CONTENTS

Acknowledgement

I would like to express my heartfelt gratitude to our respected Principal, **Dr. Asheesh Mishra**, for providing us with the opportunity and resources to work on this Computer Science project.

I extend my sincere appreciation to my Computer Science teacher, **Ms. Sona O.K.**, for her continuous guidance, support, and encouragement throughout the project. Her expertise and feedback have been invaluable in helping me complete this work successfully.

Lastly, I thank my family and friends for their unwavering support and motivation.

INTRODUCTION

This project aims to create a management system for a clothing store, designed to streamline inventory operations. It includes features like adding, updating, deleting, and displaying product records to ensure efficient stock management.

Using Python as the primary programming language and MySQL as the database, the system is both functional and user-friendly. This project demonstrates the practical application of programming and database concepts to solve real-world challenges.

FUNCTION:

A function is a block of code, which only runs when it is called. You can pass data, known as parameters, into a function. A function can return data as a result.

1. Built-in functions: These predefined functions are always available for use. Eg: - len(), int() etc.
2. Functions defined in modules: These are predefined functions available in specific modules and can be used only when that module is imported. Eg: - random.randint() gives a random integer value in a given interval. This is available in the random module.
3. User defined functions: the user defines these functions.

Functions used in this project:

Built-in functions	User-defined functions
• cursor() • execute() • print() • int() • fetchall() • commit() • fetchone() • connect()	• menu() • adddata() • update1() • delete1() • display1()

MODULES:

A Python module is a file containing Python definitions and statements. A module can define functions, classes, and variables.

Modules used in this project:

Mysql.connector: This method sets up a connection, establishing a session with the MySQL server. If no arguments are given, it uses the already configured or default values. A connection with the MySQL server can be established using either the mysql.connector.connect() method or the mysql.connector.MySQLConnection()

USER MANUAL

This program is designed to help manage the inventory and operations of a clothing store. Below are the features of the program and instructions for using it:

Main Menu Options:

After running the program, you will be greeted with the main menu. Select an option by entering its corresponding number.

1. **Add Record**
    - Allows the admin to add a new product to the inventory.
    - Enter the product number, ID, name, price, and stock when prompted.
2. **Update Record**
    - Allows the admin to update the price of an existing product.
    - Provide the product ID and the new price when prompted.
3. **Delete Record**
    - Enables the admin to delete a product from the inventory.

- Provide the product ID to remove the record.

4. **Display Record**

   - Displays all the records of products in the inventory, including product number, ID, name, price, and stock.

5. **Close the Program**

   - Ends the program session.

SOURCE CODE

```
import mysql.connector
```

#CREATING THE DATABASE

```
mydb=mysql.connector.connect(host="localhost",user="root",passwd="suraj1807")
my=mydb.cursor()
my.execute("create database if not exists Cloth")
mydb.commit()
print("database created")
my.close()
mydb.close()
```

#CREATING TABLE

```
mydb=mysql.connector.connect(host="localhost",user="root",passwd="suraj1807",
database="Cloth")
my=mydb.cursor()
my.execute("create table if not exists store(pno int,pid int,pname
varchar(100),price float,stock int)")
print("table store created")
mydb.commit()
my.close()
mydb.close()
```

#INSERTING VALUE INTO TABLE

```
mydb=mysql.connector.connect(host="localhost",user="root",passwd="suraj1807",
database="Cloth")

my=mydb.cursor()

my.execute("insert into store values(101,30567,'silk chiffon dress',400000,6500)")

my.execute("insert into store values(102,30568,'polka dots and GG silk
dress',370000,4000)")

my.execute("insert into store values(103,30569,'viscose rib stitch
dress',170000,7000)")

my.execute("insert into store values(104,30570,'gemstone embellished fitted
dress',310000,5000)")

my.execute("insert into store values(105,30571,'silk viscose jumsuit with
cuffs',447000,4000)")

my.execute("insert into store values(106,30572,'checked shirts
dress',340000,3000)")

my.execute("insert into store values(107,30573,'wool dress with
pleats',430000,8000)")

my.execute("insert into store values(108,30574,'viscose dress with double G
chain',317000,3000)")

mydb.commit()

print("values inserted")

my.close()

mydb.close()
```

#PROGRAM TO PERFOR ALL THE OPERATIONS ON TABLE

```
import mysql.connector


def menu():
    c='y'
    while c=='y':
        print(">>>>>>>>>>>>>>>>WELCOME TO ORCHIDS CLOTHING
        STORE<<<<<<<<<<<<<<<<")
        print("1.add record")
        print("2.update record")
        print("3.delete record")
        print("4.display record")
        print("5.close the program")
        choice=int(input('\nEnter your choice: '))
        if choice==1:
            adddata()
        elif choice==2:
            update1()
        elif choice==3:
            delete1()
        elif choice==4:
            display1()
        elif choice==5:
            print("\nClosing the program")
            break
```

```
    else:
        print("\nPlease check")
  c=input("\nDo you want to continue..(y/n)")
```

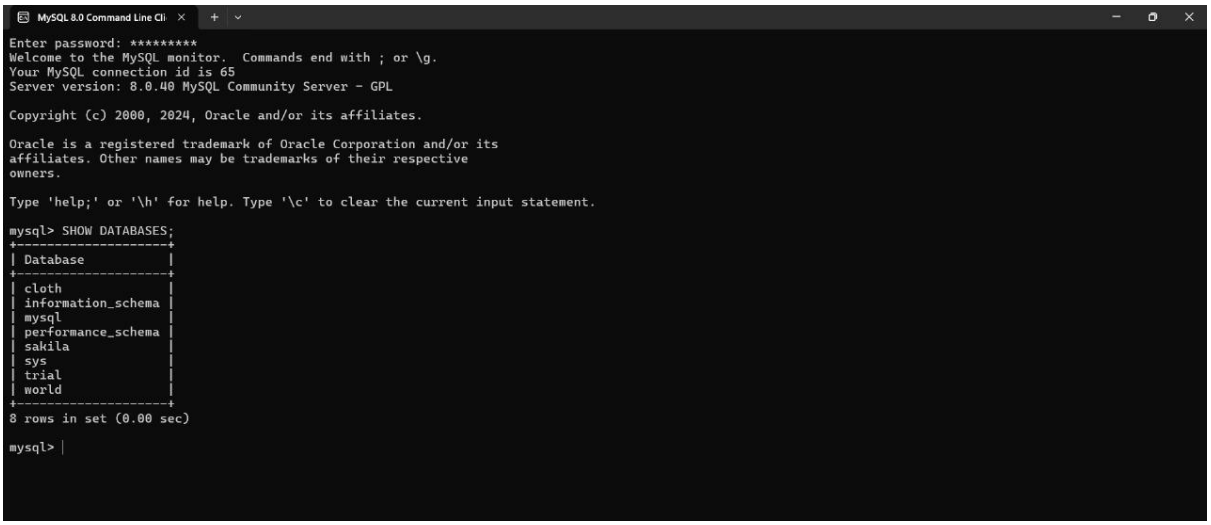
#PERFORMING ALL THE OPERATIONS

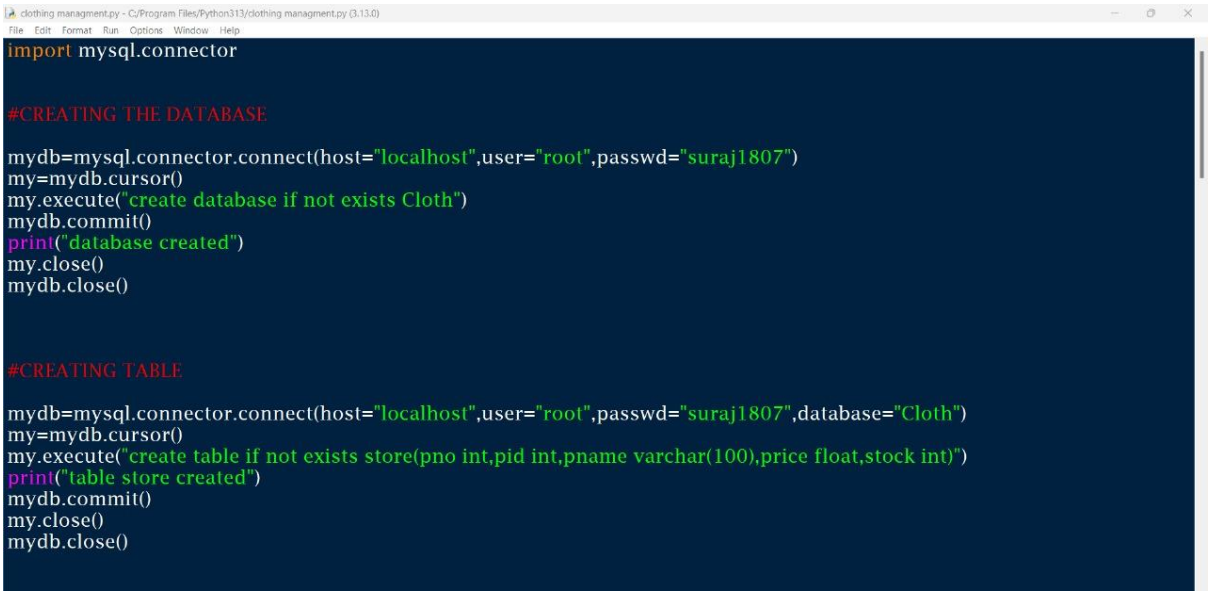
```
def adddata():
  mydb=mysql.connector.connect(host="localhost",user="root",passwd="suraj180
   7",database="Cloth")
  my=mydb.cursor()
  pno=int(input('\nEnter the product no'))
  pid=int(input('Enter the product id'))
  pname=input('Enter the product name')
  price=int(input('Enter the price of the product'))
  sa=int(input('Enter the stock available'))
  q="insert into store values({},{},'{}',{},{})".format(pno,pid,pname,price,sa)
  my.execute(q)
  mydb.commit()
  print("\nValue inserted")
  my.close()
  mydb.close()

def update1():
  try:
    mydb=mysql.connector.connect(host="localhost",user="root",passwd="suraj1
    807",database="Cloth")
    my=mydb.cursor()
    pid=int(input("\nEnter the product id to be changed"))
```

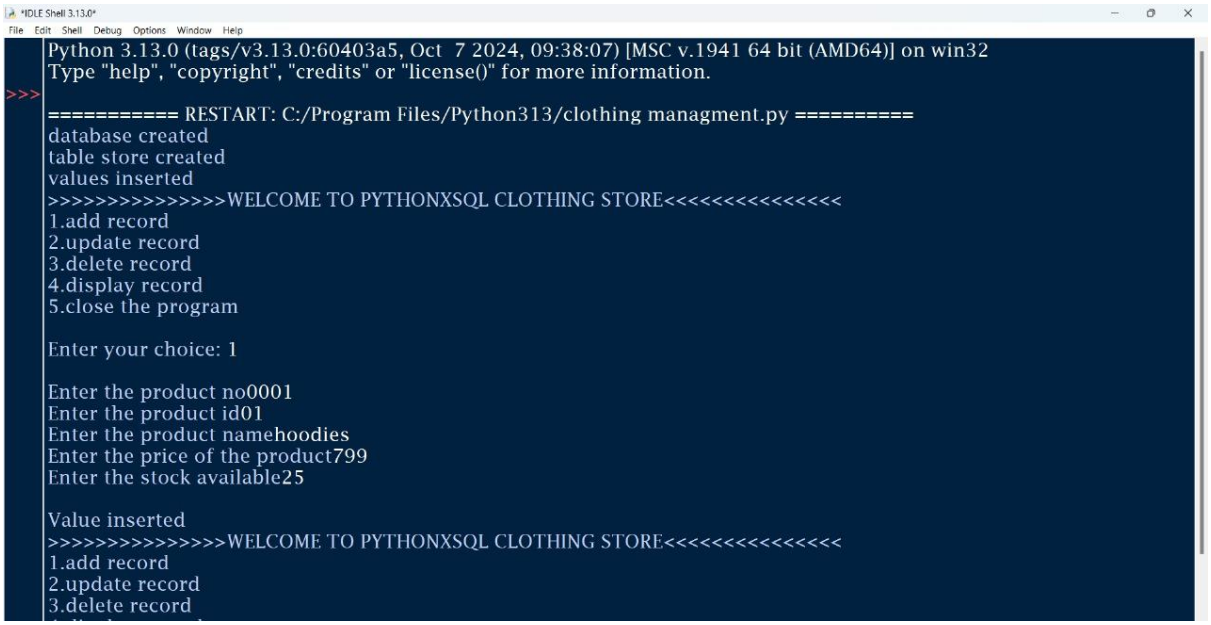
```
        price=int(input('Enter the price to be changed'))
        q="update store set price={} where pid={}".format(price,pid)
        my.execute(q)
        mydb.commit()
        my.close()
        mydb.close()
        print("\nUpdated Successfully")
    except mysql.connector.Error as error:
        print("\nFailed to update record to database: {}".format(error))


def delete1():
    mydb=mysql.connector.connect(host="localhost",user="root",passwd="suraj180
    7",database="Cloth")
    my=mydb.cursor()
    pid=int(input('Enter the respective product id'))
    t="delete from store where pid={}".format(pid)
    my.execute(t)
    mydb.commit()
    print("\nDeleted")
    my.close()
    mydb.close()


def display1():
    mydb=mysql.connector.connect(host="localhost",user="root",passwd="suraj180
    7",database="Cloth")
    my=mydb.cursor()
    my.execute("select * from store")
```

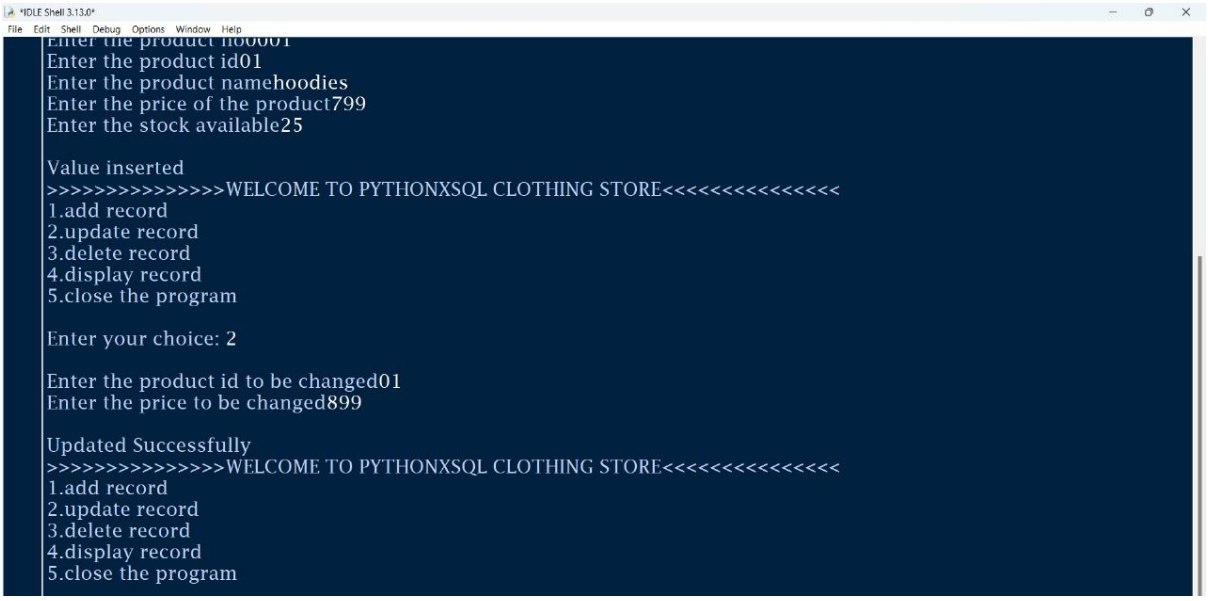
```
    k=my.fetchall()
    for a in k:
        print(a)
        my.close()
        mydb.close()
menu()
```

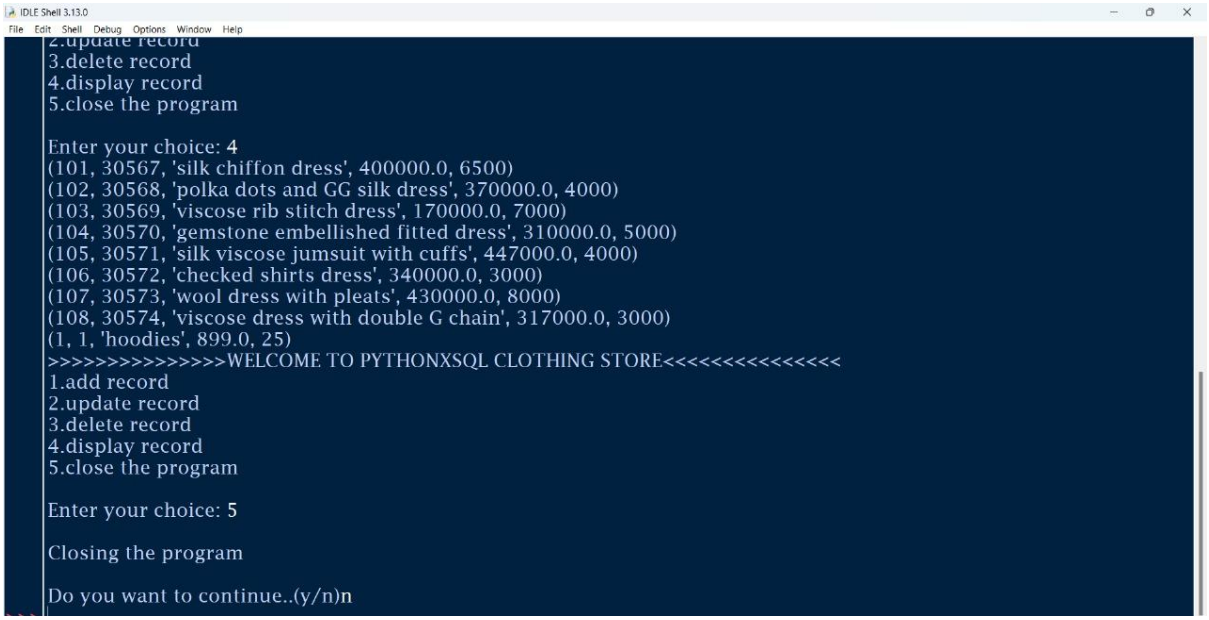
OUTPUT

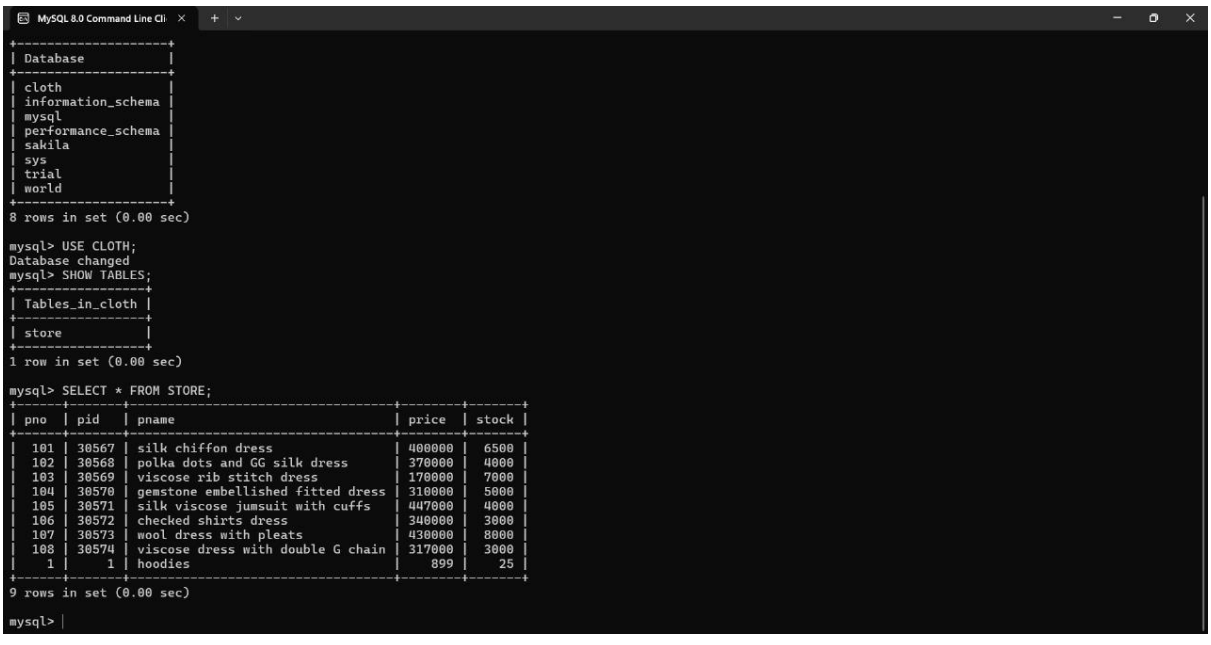

MySQL 8.0 Command Line Cli
Enter password: *********
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 65
Server version: 8.0.40 MySQL Community Server - GPL
Copyright (c) 2000, 2024, Oracle and/or its affiliates.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> SHOW DATABASES;
| Database |
| cloth |
| information_schema |
| mysql |
| performance_schema |
| sakila |
| sys |
| trial |
| world |
8 rows in set (0.00 sec)
mysql>


clothing managment.py - C:/Program Files/Python313/clothing managment.py (3.13.0)
File Edit Format Run Options Window Help
import mysql.connector
#CREATING THE DATABASE
mydb=mysql.connector.connect(host="localhost",user="root",passwd="suraj1807")
my=mydb.cursor()
my.execute("create database if not exists Cloth")
mydb.commit()
print("database created")
my.close()
mydb.close()
#CREATING TABLE
mydb=mysql.connector.connect(host="localhost",user="root",passwd="suraj1807",database="Cloth")
my=mydb.cursor()
my.execute("create table if not exists store(pno int,pid int,pname varchar(100),price float,stock int)")
print("table store created")
mydb.commit()
my.close()
mydb.close()


IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:/Program Files/Python313/clothing managment.py ==========
database created
table store created
values inserted
>>>>>>>>>>>>>>>>WELCOME TO PYTHONXSQL CLOTHING STORE<<<<<<<<<<<<<<<<
1.add record
2.update record
3.delete record
4.display record
5.close the program
Enter your choice: 1
Enter the product no0001
Enter the product id01
Enter the product namehoodies
Enter the price of the product799
Enter the stock available25
Value inserted
>>>>>>>>>>>>>>>>WELCOME TO PYTHONXSQL CLOTHING STORE<<<<<<<<<<<<<<<<
1.add record
2.update record
3.delete record


IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Enter the product id01
Enter the product namehoodies
Enter the price of the product799
Enter the stock available25
Value inserted
>>>>>>>>>>>>>>>>WELCOME TO PYTHONXSQL CLOTHING STORE<<<<<<<<<<<<<<<<
1.add record
2.update record
3.delete record
4.display record
5.close the program
Enter your choice: 2
Enter the product id to be changed01
Enter the price to be changed899
Updated Successfully
>>>>>>>>>>>>>>>>WELCOME TO PYTHONXSQL CLOTHING STORE<<<<<<<<<<<<<<<<
1.add record
2.update record
3.delete record
4.display record
5.close the program


IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
3.delete record
4.display record
5.close the program
Enter your choice: 4
(101, 30567, 'silk chiffon dress', 400000.0, 6500)
(102, 30568, 'polka dots and GG silk dress', 370000.0, 4000)
(103, 30569, 'viscose rib stitch dress', 170000.0, 7000)
(104, 30570, 'gemstone embellished fitted dress', 310000.0, 5000)
(105, 30571, 'silk viscose jumsuit with cuffs', 447000.0, 4000)
(106, 30572, 'checked shirts dress', 340000.0, 3000)
(107, 30573, 'wool dress with pleats', 430000.0, 8000)
(108, 30574, 'viscose dress with double G chain', 317000.0, 3000)
(1, 1, 'hoodies', 899.0, 25)
>>>>>>>>>>>>>>>>WELCOME TO PYTHONXSQL CLOTHING STORE<<<<<<<<<<<<<<<<
1.add record
2.update record
3.delete record
4.display record
5.close the program
Enter your choice: 5
Closing the program
Do you want to continue..(y/n)n


MySQL 8.0 Command Line Cli
+--------------------+
| Database |
+--------------------+
| cloth |
| information_schema |
| mysql |
| performance_schema |
| sakila |
| sys |
| trial |
| world |
+--------------------+
8 rows in set (0.00 sec)
mysql> USE CLOTH;
Database changed
mysql> SHOW TABLES;
+----------------+
| Tables_in_cloth |
+----------------+
| store |
+----------------+
1 row in set (0.00 sec)
mysql> SELECT * FROM STORE;
+------+-------+-----------------------------------+--------+-------+
| pno | pid | pname | price | stock |
+------+-------+-----------------------------------+--------+-------+
101	30567	silk chiffon dress	400000	6500
102	30568	polka dots and GG silk dress	370000	4000
103	30569	viscose rib stitch dress	170000	7000
104	30570	gemstone embellished fitted dress	310000	5000
105	30571	silk viscose jumsuit with cuffs	447000	4000
106	30572	checked shirts dress	340000	3000
107	30573	wool dress with pleats	430000	8000
108	30574	viscose dress with double G chain	317000	3000
1	1	hoodies	899	25
+------+-------+-----------------------------------+--------+-------+
9 rows in set (0.00 sec)
mysql>

BIBLIOGRAPHY

1. Computer Science with Python by Sumita Arora for Class XII
2. Computer Science with Python by Sumita Arora for Class XI