CSLR 51 : DBMS LAB-8

Roll no. : **106119100**

Name : **Rajneesh Pandey**

Section : **CSE-B**

Design a simple database for Online Shopping Cart using Python to access the back-end MySQL database. The online shopping cart must contain the following modules.

```
import mysql.connector
from datetime import date, datetime
mydb = mysql.connector.connect(host="localhost", user='root',
password='12345')

mycursor = mydb.cursor()
mycursor.execute("DROP DATABASE shopDB")
mycursor.execute("CREATE DATABASE shopDB")
mycursor.execute("USE shopDB")
mycursor.execute("Create table product(id int PRIMARY KEY, name
varchar(20),category varchar(20), quantity int, price float, discount
float, manf date,exp date)")


cont = "T"
opt = 0
while (cont == "t" or cont == "T"):
    opt = int(input("1.
insert\n2.find\n3.search_in_category\n4.update\n5.provide_discount\n6.delet
e\n7.notification\nEnter your choice: "))
    if (opt == 1):
        insert_record()
    elif(opt == 2):
        find_record()
    elif(opt == 3):
        search_category()
    elif(opt == 4):
        update_record()
    elif(opt == 5):
        provide_discount()
    elif(opt == 6):
        delete_record()
    elif(opt == 7):
        notify()
    else:
        print("invalid option")
    cont = input("do you want to continue(T/F): ")
```

1.The insert module must be able to accept the prod_id(primary key),product name, category, quantity, price, discount, date of manufacture and date of expiry and store it in the database.

```
def insert_record():
        prod_id = int(input("Enter id: "))
        name = input("Enter name: ")
        category = input("Enter category: ")
        quantity = int(input("Enter quantity: "))
        price = float(input("Enter price in RS: "))
        discount = float(input("Enter discount in %: "))
        inputDate = input("Enter the date in format 'dd/mm/yy' : ")
        day1, month1, year1 = (inputDate.split('/'))
        manf = date(int(year1), int(month1), int(day1))
        inputDate = input("Enter the date in format 'dd/mm/yy' : ")
        day, month, year = inputDate.split('/')
        exp = date(int(year), int(month), int(day))
        sql = "INSERT INTO product(id, name, category, quantity, price,
discount,manf, exp) values (%s, %s, %s, %s, %s, %s, %s, %s)"
        val = (prod_id, name, category, quantity, price, discount, manf,
exp)
        mycursor.execute(sql, val)
        mydb.commit()
        print("record inserted")
```

Input/ Output :

```
PS D: \Documents\NIT Trichy\V-Sem\DBMS-LAB\Lab-8 > conda activate base
PS D: \Documents\NIT Trichy\V-Sem\DBMS-LAB\Lab-8 > python .\shopping.py
1.insert
2.find
3.search_in_category
4.update
5.provide_discount
6.delete
7.notification
Enter your choice: 1
Enter id: 1
Enter name: mobile
Enter category: electronics
Enter quantity: 5
Enter price: 10000.0
Enter discount: 10.0
Enter the date in format dd/mm/yy: 12/06/2007
Enter the date in format dd/mm/yy: 30/07/2027
record inserted
do you want to continue(T/F): T
1.insert
2.find
3.search_in_category
4.update
5.provide_discount
6.delete
7.notification
Enter your choice: 1
Enter id: 3
Enter name: earphones
Enter category: accessories
Enter quantity: 10
Enter price: 2000.0
Enter discount: 5.0
Enter the date in format dd/mm/yy: 11/06/2012
Enter the date in format dd/mm/yy: 30/10/2015
record inserted
do you want to continue(T/F): T
1.insert
2.find
3.search_in_category
4.update
5.provide_discount
6.delete
7.notification
Enter your choice: 1
Enter id: 4
Enter name: router
Enter category: electronics
Enter quantity: 3
Enter price: 5000.0
Enter discount: 2.0
Enter the date in format dd/mm/yy: 03/10/2009
Enter the date in format dd/mm/yy: 03/10/2014
record inserted
do you want to continue(T/F): F
```

2.The find module must be able to accept the name of the product and display all the details of the product.

```
def find_record():
        name = input("Enter name: ")
        sql = f'SELECT * FROM product WHERE name = "{name}"'
        mycursor.execute(sql)
        myresult = mycursor.fetchall()
        if (len(myresult) == 0):
            print("no records")
        else:
            for x in myresult:
                print(x)
```

Input/Output :

```
PS D:\Documents\NIT Trichy\V-Sem\DBMS-LAB\Lab-8> conda activate base
PS D:\Documents\NIT Trichy\V-Sem\DBMS-LAB\Lab-8> python .\shopping.py
1.insert
2.find
3.search_in_category
4.update
5.provide_discount
6.delete
7.notification
Enter your choice: 2
Enter name: router
(4, 'router', 'electronics', 3, 5000.0, 2.0,
datetime.date(2009, 10, 03), datetime.date(2014, 10, 03))
do you want to continue(T/F): F
```

3.The search_in_category module must be able to accept the name of the category and display all the details of the products in the categories.

```python
def search_category():
        category = input("Enter category: ")
        sql = f'SELECT * FROM product WHERE category = "{category}"'
        # que = (category)
        print(sql)
        mycursor.execute(sql)
        myresult = mycursor.fetchall()
        if (len(myresult) == 0):
            print("no records")
        else:
            for x in myresult:
                print(x)
```

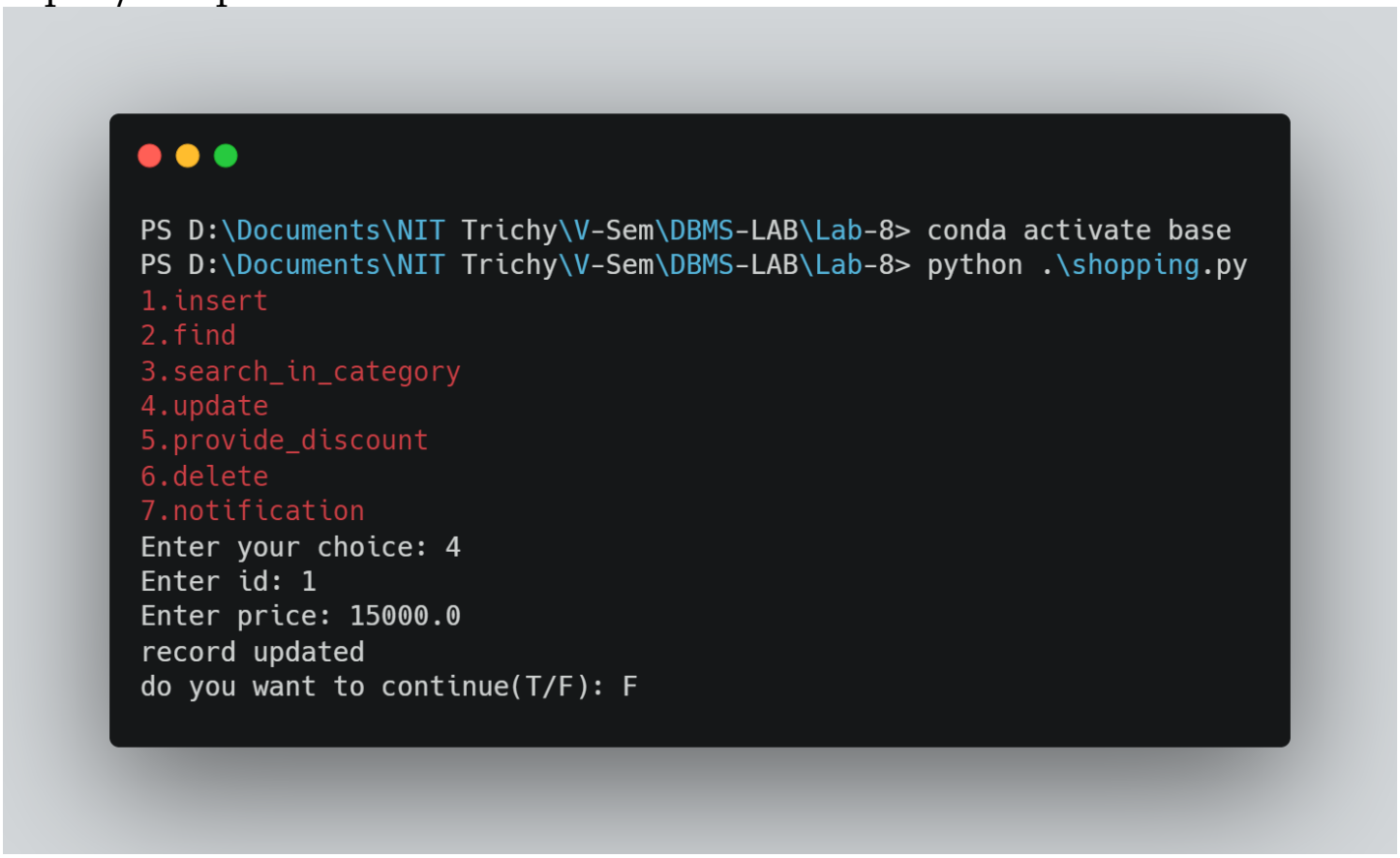
Input/Output :

```
PS D:\Documents\NIT Trichy\V-Sem\DBMS-LAB\Lab-8> conda activate base
PS D:\Documents\NIT Trichy\V-Sem\DBMS-LAB\Lab-8> python .\shopping.py
1.insert
2.find
3.search_in_category
4.update
5.provide_discount
6.delete
7.notification
Enter your choice: 3
Enter category: electronics
(1, 'mobile', 'electronics', 5, 10000.0, 10.0,
datetime.date(2007, 06, 12), datetime.date(2027, 07, 30))
(4, 'router', 'electronics', 3, 5000.0, 2.0,
datetime.date(2009, 10, 03), datetime.date(2014, 10, 03))
do you want to continue(T/F): F
```

4.The update module must be able to update the price of the products.

```
def update_record():
        prod_id = int(input("Enter id: "))
        price = float(input("Enter price: "))
        sql = f'UPDATE product SET price = "{price}" WHERE id =
"{prod_id}"'
        # val = (price, prod_id)
        mycursor.execute(sql)
        mydb.commit()
        print("record updated")
```

Input/Output :



5.The provide_discount module must be able to provide the discount of the products in a particular category.

```
def provide_discount():
        category = input("Enter category: ")
        sql = f'SELECT name, discount FROM product WHERE category =
"{category}"'
        # que = (category)
        mycursor.execute(sql)
        myresult = mycursor.fetchall()
        if (len(myresult) == 0):
            print("no records")
        else:
            for x in myresult:
                print(x[0]+' : %d'%x[1])
```

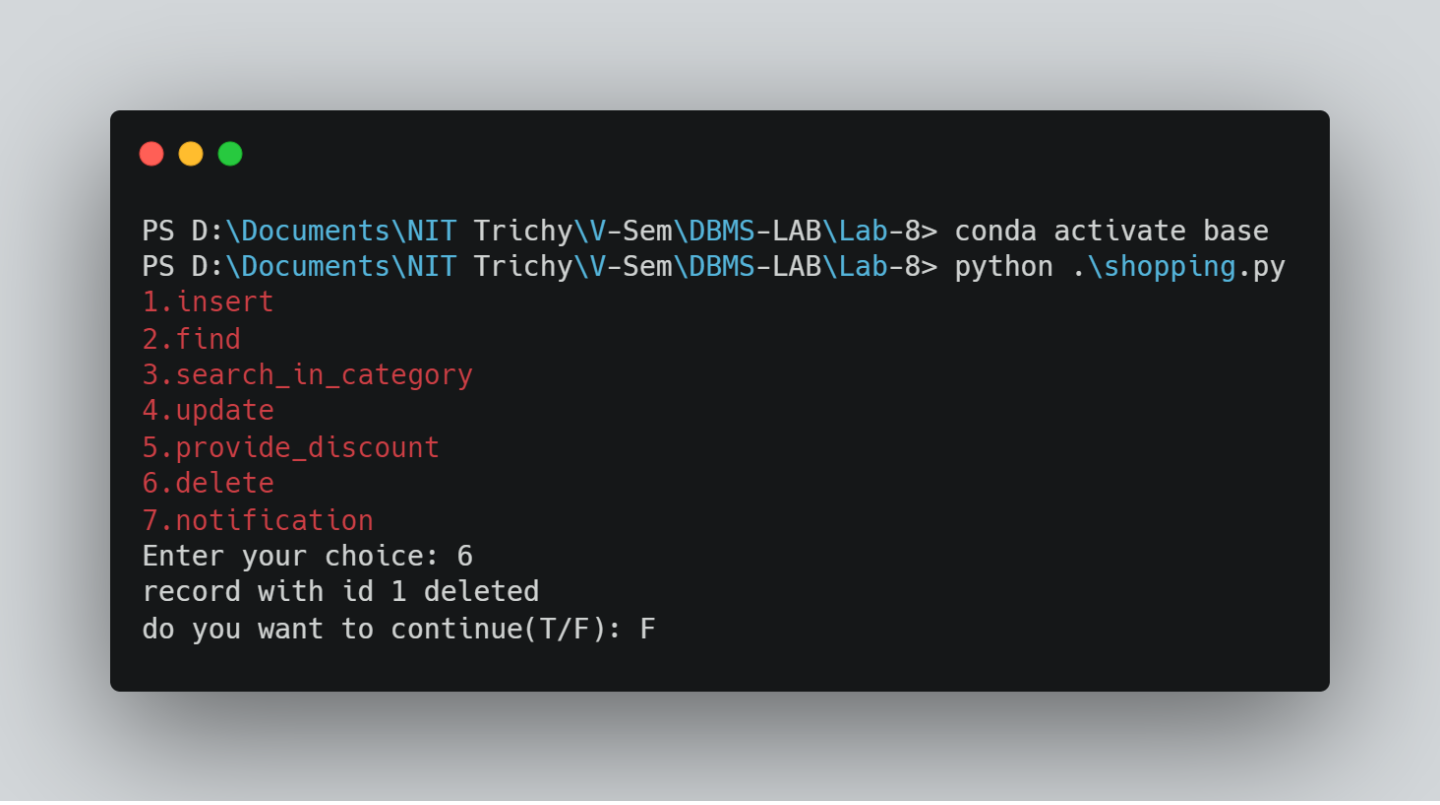
Input/Output :

```
PS D:\Documents\NIT Trichy\V-Sem\DBMS-LAB\Lab-8> conda activate base
PS D:\Documents\NIT Trichy\V-Sem\DBMS-LAB\Lab-8> python .\shopping.py
1.insert
2.find
3.search_in_category
4.update
5.provide_discount
6.delete
7.notification
Enter your choice: 5
Enter category: accessories
earphones: 5.0
do you want to continue(T/F): F
```

6.The delete_module must be able to delete/ cancel the product if the expiry date is not valid.

```
def delete_record():
        mycursor.execute('SELECT * FROM product')
        myresult = mycursor.fetchall()
        if (len(myresult) == 0):
            print("no records")
        else:
            for x in myresult:
                if (x[7] < datetime.now().date()):
                    print("record with id %d deleted" % x[0])
        sql = f"DELETE FROM product WHERE id = {x[0]}"
        mycursor.execute(sql)
        mydb.commit()
```

Input/Output :



7.The notification_module must indicate free shipping if the total product price exceeds Rs.1000.

```
def notify():
        prod_id = int(input("Enter id: "))
        sql = f'SELECT price, discount FROM product WHERE id = {prod_id}'
        que = (prod_id)
        mycursor.execute(sql, que)
        myresult = mycursor.fetchall()
        if (len(myresult) == 0):
            print("no records")
        else:
             s = 0
             for x in myresult:
                s += x[0]
                s -= (x[0]*x[1]*0.01)
             if (s > 1000):
               print("Free shipping")
             else:
               print("no free shipping")
```

Input/Output :