**FP5.0 Module-1 Assignment-3**

**Batch Name:**

Infosys FP5.0 Summer 2018

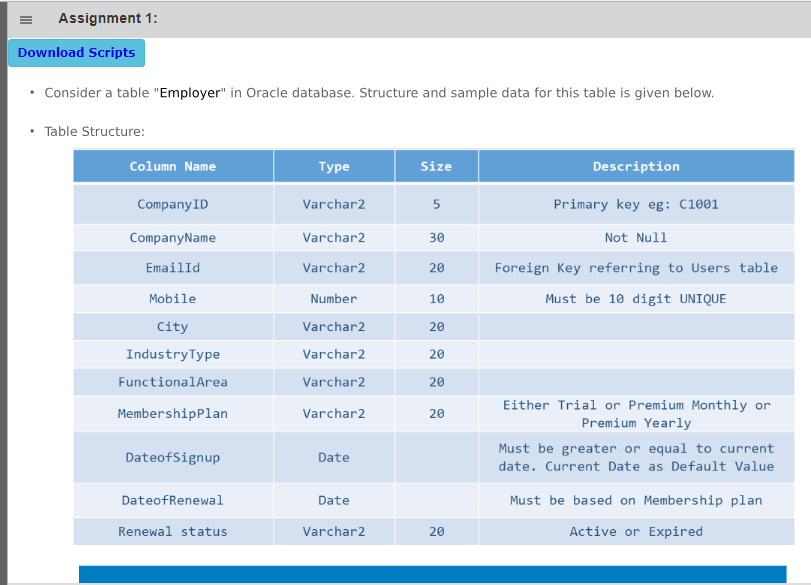
**Enrollment Number:** R171217041

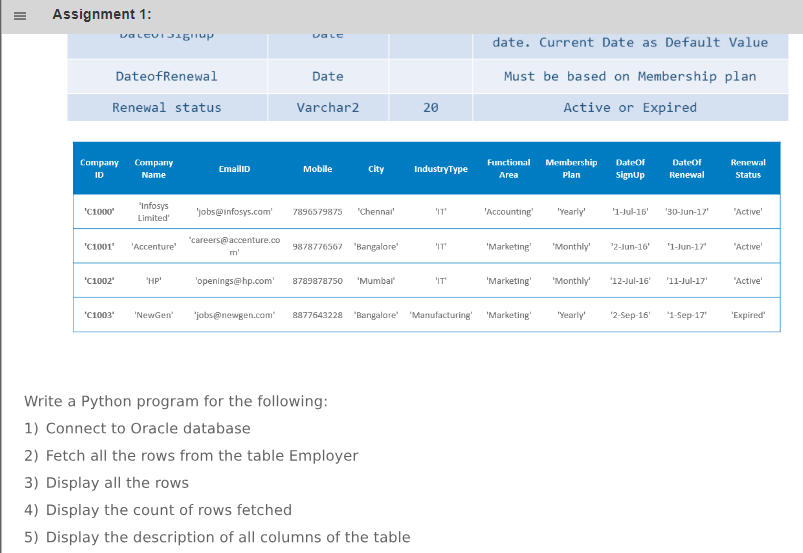
**SAPID:** 500060720

**Name:** Nishkarsh Raj Khare

**Sem:** Semester-III

**Branch:** CSE-DevOps-Xebia

**Assignment-1 QUestion**

****

**Assignment-1 code**

'''1) Connect to Oracle Database '''

import cx\_Oracle

connection\_object = cx\_Oracle.connect("system/user123@XE") #Connection IP

cursor\_object = connection\_object.cursor()

'''2) Fetch all rows from Table Employee '''

cursor\_object.execute("select \* from Employer")

'''3) Display all the rows '''

i=1

for row in cursor\_object:

print("Row number",i,"is:",row)

i = i+1

'''4) Display count of rows fetched '''

cursor\_object.fetchall() #Without fetchall() the rowcount attribute does not works

print("The number of rows fetched are:",cursor\_object.rowcount)

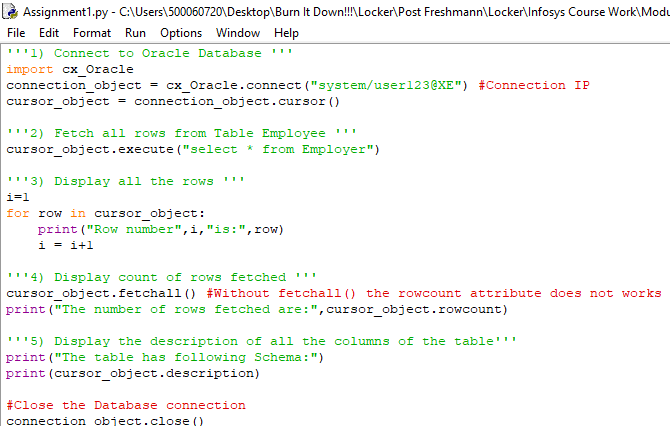
'''5) Display the description of all the columns of the table'''

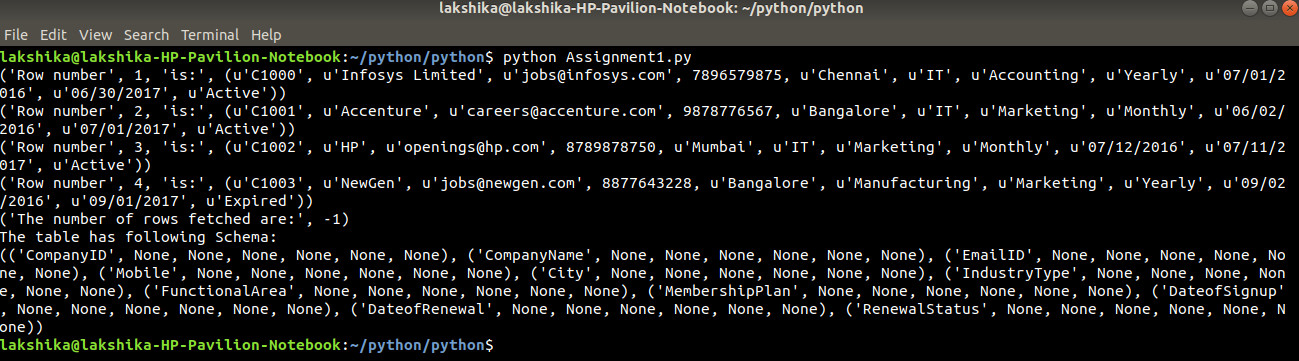
print("The table has following Schema:")

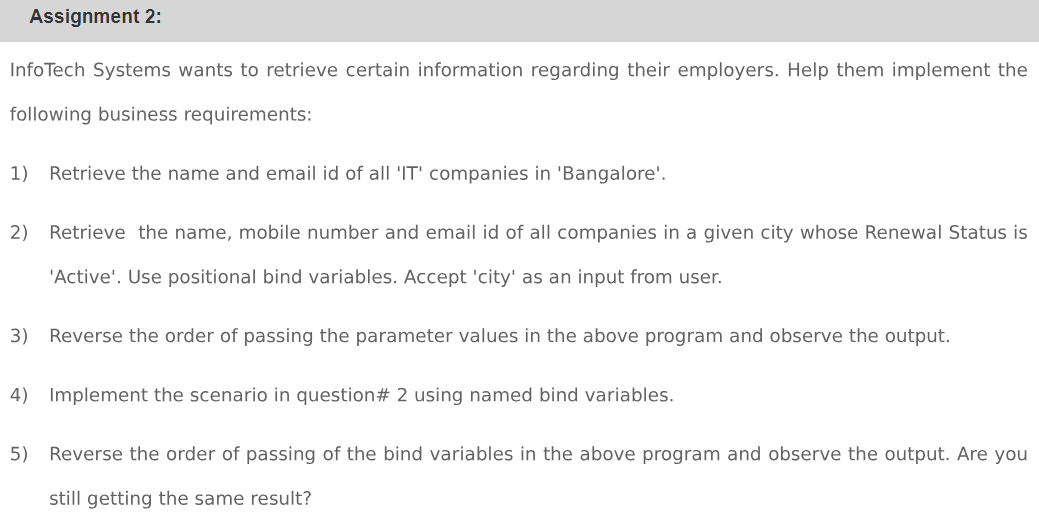
print(cursor\_object.description)

#Close the Database connection

connection\_object.close()

**Assignment-1 Code Screenshot**

**Assignment-1 Output Screenshot**

**Assignment-2 QUestion**

**Assignment-2 code**

print("InfoTech Systems Query")

import cx\_Oracle

connection\_object = cx\_Oracle.connect("system/user123@XE") #Connection IP

cursor\_object = connection\_object.cursor()

'''1) Retrieve the name and email id of all the "IT" Companies in "Bangalore" '''

cursor\_object.execute("select ComapnyName,EmailID from Employer where IndustryType='IT' and City='Bangalore'")

print(cursor\_object.fetchall())

'''2) Retrieve the name,mobile number and email id of all the companies in a given city whose Renewal Status is 'Active'. Use positional bind variables. Accept 'city' as an input from the user '''

city = input("Enter the name of the city whose details you want!")

cursor\_object.execute("select CompanyName,,EmailID from Employer where City=:1 and RenewalStatus=:2",(city,'Active'))

print("Results shown using positional bind variables in order are:")

print(cursor\_object.fetchall())

'''3) Reverse the order of passing the parameters values in above program and observe the output'''

cursor\_object.execute("select CompanyName,,EmailId from Employer where City=:1 and RenewalStatus=:2",('Active',city))

print("Results shown using positional bind variables in reverse order are:")

print(cursor\_object.fetchall())

'''4) Implement the scenario in question #2 using named bind variables '''

cursor\_object.execute("select CompanyName,,EmailID from Employer where City=:city and RenewalStatus=:renew",{"city":city,"renew":"Active"})

print("Results shown using Named bind variables in order are:")

print(cursor\_object.fetchall())

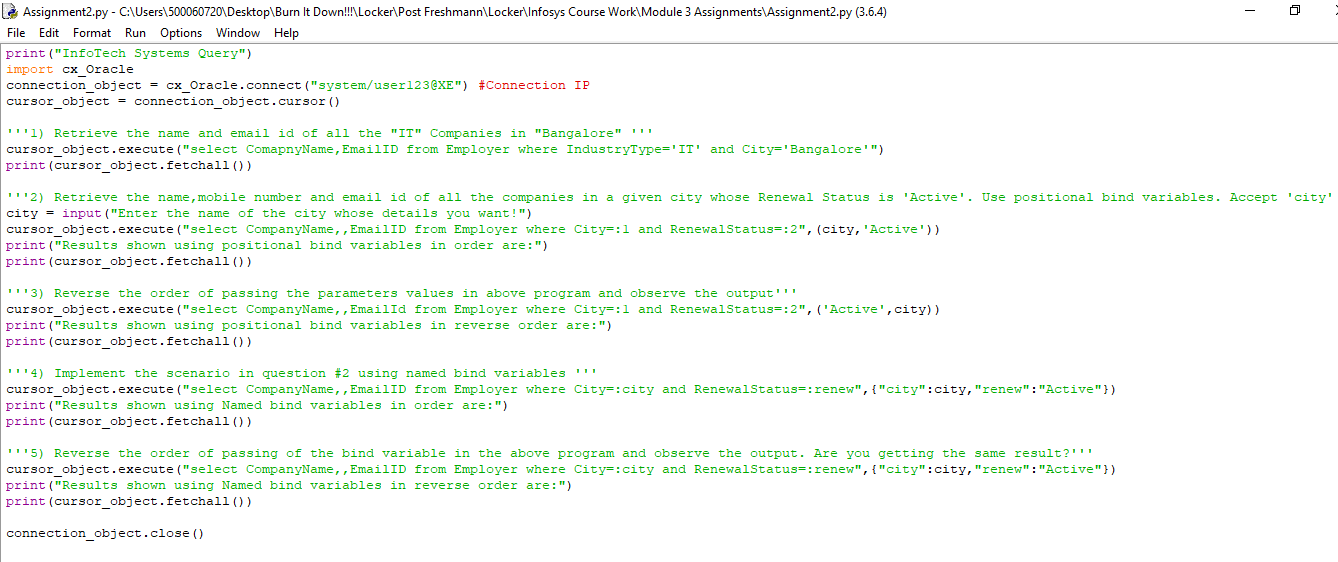
'''5) Reverse the order of passing of the bind variable in the above program and observe the output. Are you getting the same result?'''

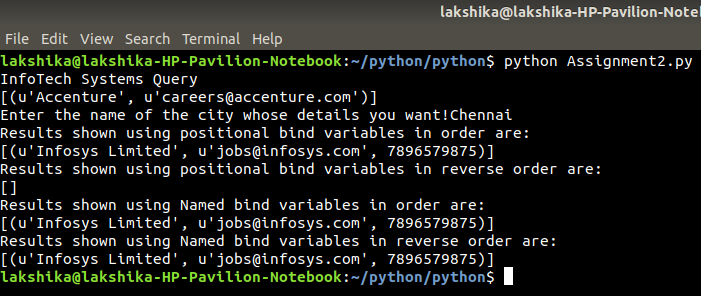
cursor\_object.execute("select CompanyName,,EmailID from Employer where City=:city and RenewalStatus=:renew",{"city":city,"renew":"Active"})

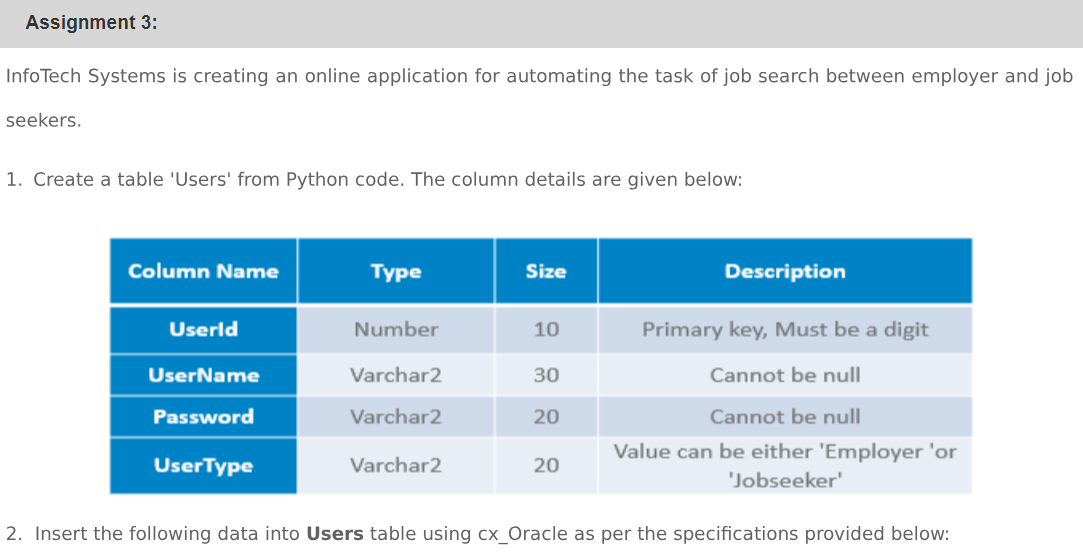
print("Results shown using Named bind variables in reverse order are:")

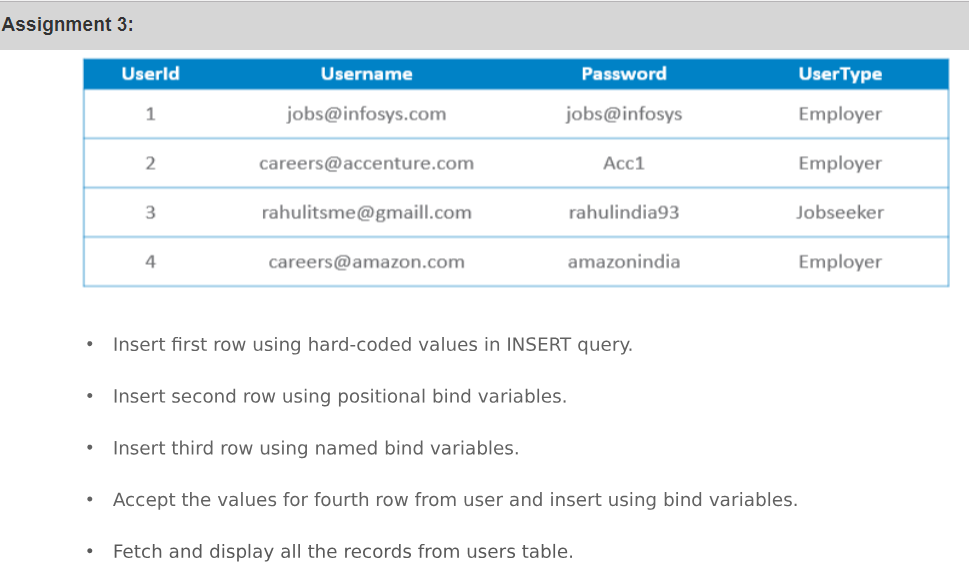
print(cursor\_object.fetchall())

connection\_object.close()

**Assignment-2 Code Screenshot**

**Assignment-2 Output Screenshot**

**Assignment-3 QUestion**

****

**Assignment-3 code**

print("InfoTech Systems Users")

import cx\_Oracle

con = cx\_Oracle.connect("system/user123@XE") #Connection IP

cur = con.cursor()

'''1) Create a table "Users" from Python code. '''

statement='''create table Users(

UserId number(10) primary key,

UserName varchar(30) not null,

Password varchar(20) not null,

UserType varchar(20) check (UserType in ('Employer','Jobseeker')))'''

cur.execute(statement)

'''2) Insert data into Users Table using cx\_Oracle '''

#2.1) Insert first row using hard-coded values in Insert query

cur.execute("insert into Users values (1,'jobs@infosys.com','jobs@infosys','Employer')")

con.commit()

#2.2) Insert second row using positional bind variables

cur.execute("insert into users values (:v1,:v2,:v3,:v4)",(2,'careers@accenture.com','Acc1','Employer'))

con.commit()

#2.3) Insert third row using named bind variables

cur.execute("insert into users values (:v1,:v2,:v3,:v4)",{"v1":3,"v2":'rahulitsme@gmail.com',"v3":'rahulindia93',"v4":'Jobseeker'})

con.commit()

#2.4) Accept the values for fourth row from user and insert using bind values

print("Enter Details to be inserted in the database")

userid = input("UserID: ") #Enter 4

username = input("UserName: ") #Enter careers@amazon.com

password = input("Password: ") #Enter amazonindia

usertype = input("UserType: ") #Enter Employer

cur.execute("insert into users values (:v1,:v2,:v3,:v4)",{"v1":userid,"v2":username,"v3":password,"v4":usertype})

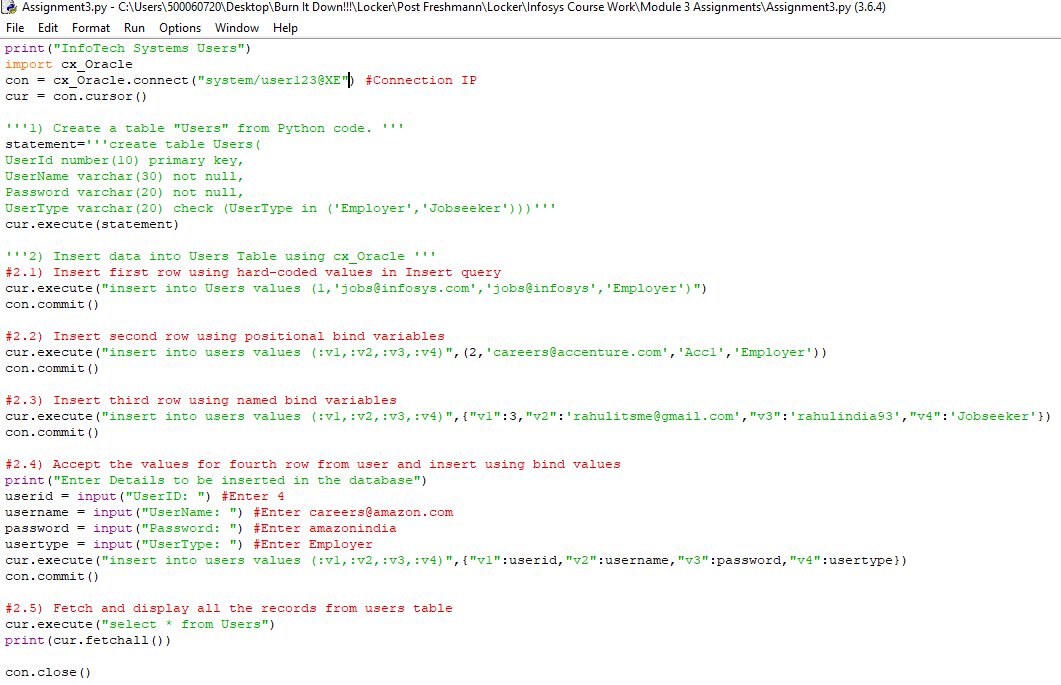
con.commit()

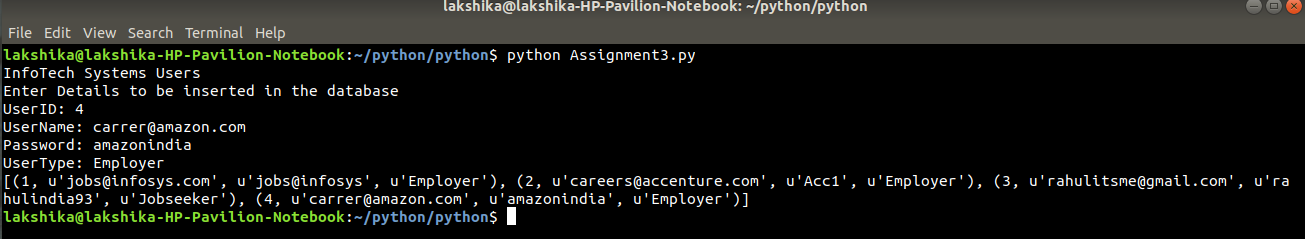
#2.5) Fetch and display all the records from users table

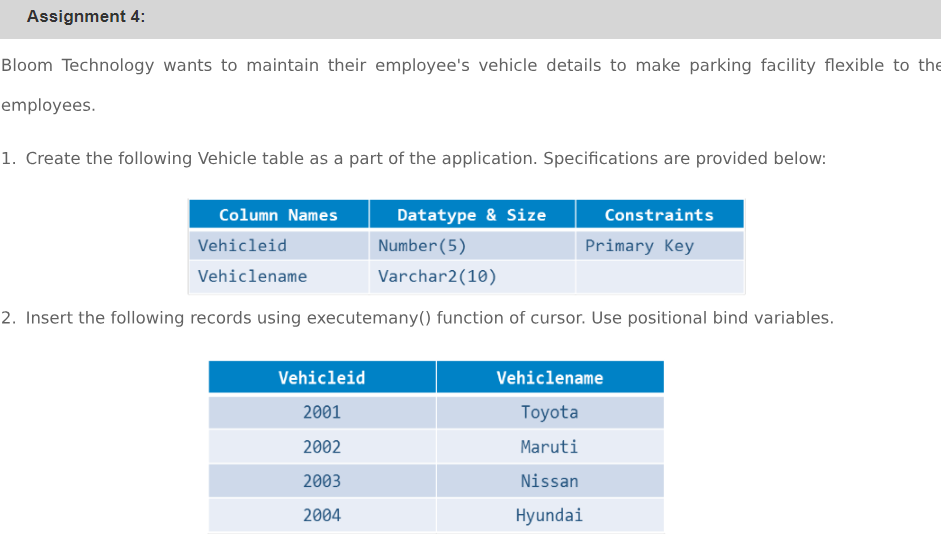
cur.execute("select \* from Users")

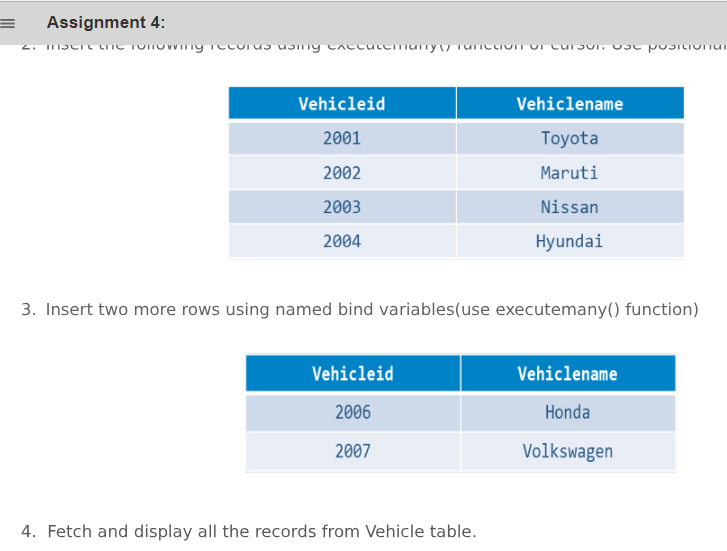
print(cur.fetchall())

con.close()

**Assignment-3 Code Screenshot**

**Assignment-3 Output Screenshot**

**Assignment-4 Question**

****

**Assignment-4 code**

print("Bloom Technology Parking Logs")

import cx\_Oracle

con = cx\_Oracle.connect("system/user123@XE") #ConnectionIP

cur = con.cursor()

'''1) Create the following Vehicle Table as part of the application.'''

statements = '''create table Vehicle(

Vehicleid number(5) primary key,

Vehiclename varchar(10))'''

cur.execute(statements)

con.commit()

'''2) Insert the following records using executemany() function'''

cur.executemany("insert into Vehicle values (:v1,:v2)",

[(2001,'Toyota'),(2002,'Maruti'),(2003,'Nissan'),(2004,'Hyundai')])

con.commit()

'''3) Insert two more rows using named bind variables using executemany()'''

cur.executemany("insert into Vehicle values (v1:,v2:)",

[{"v1":2006,"v2":'Honda'},

{"v1":2007,"v2":'Volkswagen'}])

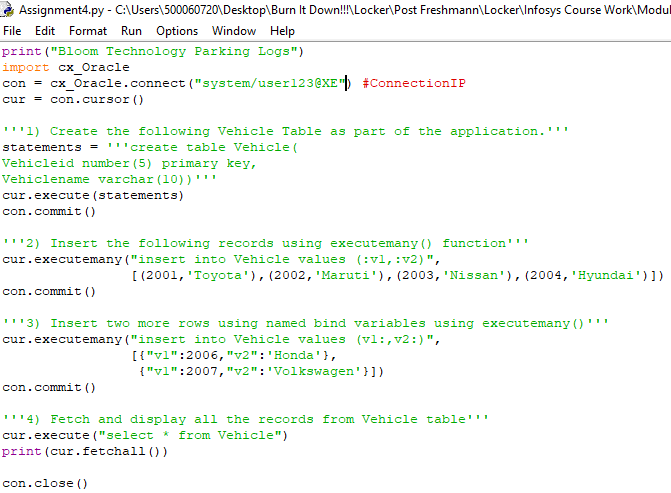
con.commit()

'''4) Fetch and display all the records from Vehicle table'''

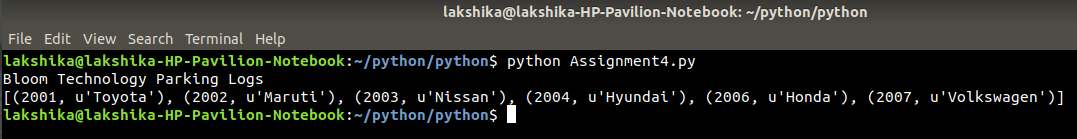
cur.execute("select \* from Vehicle")

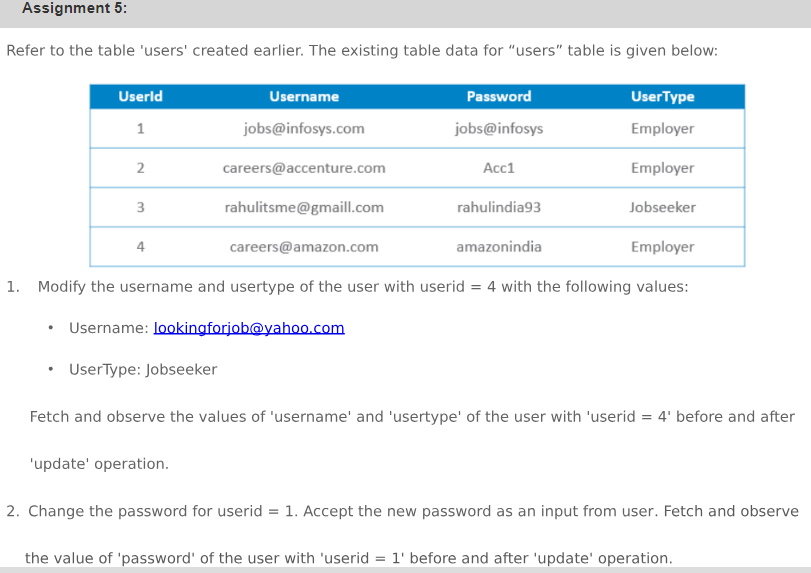
print(cur.fetchall())

con.close()

**Assignment-4 Code Screenshot**

**Assignment-4 Output Screenshot**

****

**Assignment-5 Question**

**Assignment-5 code**

print("InfoTech Systems Users")

import cx\_Oracle

con = cx\_Oracle.connect("system/user123@XE") #Connection IP

cur = con.cursor()

def show(cursor): #function to show the current contents of the database

cursor.execute("select \* from Users")

i = 1

for row in cursor:

print("Content of row ",i,"is:",row)

i = i+1

'''1) Modify the UserName and UserType with UserId=4 such as:

UserName: lookingforjob@yahoo.com

UserType: Jobseeker '''

print("Content of the Table 'Users' before updating values are")

show(cur)

cur.update("update Users set UserName = 'lookingforjob@yahool.com',UserType = Jobseeker where UserId = 4")

print("Content of the Table 'Users' after updating values are")

show(cur)

'''2) Change the password for userid=1. Accept new password from the user during runtime. Fetch and show password for userid = 1 before and after execution of program'''

print("Content of the Table 'Users' before updating values are")

show(cur)

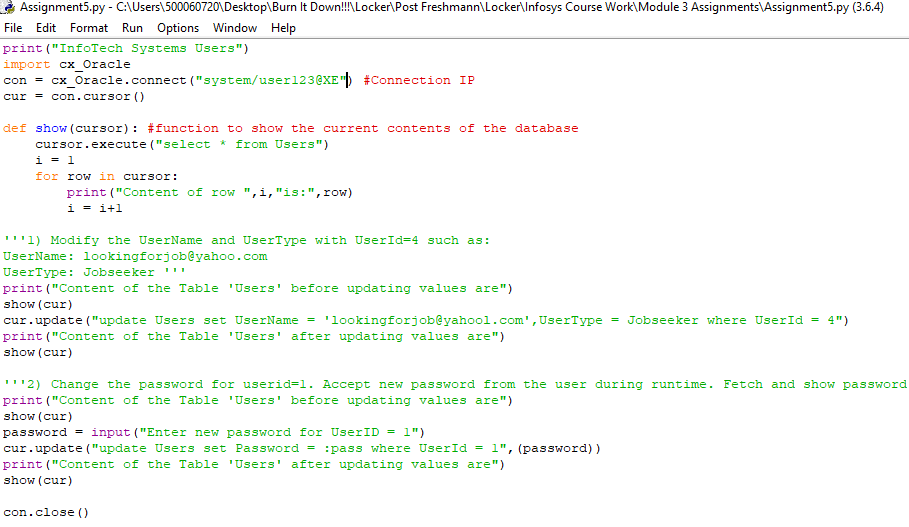
password = input("Enter new password for UserID = 1")

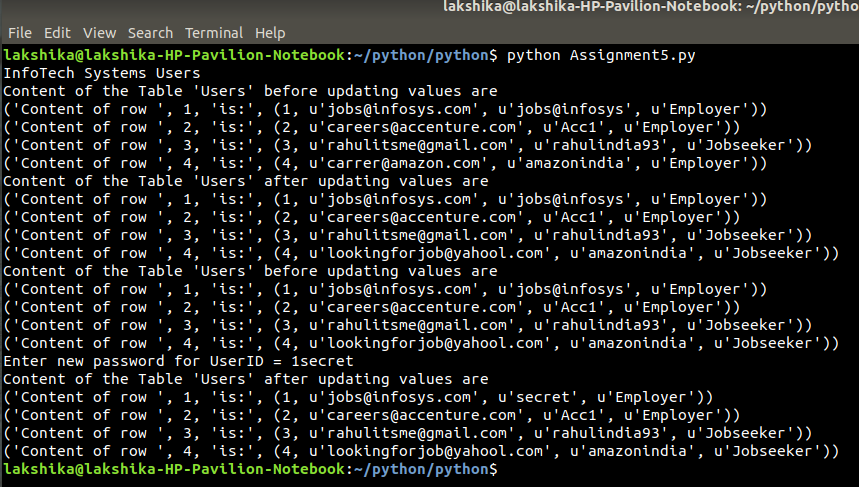
cur.update("update Users set Password = :pass where UserId = 1",(password))

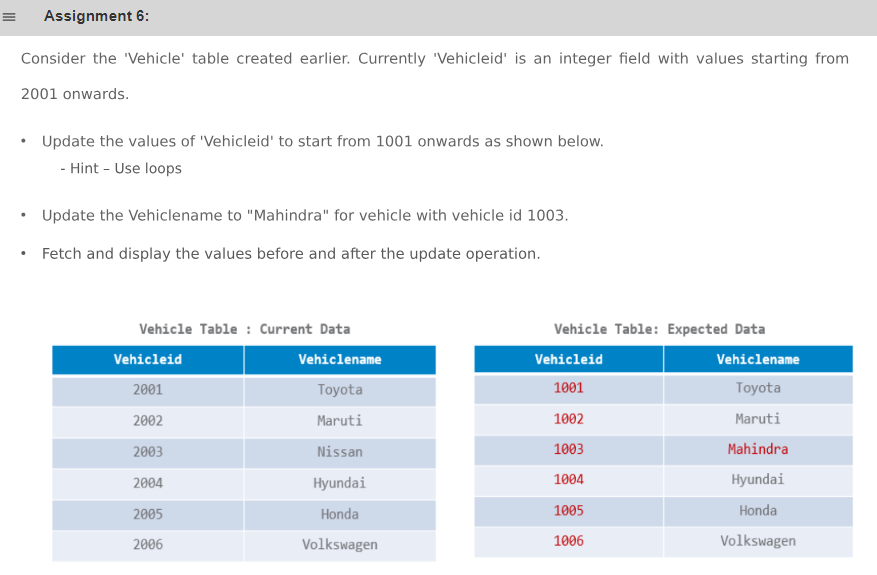
print("Content of the Table 'Users' after updating values are")

show(cur)

con.close()

**Assignment-5 Code Screenshot**

**Assignment-5 Output Screenshot**

**Assignment-6 QUestion**

**Assignment-6 code**

**import cx\_Oracle**

**con = cx\_Oracle.connect("system/user123@XE") #ConnectionIP**

**cur = con.cursor()**

**#Function for showing current state of Table Vehicle**

**def show(cursor):**

**cursor.execute("select \* from Vehicle")**

**i = 1**

**for row in cursor:**

**print("Content of row number",i,"is:",row)**

**i = i+1**

**'''Update Vehicle Id ranging from 2001-2007 to 1001-1007'''**

**print("State of table 'Vehilce' before updating Vehicle ID is")**

**show(cur)**

**for k in range (2001,2008):**

**if k!=2005:**

**cur.execute("update Vehicle set Vehicleid = :1 where Vehicleid = :2",(k-1000,k))**

**else:**

**continue**

**con.commit()**

**print("State of table 'Vehicle' after updating Vehicle ID is")**

**show(cur)**

**'''Update Vehicle name for VehicleID = 1003 to Mahindra '''**

**print("State of Table Vehicle before updating Vehcile ID 1003")**

**show(cur)**

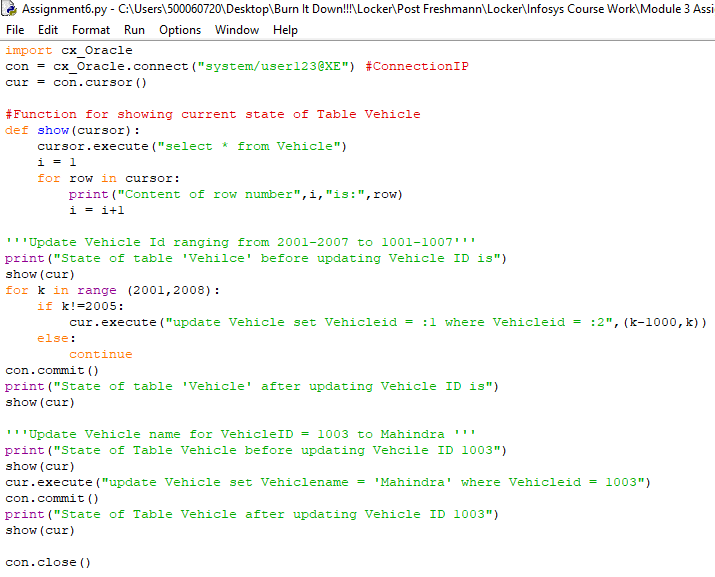
**cur.execute("update Vehicle set Vehiclename = 'Mahindra' where Vehicleid = 1003")**

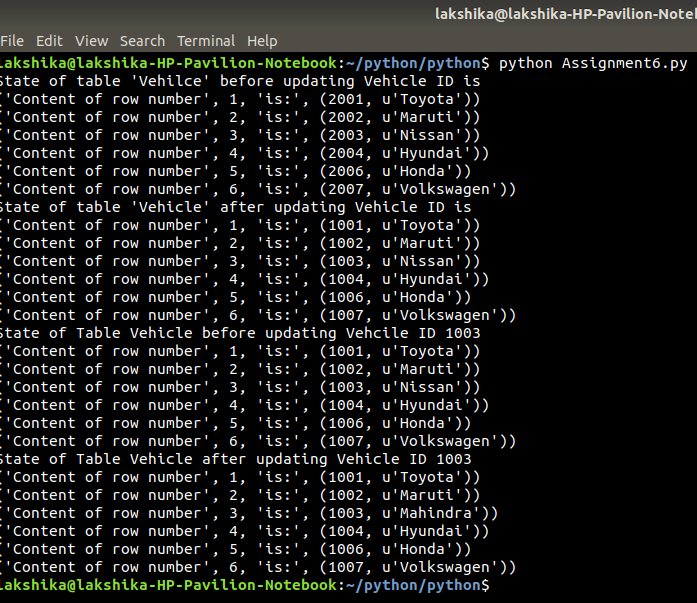
**con.commit()**

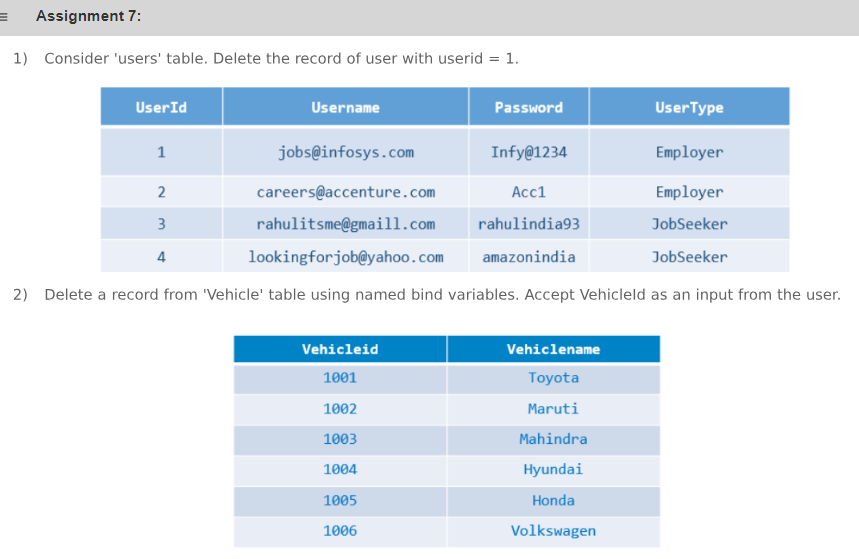
**print("State of Table Vehicle after updating Vehicle ID 1003")**

**show(cur)**

**con.close()**

**Assignment-6 Code Screenshot**

**Assignment-6 Output Screenshot**

**Assignment-7 Question**

**Assignment-7 code**

import cx\_Oracle

con = cx\_Oracle.connect("system/user123@XE") #Same connection can be used to access anytable from the current database

cur = con.cursor()

'''1) Delete record of userid = 1 from Users Table '''

cur.execute("delete from Users where UserId = 1")

cur.execute("select \* from Users")

print("The new content of table Users is:")

print(cur.fetchall())

'''2) Delete a record from 'Vehicle' table using named bind variables with Vehicleid accepted from user '''

vid = int(input("Enter VehicleId between 1001 to 1007 for Deletion"))

if vid in (1001,1002,1003,1004,1006,1007):

cur.execute("delete from Vehicle where Vehicleid=:v1",{"v1": vid})

else:

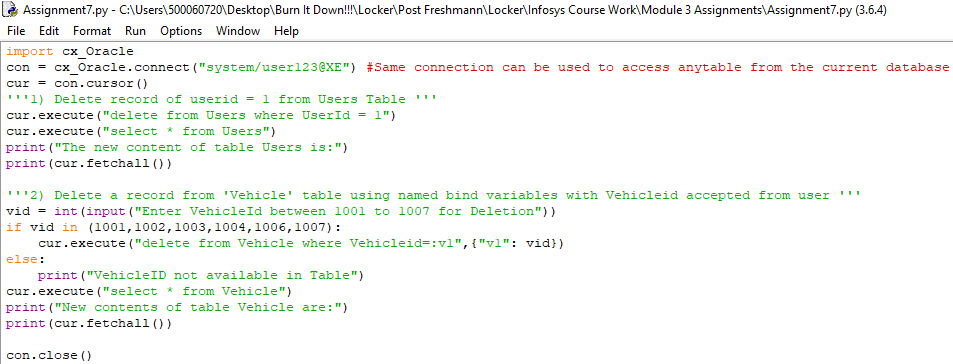
print("VehicleID not available in Table")

cur.execute("select \* from Vehicle")

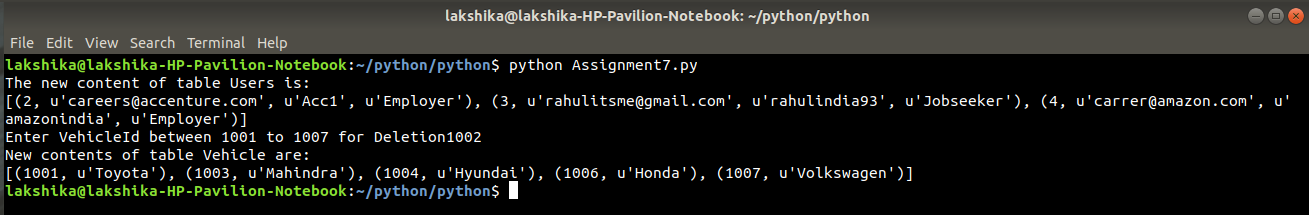
print("New contents of table Vehicle are:")

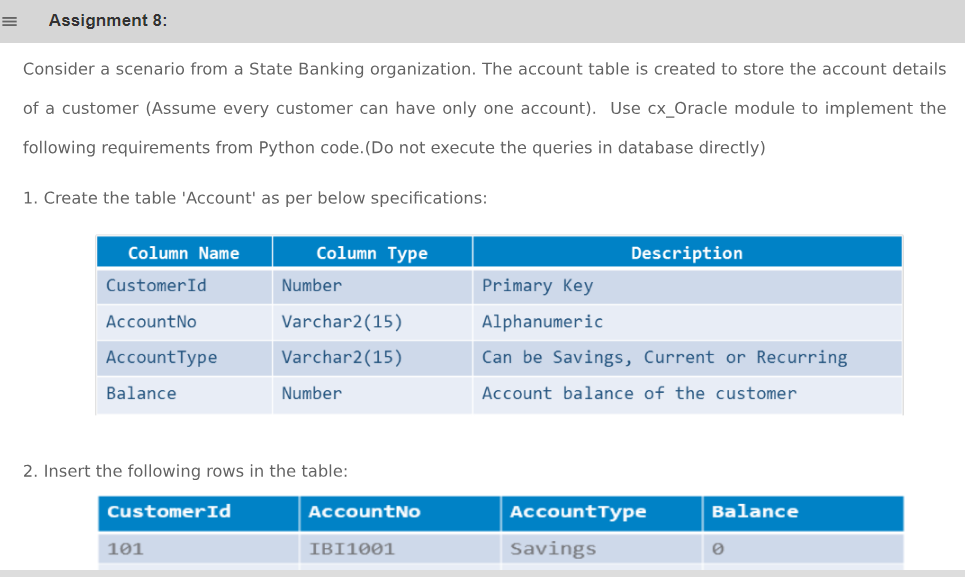
print(cur.fetchall())

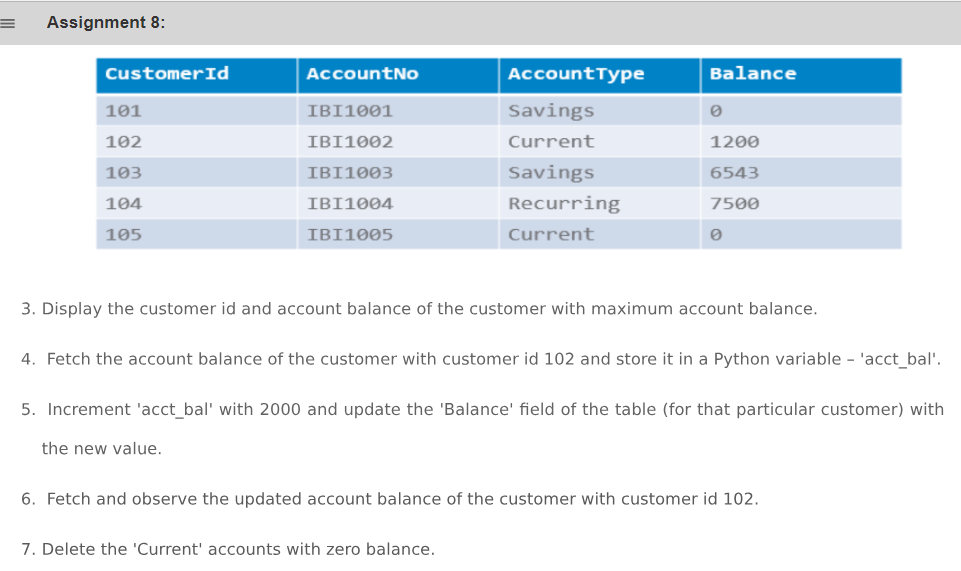
con.close()

**Assignment-7 Code Screenshot**

**Assignment-7 Output Screenshot**

****

**Assignment-8 QUestion**

****

**Assignment-8 code**

print("State Banking Organization")

import cx\_Oracle

con = cx\_Oracle.connect("system/user123@XE") #ConnectionIP

cur = con.cursor()

'''1) Create the table 'Account' as per below specifications'''

cur.execute('''create table Account(

CustomerId number(5) primary key,

AccountNo varchar(15),

AccountType varchar(15) check (AccountType in ('Savings','Current','Recurring')),

Balance number(7)

)''')

con.commit()

'''2) Insertion of the Data'''

cur.executemany("insert into Account values (:v1,:v2,:v3,:v4)",

[{"v1":101,"v2":'IBI1001',"v3":'Savings',"v4":0},

{"v1":102,"v2":'IBI1002',"v3":'Current',"v4":1200},

{"v1":103,"v2":'IBI1003',"v3":'Savings',"v4":6543},

{"v1":104,"v2":'IBI1004',"v3":'Recurring',"v4":7500},

{"v1":105,"v2":'IBI1005',"v3":'Current',"v4":0}

])

con.commit()

'''3) Display customer ID and Account balance of the customer with maximum balance'''

cur.execute("select CustomerId,Balance from Account where Balance=(select max(Balance) from Account)")

print("Details of account with Maximum Balance are:",cur.fetchall())

'''4) Fetch the account balance of the customer with customer id 102 and store it in python variable 'acct\_bal' '''

cur.execute("select Balance from Account where CustomerId = 102")

acct\_balance = cur.fetchall()

print("Account Balance of the Person with Id =102 is:",acct\_balance)

'''5) Increment 'acct\_balance with 2000 and update the Balance field of same customer to new value'''

acct\_balance = list(acct\_balance)

acct\_balance[0]+=2000

cur.execute("update Account set Balance = :1 where CustomerId = 102",(acct\_balance[0]))

con.commit()

'''6) Fetch and observe updated account balance of Customer with CustomerId =102'''

cur.execute("select Balance from Account where CustomerId = 102")

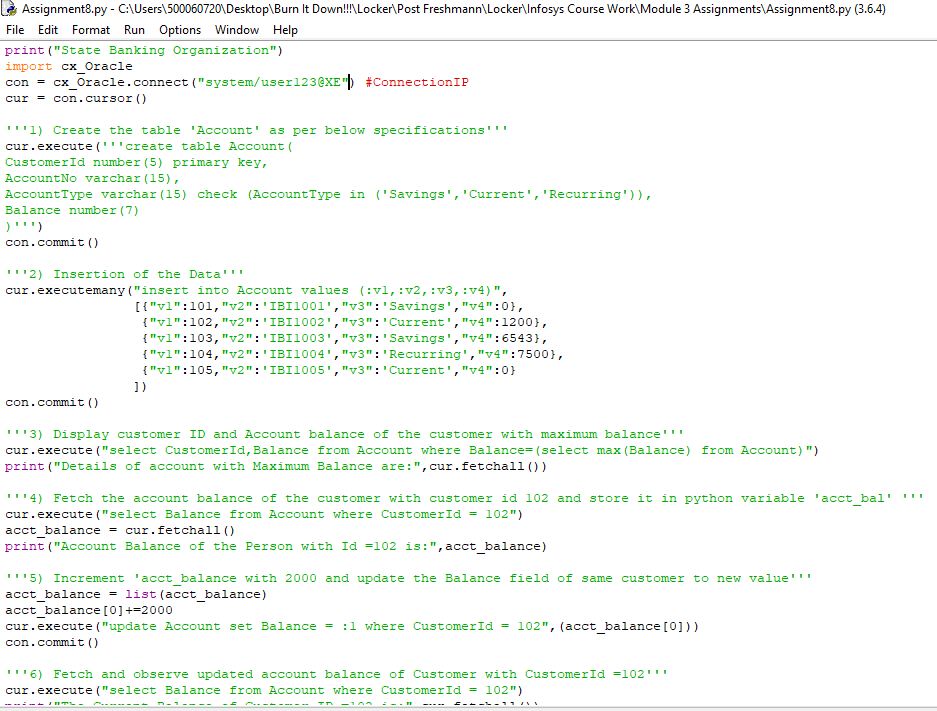
print("The Current Balance of Customer ID =102 is:",cur.fetchall())

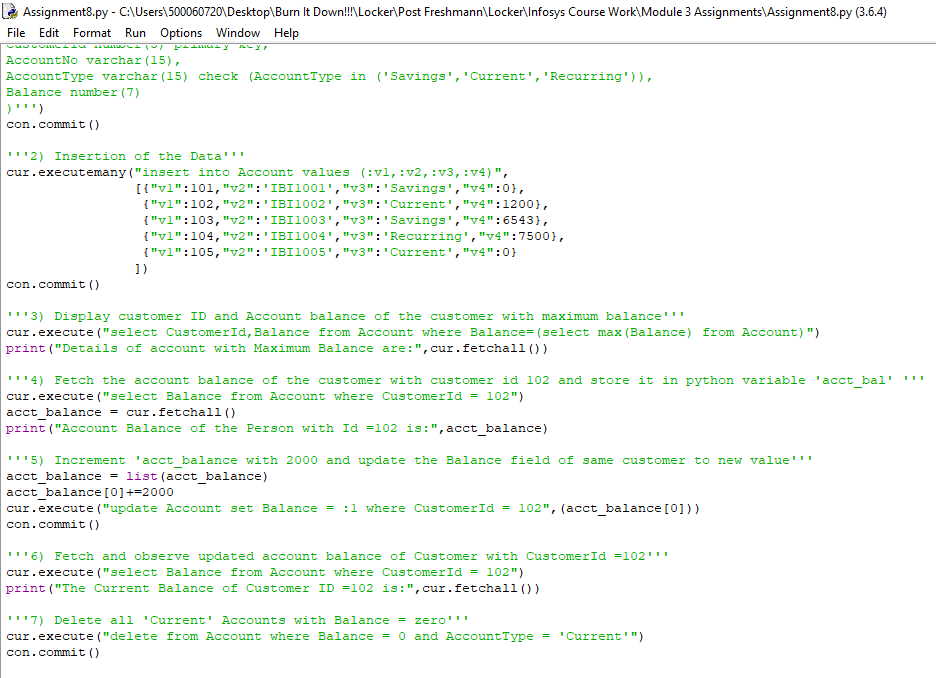
'''7) Delete all 'Current' Accounts with Balance = zero'''

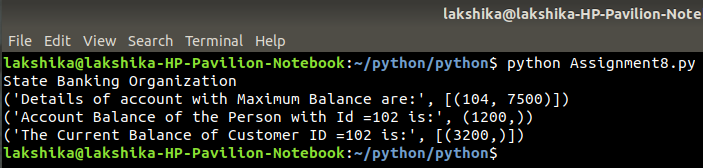
cur.execute("delete from Account where Balance = 0 and AccountType = 'Current'")

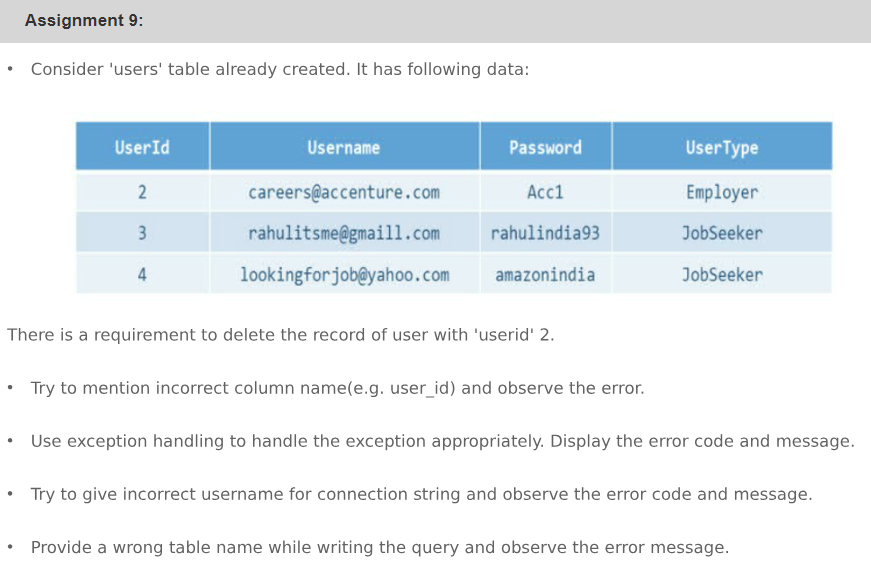
con.commit()

con.close()

**Assignment-8 Code Screenshot**

****

**Assignment-8 Output Screenshot**

**Assignment-9 Question**

**Assignment-9 code**

import cx\_Oracle

con = cx\_Oracle.connect("system/user123@XE") #Connection IP

cur = con.cursor()

try:

'''1) Try to mention incorrect column name and see the result '''

cur.execute("delete from Users where User\_Id = 2")

except cx\_Oracle.DatabaseError as e:

print("Error due to invalid column name is:",e)

finally:

print("In Finally Block")

print("Lets move to the next Task")

con.close()

try:

'''2) Enter wrong database connection string and check the Error message!!'''

con = cx\_Oracle.connect() #Please enter wrong connection Id

print("If this is printed! Connection Id entered was right!!!")

except cx\_Oracle.DatabaseError as e:

print("Error due to wrong connection string is:",e)

finally:

print("In Finally Block")

print("Lets move to the next Task")

try:

'''3) Enter wrong Table name and check the error!!!'''

con2 = cx\_Oracle.connect() #Enter correct string connection

cur = con2.cursor()

cur.execute("delete from Userss where UserId = 2")

except cx\_Oracle.DatabaseError as e:

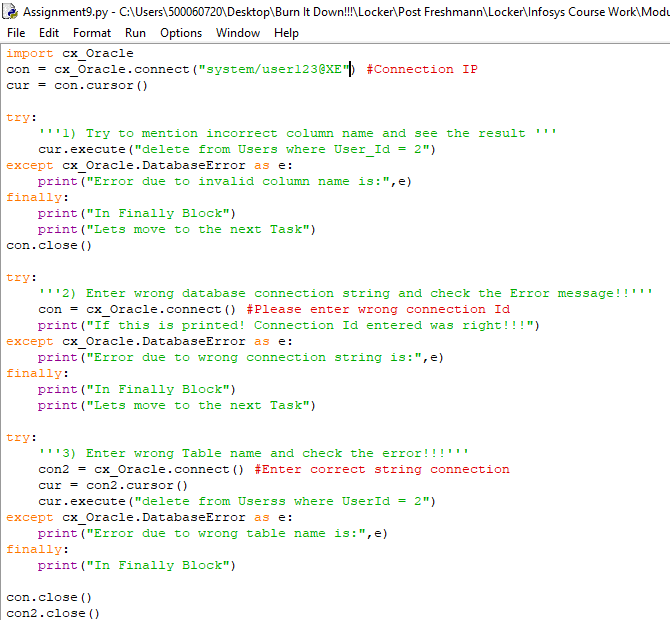
print("Error due to wrong table name is:",e)

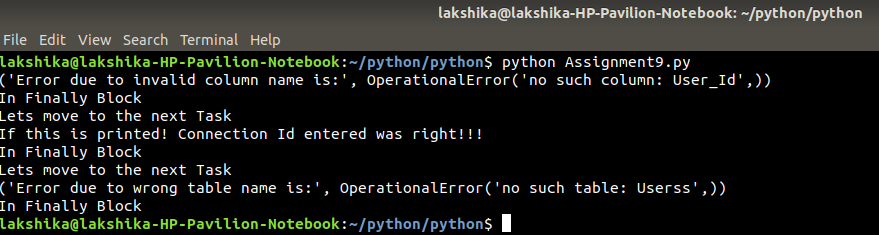
finally:

print("In Finally Block")

con.close()

con2.close()

**Assignment-9 Code Screenshot**

**Assignment-9 Output Screenshot**

**Assignment-10 Question**

**Assignment-10 code**

#Before executing this program create the table using attached file Assignment10\_Table

#Checking given program for errors

import cx\_Oracle

con = cx\_Oracle.connect("system/user123@XE") #Connection IP

cur = con.cursor()

try:

cur.execute("insert into product values ('P106','Jams',150)")

except cx\_Oracle.DatabaseError as e:

print("In except block! Exception encountered!!")

print("Encountered exception is:",e)

finally:

print("In finally block")

con.close()

#Correct code for insertion

con2 = cx\_Oracle.connect() #Connection IP

cur = con2.cursor()

try:

cur.execute("insert into product values ('P106','Jams',150,30)")

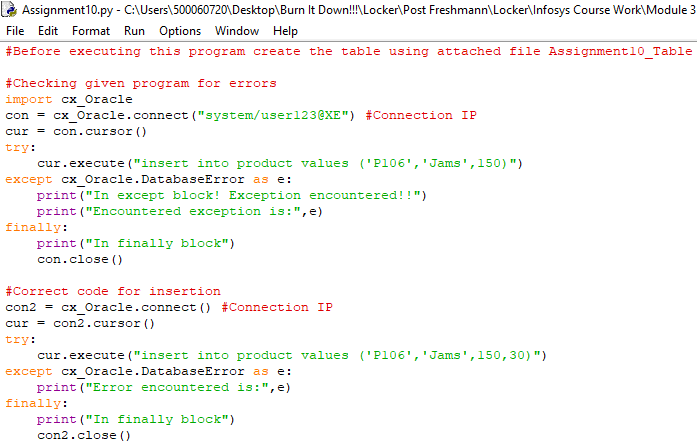
except cx\_Oracle.DatabaseError as e:

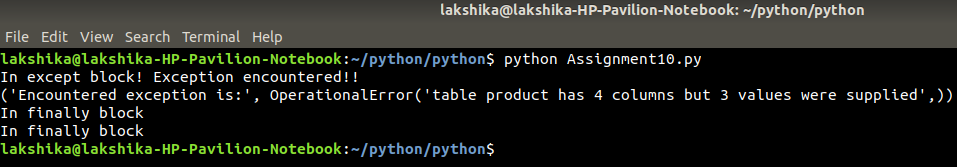
print("Error encountered is:",e)

finally:

print("In finally block")

con2.close()

**Assignment-10 Code Screenshot**

**Assignment-10 Output Screenshot**