```
def dropTable():
    try:
        con = sqlite3.connect('test.db')
        con.execute('DROP TABLE IF
EXISTS product')
    except con.Error as e:
        if con:
            print("error is " + e)
    con.close()
def createTable():
    dropTable()
    try:
        con = sqlite3.connect('test.db')
        sql = 'CREATE TABLE product (pid
INT PRIMARY KEY NOT NULL,
 pname TEXT, pqty REAL,pcost REAL,
pprice REAL);'
        con.execute(sql)
    except con.Error as e:
        if con:
            print("error is " + e)
    con.close()
def insertTable(id, name, qty, cost):
    try:
        con = sqlite3.connect('test.db')
        sql =   'INSERT INTO product (pid,
pname, pqty, pcost, pprice) '\
              'VALUES ("' +
id+'","'+name+'",'+str(qty)+','+str(cost
)+','+str(cost*1.2) + ')'
        print(sql)
        con.execute(sql)
        con.commit()
    except con.Error as e:
        if con:
            print("error is "+e)
            con.rollback();
    finally:
        if con:
            con.close()
```

```
def updateTable(id, name, qty, cost):
    try:
        con = sqlite3.connect('test.db')
        sql='UPDATE product SET
pname="'+name+'",pqty=
'+str(qty)+',pcost= '+ str(cost) + ', pprice =
' + str(cost*1.2) + ' WHERE PID ="'+id+'"'
        print(sql)
        con.execute(sql)
        con.commit()
    except con.Error as e:
        if con:
            print("error is "+e)
            con.rollback();
    finally:
        if con:
            con.close()
def selectTable(id):
    try:
        con = sqlite3.connect('test.db')
        if id == "*":
            sql = 'SELECT * FROM product'
        else:
            sql = 'SELECT * FROM product
WHERE pid ="' + id + '"'
        data = con.execute(sql)
        print("%-8s%-15s%10s%10s%10s" %
('ID', 'NAME', 'QUANTITY', 'COST',
'PRICE'))
        for row in data:
            print("%-8s%-15s%10s%10s%10s"
% (row[0], row[1], row[2], row[3],
row[4]))
    except con.Error as e:
        if con:
            print("error is " + e)
    finally:
        if con:
            con.close()
```

```
def selectforsale(id):
    global result
    try:
        con = sqlite3.connect('test.db')
        sql = 'SELECT * FROM product
WHERE pid ="' + id + '"'
        data = con.execute(sql)
        for row in data:
            result = row[1] + ';' +
str(row[4])+ ';' + str(row[2])
        return result
    except con.Error as e:
        if con:
            print("error is " + e)
    finally:
        if con:
            con.close()
def updateSale(pid,qty):
    try:
        con = sqlite3.connect('test.db')
        sql='UPDATE product SET
pqty="'+str(qty)+'"' + ' WHERE PID
="'+pid+'"'
        print(sql)
        con.execute(sql)
        con.commit()
    except con.Error as e:
        if con:
            print("error is "+ e)
            con.rollback();
    finally:
        if con:

            con.close()
```

```
def createUser():
    # drop
    try:
        con = sqlite3.connect('test.db')
        con.execute('DROP TABLE IF
EXISTS signin')
    except con.Error as e:
        if con:
            print("error is " + e)
    con.close()
    #create
    try:
        con = sqlite3.connect('test.db')
        sql = 'CREATE TABLE signin
(uname TEXT PRIMARY KEY NOT
NULL, upass TEXT);'
        con.execute(sql)
    except con.Error as e:
        if con:
            print("error is " + e)
    con.close()
#insert
    try:
        con = sqlite3.connect('test.db')
        sql = 'INSERT INTO signin (uname,
upass)''VALUES ("+ admin+","1234")'

        con.execute(sql)
        con.commit()
    except con.Error as e:
        if con:
            print("error is "+e)
            con.rollback();
    finally:
        if con:
            con.close()
```

```
def selectsignin (uname,upass):
    global result
    try:
        con = sqlite3.connect('test.db')
        sql = 'SELECT * FROM signin
WHERE uanme ="' + uname + 'and
upass = "'+upass+'""
        data = con.execute(sql)

        for row in data:
            result = row[0] + ';' +
(row[1])
        return result
        if con:
            print("error is " + e)
    finally:
        if con:
            con.close()

    from MyDB import *
    '''createUser()


uname = input('Enter Username')
upass = input('Enter Password')
result = selectsignin(uname, upass)
data = result.split(";")
name = data[0]
psw = data[1]
if uname == name&psw == upass:
    print("Welcome")
else:
    print("Sorry")
'''
print('1. create  table')
print('2. insert data')
print('3. update data')
print('4. select data')
print('5. sales data')
n = 0
```

```
while n != 9:
    n = int(input('Enter  choice: 1-4 >>>  '))
    if  not ((n >= 0) and  (n <= 6)):
        print('ERROR')
        n = int(input('Enter chioce: 1-5 >>> '))
    if n == 1:
        createTable()

    if n == 2:
        id = input('Enter ID >>> ')
        name = input('Enter name >>>')
        qty = int(input('Enter QUANTITY >>> '))
        cost = float(input('Enter COST >>> '))
        insertTable(id, name, qty, cost)

    if n == 3:
        id = input('Enter ID >>> ')
        selectTable(id)
        name = input('Enter name >>>')
        qty = int(input('Update QUANTITY >>>
'))
        cost = float(input('Update COST >>> '))
        updateTable(id, name, qty, cost)

    if n ==4:
        selectTable("*")

if n == 5:
        sid = input('Enter Sales ID >>>')
        pid = input('Enter producr ID >>> ')
        sqty = int(input('Enter quantity :  '))

        insertSales(sid,pid,sqty)

        result = selectforsale(pid)

        data = result.split(";")
        name = data[0]
        price = float(data[1])
        qty = float(data[2])
        print(qty)
        if qty < sqty:
            print("Quantity is invalid")
        else:
            print('name = ' +name + '\nQuantity = ' +
str(sqty) +'\nprice = ' + str(price) + '\ntotal = ' +
str(price*sqty))

            updateSale(pid,(qty-sqty))

        break
```

```
if n == 6:
        sid = input('Enter producr ID >>> ')
        sqty = int(input('Enter quantity :  '))
        result = selectforsale(sid)
        data = result.split(";")
        name = data[0]
        price = float(data[1])
        qty = float(data[2])
        print(qty)
        if sqty <= 0:
            print("Quantity is invalid")
        else:
            print('name = ' +name + '\nQuantity = ' +
str(sqty) +'\nprice = ' + str(price) + '\ntotal = ' +
str(price*sqty))
            updateSale(sid, (qty+sqty))

    if n == 0:
        print('SEE U')
        break
```