   cursor.execute(q)
   rows=cursor.fetchall()

   x=[]
   x.append(['f_name','house_no.','society'])
```

```
    x.extend(rows)

    t = Texttable()
    t.add_rows(x,header=True)
    print(t.draw())
    return

def q2():
    q = """SET SEARCH_PATH TO Municipal_Corporation;select contactpersonname, contactno
from company
inner join ( select r3.company_name
from (select r2.phaseno,r2.company_name
from (phase natural join construction) as r2
where r2.phasename='Halted') as r3) as r4
on company.companyname=r4.company_name;"""
    cursor.execute(q)
    rows=cursor.fetchall()
    x=[]
    x.append(['person name', 'Cont. NO.'])

    for r in rows:
        x.append(r)
    t = Texttable()
    t.add_rows(x,header=True)
    print(t.draw())
    return

def q3():
    q = """SET SEARCH_PATH TO Municipal_Corporation;select r1.pincode, SUM(r1.cost) as s from
(select p.pincode, c.constructionid, c.cost
from publicproperty as p
inner join costonconstruction as c
on p.established=c.constructionid) as r1
group by r1.pincode
order by s desc;"""
    cursor.execute(q)
    rows=cursor.fetchall()

    x=[]
    x.append(['PINCODE', 'Total_cost'])
    for r in rows:
        x.append(r)
    t = Texttable()
    t.add_rows(x,header=True)
    print(t.draw())
    return

def q4():
    q = """SET SEARCH_PATH TO Municipal_Corporation;select d.dname,count(r1.complaintid)
from department as d
inner join (select s.departmentid,c.cstatus,c.complaintid
from services as s
inner join complains as c
on s.serviceid=c.serviceid
where cstatus='unsolved' or cstatus='progress') as r1 on r1.departmentid=d.d_id
```

```
group by d.dname;"""
   cursor.execute(q)
   rows=cursor.fetchall()
   x=[]
   x.append(['D_Name', 'No. of Complaints'])

   for r in rows:
      x.append(r)
   t = Texttable()
   t.add_rows(x,header=True)
   print(t.draw())
   return

def q5():
   q = """SET SEARCH_PATH TO Municipal_Corporation;select c.citizenid,c.f_name, c.income, SUM(r2.a
mount)
from citizen as c
inner join (select r1.owner, r1.amount
from employee as e
inner join (select pr.owner,p.propertyid,p.amount
from penalty as p
inner join property as pr
on pr.proid=p.propertyid) as r1
on r1.owner=e.citizenid ) as r2
on c.citizenid=r2.owner
group by c.f_name, c.citizenid, c.income;"""
   cursor.execute(q)
   rows=cursor.fetchall()
   x=[]
   x.append(['ID', 'Name','Income','total penalty'])

   for r in rows:
      x.append(r)
   t = Texttable()
   t.add_rows(x,header=True)
   print(t.draw())
   return

def q6():
   q = """SET SEARCH_PATH TO Municipal_Corporation;select c.citizenid,c.f_name, c.income, count(r1.a
mount) as No_of_penalties,
SUM(r1.amount) as Total_Amount
from citizen as c
inner join (select pr.owner,p.propertyid,p.amount
from penalty as p
inner join property as pr
on pr.proid=p.propertyid) as r1
on c.citizenid=r1.owner
group by c.f_name, c.citizenid, c.income
order by No_of_penalties desc;"""
   cursor.execute(q)
   rows=cursor.fetchall()
   x=[]
   x.append(['ID','Name','Income','No. of Penalty','Total penalty'])
```

```
    for r in rows:
        x.append(r)
    t = Texttable()
    t.add_rows(x,header=True)
    print(t.draw())
    return



def command(choice):
    if choice=='1':
        q1()
    elif choice=='2':
        q2()
    elif choice=='3':
        q3()
    elif choice=='4':
        q4()
    elif choice=='5':
        q5()
    elif choice=='6':
        q6()
    else:
        print("Invalide choice")
    return




while True:
    # Queries
    print("1. Extract the Name, house number and society of the citizen who has a complaint regarding
water management in area having pincode 398965.")
    print("2. extract the name and conta t no of the company-person whose construction is halted.")
    print("3. Extract the area with the money spent on public propert and arrange them in descending o
rder")
    print("4. Which department has complaints left unsolved or in progress? Also count the number of c
omplaints")
    print("5. Find out which employees have penalties? Also show their Citizen ID and Income.")
    print("6. Find out which citizens have penalties? Also show their Citizen ID and Income, number of p
enalties and Total_Amount.")

    print("--->Type 'q' for end session")

    choice=input("Enter Your Choice:")
    if choice=='q':
        break
    command(choice)
if(connection):
    #close the cursor
    cursor.close()
    #Close connection
    connection.close()
    print("Connection closed to 201701198")
```