SNBP International School, Rahatani, Pune

Academic Year ~ 2021-22

Project Report

Inventory Management System

Roll No.	:	12505
Name	:	Apoorv Ajay Thite
Class	:	XII-E
Subject	:	Informatics Practices
Submitted to	:	Mrs. Aparna Bansal (IP Teacher)

ACKNOWLEDGEMENT

I would like to thank and express my gratitude to to my IP teacher, **Mrs. Aparna Bansal** for her valuable support throughout the course of this project. Her continuous motivation and constructive advice have been a true integral part for the successful completion of the project. I would also like to thank Principal **Mrs. Jayshree Venkataraman** for her precious encouragement and and for being a great inspiration.

I would also like to express my heartfelt gratitude to my parents for being supportive, caring and understanding.

SNBP INTERNATIONAL SCHOOL, RAHATANI

CERTIFICATE

This is to certify that ____Apoorv Ajay Thite ____, student of class ___XII-E____ has successfully completed the __Informatics Practices Inventory Management System___project under the guidance of Ms. ___Aparna Bansal___during the year 2021-22 in the partial fulfilment of practical examination conducted by AISSCE, New Delhi.

Signature of external examiner

Signature of the subject teacher

INDEX

INTRODUCTION

Inventory Management or Stock Management is a system of ordering, purchasing, storing resources and materials.

The Practice of Inventory Management is useful in every sector of industry and every item a business uses to produce its products or services - From Raw Materials to Intermediate and Final Products.

My main objective behind choosing this topic for my project is considering the fact that my project could explore through wide areas of fields and be useful to more number of people to make their life sorted and less complicated.

There are numerous advantages of having a reliable inventory management system:-

1. Helps in keeping a hassle-free track of the orders, sales, stock of the items

2. Ensures justified investment of cash by buying the right quantity of each product, so as to not run out of supply and also not have a product in excess.

3. Customer Satisfaction is paramount and this system makes sure that the customer does not face any difficulty in ordering an item and keeping record of it

4. Certain threats and financial losses are almost zeroed down as the system helps in identifying the problem quickly and resolving it without any delay

5. Running an efficient and profitable business is all about sharing and using accurate information. One way to do this is through an software system that integrates with everything from your point of sale devices to barcode scanners. This System does exactly that.

ABOUT THE PROJECT

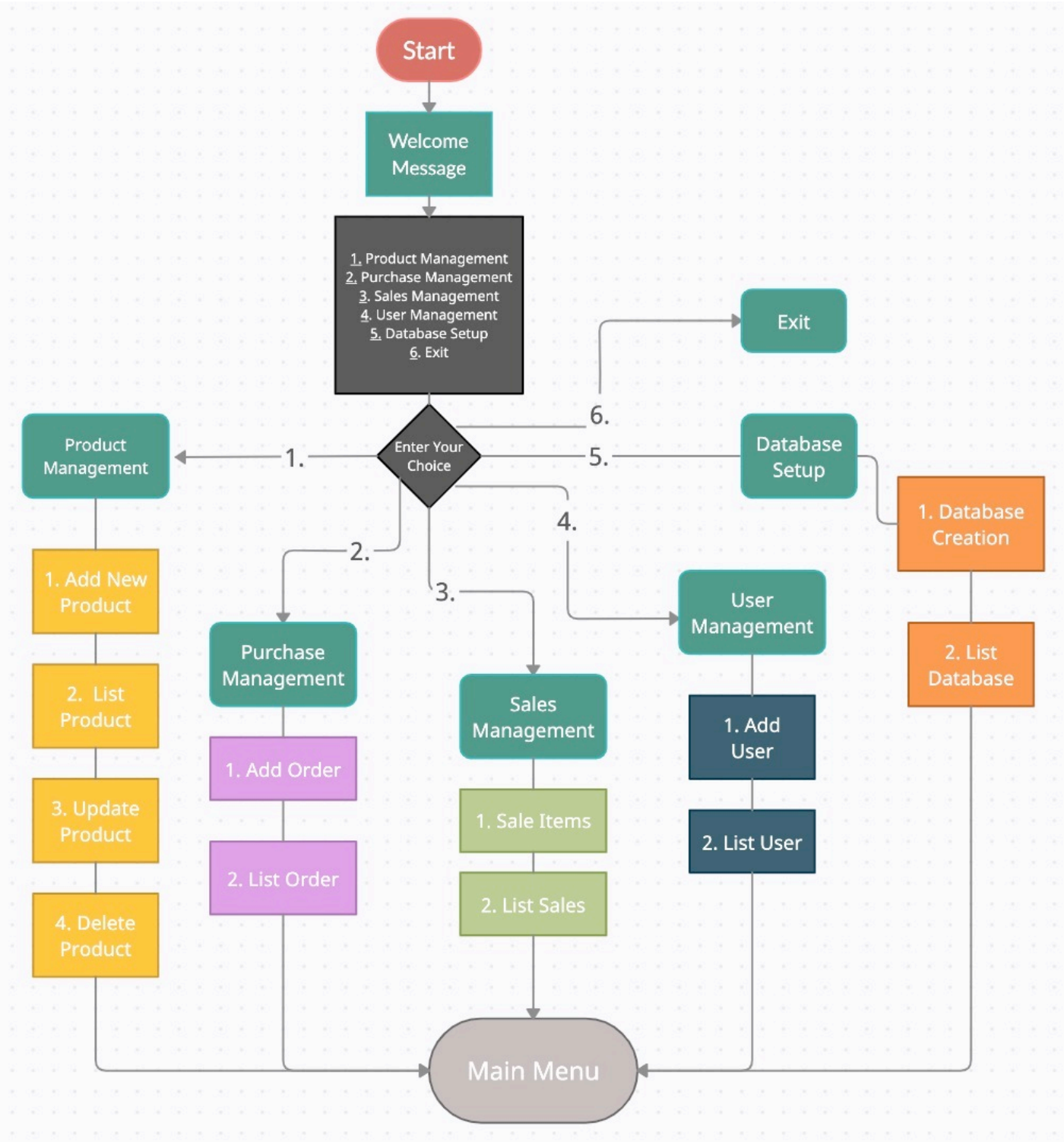
The Project contains the following modules :-

1. **Product Management** - To Add, Update, Delete Product
2. **Purchase Management** - Managing the Purchases and Orders
3. **Sales Management** - Managing the sales
4. **User Management** - To Add or Delete an User
5. **Database Setup** - Setting up the primary Database System

As the user selects a number from 1-5, he is directed to the particular module

- The Product Management module helps the user to add a product, update the specifications of the product and delete any of those
- The Purchase Management module manages the purchases and order requirements by enabling the users to put in their orders
- The Sales Management provides the user an option to manage the sales of the product like selling a particular item from the stock
- The User Management involves the creation of a proper account of the user by painting their details so as to give it a personalised touch.
- The 5th Module is the Database setup which is a one time creation of a database and subsequent addition of tables based on the particular requirements of the business.

FLOWCHART


Start
Welcome Message
1. Product Management
2. Purchase Management
3. Sales Management
4. User Management
5. Database Setup
6. Exit
Enter Your Choice
1.
2.
3.
4.
5.
6.
Exit
Product Management
1. Add New Product
2. List Product
3. Update Product
4. Delete Product
Purchase Management
1. Add Order
2. List Order
Sales Management
1. Sale Items
2. List Sales
User Management
1. Add User
2. List User
Database Setup
1. Database Creation
2. List Database
Main Menu

OBJECTIVES AND SCOPE

OBJECTIVES:-

The Main Objectives kept in mind while creating this Inventory Management System are:-

1. To make the system Well-Structured, Efficient and User-Friendly
2. To ensure a simple layout for optimum usage
3. To make the system compatible to different devices and users

SCOPE:-

Future Scope:-

1. A set of new modules will be added which will ensure a more detailed record of the products.
2. The User-Interface (UI) of the output will be improved to have more customised look to the system
3. The system will have more inbuilt mechanism of autofilling certain data

HARDWARE AND SOFTWARE SPECIFICATIONS

SOFTWARE SPECIFICATION:-

Operating System	:	MacOS Big Sur 11.4
Platform	:	Python IDLE 2.7
Database	:	MySQL
Languages	:	Python

HARDWARE SPECIFICATION:-

Processor	:	2.2 GHz Quad-Core Intel Core i7
Hard Disk	:	256 GB
Ram	:	8 GB
Model	:	Apple MacBook Pro (Mid-2014)
Applications used	:	Terminal, MYSQL, Spyder, IDLE

DATABASE AND TABLES

1) Order Table

Field	Type	Null	Key	Default	Extra
orderid	int	NO	PRI	NULL	
orderdate	date	YES		NULL	
pcode	char(30)	NO		NULL	
pprice	float(8,2)	YES		NULL	
pqty	int	YES		NULL	
supplier	char(50)	YES		NULL	
pcat	char(30)	YES		NULL	

2) Product Table

Field	Type	Null	Key	Default	Extra
pcode	int	NO	PRI	NULL	
pname	char(30)	NO		NULL	
pprice	float(8,2)	YES		NULL	
pqty	int	YES		NULL	
pcat	char(30)	YES		NULL	

3) Sales Table

Field	Type	Null	Key	Default	Extra
salesid	int	NO	PRI	NULL	
salesdate	date	YES		NULL	
pcode	char(30)	YES		NULL	
pprice	float(8,2)	YES		NULL	
pqty	int	YES		NULL	
Total	double(8,2)	YES		NULL	

4) User Table

Field	Type	Null	Key	Default	Extra
uid	char(15)	NO	PRI	NULL	
uname	char(30)	NO		NULL	
upwd	char(30)	YES		NULL	

LIMITATIONS

1. Expensive :-

- Extremely beneficial in many aspects, this management software is available in the market at a high cost.
- Although the system provides such great features and makes the entire business a lot better and efficient, all this comes at a cost.
- Big time businesses can cover up the cost or the one time investment in some time but in the case of small or medium-sized businesses, it is at times not feasible to maintain such software.

2. Complexity :-

- Although the use of an inventory management system makes handling the inventory quite easy but learning how to operate it is quite a task.
- Learning how to effectively operate the system can be lengthy, cumbersome as well as complex.
- But once successfully installed and training completed, it can prove to be a blessing for the business and it helps a great deal in the smooth operation.

3. Limited Elimination of Risks :-

- The system helps in controlling many risks but the fear of facing and encountering many others is still open. Hence, with this system in use, many kinds of risks restricted but fail to make the entire process risk proof.

CODE

```
import os
import mysql.connector
import datetime
now = datetime.datetime.now()

print('\n')
def product_mgmt( ):
    while True :
        print('\t\t\t 1. Add New Product')
        print('\t\t\t 2. List Product')
        print('\t\t\t 3. Update Product')
        print('\t\t\t 4. Delete Product')
        print('\t\t\t 5. Main Menu')
        p=int(input('\t\t Enter Your Choice : '))

        if p ==1:
            add_product()

        if p==2:
            search_product()

        if p==3:
            update_product()

        if p==4:
            delete_product()

        if p==5 :
            break


def add_product():
    mydb = mysql.connector.connect(host='localhost', user='root', passwd='apoorv2103', database='stock')
    mycursor = mydb.cursor()
    sql = 'INSERT INTO product(pcode,pname,pprice,pqty,pcat) values (%s,%s,%s,%s,%s)'
    print('\n')
    code = int(input('\t\tEnter Product Code : '))
    search = 'SELECT count(*) FROM product WHERE pcode=%s;'
    val = (code,)
    mycursor.execute(search,val)
    for x in mycursor:
        cnt=x[0]
    if cnt==0:
        name = input('\t\tEnter Product Name : ')
        qty = int(input('\t\tEnter Product Quantity : '))
        price = float(input('\t\tEnter Product Unit Price : '))
        cat = input('\t\tEnter Product Category : ')
        val = (code,name,price,qty,cat)
        mycursor.execute(sql,val)
        mydb.commit()
    else:
        print('\t\t Product Already Exists')
```

```
def search_product():
    while True :
        print('\n')
        print('\t\t\t 1. List All Product')
        print('\t\t\t 2. List Product Code Wise')
        print('\t\t\t 3. List Product Categry Wise')
        print('\t\t\t 4. Main Menu')
        s = int(input('\t\tEnter Your Choice : '))

        if s==1 :
            list_product()
        if s==2 :
            code = int(input('Enter Product Code : '))
            list_prcat(cat)
        if s==3 :
            cat = input('Enter Category : ')
            list_prcat(cat)
        if s==4 :
            break


def list_prcode(code):
    mydb=mysql.connector.connect(host='localhost', user='root', passwd='apoorv2103', database='stock')
    mycursor=mydb.cursor()
    sql = 'SELECT * from product WHERE pcode = %s;'
    val = (code)
    mycursor.execute(sql,val)
    clrscr()
    print('\t\t\t Product Details')
    print('\t\t','-'*47)
    print('\t\t code    name  price    quantity    category')
    print('\t\t','-'*47)
    for i in mycursor:
        print('\t\t',i[0],'\t',i[1],'\t',i[2],'\t ',i[3],'\t\t',i[4])
    print('\t\t','-'*47)



def list_product():
    mydb=mysql.connector.connect(host='localhost', user='root', passwd='apoorv2103', database='stock')
    mycursor=mydb.cursor()
    sql = 'SELECT * from product'
    mycursor.execute(sql)
    clrscr()
    print('\t\t\t\t Product Details')
    print('\t','-'*60)
    print('\t code  name    price  quantity   category')
    print('\t','-'*60)
    for i in mycursor:
        print('\t',i[0],'\t',i[1],'\t',i[2],'\t',i[3],'\t',i[4])
    print('\t','-'*60)
    print('\n')
```

```
def list_prcode(code):
    mydb=mysql.connector.connect(host='localhost', user='root', passwd='apoorv2103', database='stock')
    mycursor=mydb.cursor()
    sql = 'SELECT * from product WHERE pcode = %s;'
    val = (code)
    mycursor.execute(sql,val)
    clrscr()
    print('\t\t\t Product Details')
    print('\t\t','-'*47)
    print('\t\t code    name  price    quantity    category')
    print('\t\t','-'*47)
    for i in mycursor:
        print('\t\t',i[0],'\t',i[1],'\t',i[2],'\t  ',i[3],'\t\t',i[4])
    print('\t\t','-'*47)


def list_prcat(cat):
            mydb=mysql.connector.connect(host="localhost",user="root",passwd="apoorv2103",database="stock")
            mycursor=mydb.cursor()
            sql="SELECT * from product WHERE pcat =%s"
            val=(cat,)
            mycursor.execute(sql,val)
            clrscr()
            print("\t\t\t\t PRODUCT DETAILS")
            print("\t\t","-"*47)
            print("\t\t code    name    price    quantity    category")
            print("\t\t","-"*47)
            for i in mycursor:
                    print("\t\t",i[0],"\t",i[1],"\t",i[2],"\t   ",i[3],"\t\t",i[4])
            print("\t\t","-"*47)


def update_product():
    mydb = mysql.connector.connect(host='localhost', user='root', passwd='apoorv2103', database = 'stock')
    mycursor=mydb.cursor()
    code = int(input('Enter The Product Code : '))
    qty = int(input('Enter The Quantity : '))
    price = float(input('Enter Product Price : '))
    sql = 'UPDATE product SET pqty = pqty + %s, pprice = pprice + %s WHERE pcode = %s;'
    val = (qty,code,price)
    mycursor.execute(sql,val)
    mydb.commit()
    print('\n')
    print('\t\t Product Details Updated')
    print('\n')


def delete_product():
    mydb = mysql.connector.connect(host='localhost', user='root', passwd='apoorv2103', database='stock')
    mycursor=mydb.cursor()
    code = int(input('Enter The Product Code : '))
    sql = 'DELETE FROM product WHERE pcode = %s;'
    val = (code,)
    mycursor.execute(sql,val)
    mydb.commit()
    print(mycursor.rowcount,' Record(s) Deleted ');
```

```
def purchase_mgmt( ):
    while True :
        print('\t\t\t 1. Add Order')
        print('\t\t\t 2. List Order')
        print('\t\t\t 3. Main Menu')
        o=int(input('\t\tEnter Your Choice : '))
        if o==1 :
            add_order()

        if o==2 :
            list_order()

        if o==3 :
            break


def add_order():
    mydb=mysql.connector.connect(host='localhost', user = 'root', passwd='apoorv2103', database = 'stock')
    mycursor=mydb.cursor()
    now = datetime.datetime.now()
    sql = "INSERT INTO orders(orderid,orderdate,pcode,pprice,pqty,supplier,pcat) values (%s,%s,%s,%s,%s,%s,%s)
    code = int(input('Enter Product Code : '))
    oid = now.year+now.month+now.day+now.hour+now.minute+now.second
    qty = int(input('Enter Product Quantity : '))
    price = float(input('Enter Product Unit Price : '))
    cat = input('Enter Product Category : ')
    supplier = input( 'Enter Supplier Details : ')
    val = (oid,now,code,price,qty,supplier,cat)
    mycursor.execute(sql,val)
    mydb.commit()


def list_order():
    mydb = mysql.connector.connect(host='localhost', user='root', passwd='apoorv2103', database = 'stock')
    mycursor=mydb.cursor()
    sql = 'SELECT * FROM orders'
    mycursor.execute(sql)
    clrscr()
    print('\t\t\t\t ORDER DETAILS')
    print('-'*85)
    print('orderid    Date    Product Code    Price    Quantity    Supplier    Category')
    print('-'*85)
    for i in mycursor:
        print(i[0],'\t',i[1],'\t',i[2],'\t  ',i[3],'\t',i[4],'\t   ',i[5],'\t',i[6])
    print('-'*85)
```

```
def sales_mgmt( ):
    while True :
        print('\t\t\t 1. Sale Items')
        print('\t\t\t 2. List Sales')
        print('\t\t\t 3. Main Menu')
        s=int(input('\t\tEnter Your Choice : '))
        if s==1 :
            sale_product()

        if s==2 :
            list_sale()

        if s==3 :
            break


def sale_product():
    mydb=mysql.connector.connect(host='localhost', user='root', passwd='apoorv2103', database='stock')
    mycursor=mydb.cursor()
    pcode = input('Enter Product Code : ')
    sql = 'SELECT count(*) from product WHERE pcode = %s;'
    val = (pcode,)
    mycursor.execute(sql,val)
    for x in mycursor:
        cnt=x[0]
    if cnt !=0 :
        sql = 'SELECT * from product WHERE pcode = %s;'
        val = (pcode,)
        mycursor.execute(sql,val)
        for x in mycursor:
            print(x)
            price = int(x[2])
            pqty = int(x[2])
        qty = int(input('Enter No. of Quantity : '))
        if qty <= pqty:
            total=qty*price;
            print( 'Collect Rs. ', total)
            sql = 'INSERT INTO sales values (%s,%s,%s,%s,%s,%s,%s)'
            val = (int(cnt)+1,datetime.datetime.now(),pcode,price,qty,total)
            mycursor.execute(sql,val)
            sql = 'UPDATE product SET pqty=pqty-%s WHERE pcode=%s'
            val = (qty,pcode)
            mycursor.execute(sql,val)
            mydb.commit()
        else:
            print('Quantity not Available')
```

```
def list_sale():
          mydb=mysql.connector.connect(host="localhost", user="root", passwd="apoorv2103", database="stock")
          mycursor=mydb.cursor()
          sql="SELECT * FROM sales"
          mycursor.execute(sql)
          print('\n')
          print(" \t\t\tSALES DETAILS")
          print("-"*70)
          print("Sales id    Date    Product Code    Price    Quantity    Total")
          print("-"*70)
          for x in mycursor:
              print(x[0],"\t",x[1],"\t",x[2],"\t   ",x[3],"\t",x[4],"\t",x[5])
          print("-"*70)

def user_mgmt( ):
    while True :
        print('\t\t\t 1. Add User')
        print('\t\t\t 2. List User')
        print('\t\t\t 3. Main Menu')
        u = int(input('\t\tEnter Your Choice : '))
        if u==1:
            add_user()

        if u==2:
            list_user()

        if u==3:
            break


def add_user():
          mydb=mysql.connector.connect(host="localhost",user="root",passwd="apoorv2103",database="stock")
          mycursor=mydb.cursor()
          uid=input("Enter emaid id : ")
          name=input("Enter Name : ")
          paswd=input("Enter Password :")
          sql="INSERT INTO user values (%s,%s,%s);"
          val=(uid,name,paswd)
          mycursor.execute(sql,val)
          mydb.commit()
          print(mycursor.rowcount, " user created")

def list_user():
          mydb=mysql.connector.connect(host="localhost",user="root",passwd="apoorv2103",database="stock")
          mycursor=mydb.cursor()
          sql="SELECT uid,uname from user"
          mycursor.execute(sql)
          clrscr()
          print("\t\t\tUSER DETAILS")
          print("\t\t","-"*27)
          print("\t\t UID     Name")
          print("\t\t","-"*27)
          for i in mycursor:
                    print("\t\t",i[0],"\t",i[1])
          print("\t\t","-"*27)
```

```
def db_mgmt( ):
    while True :
        print('\t\t\t 1. Database Creation')
        print('\t\t\t 2. List Database')
        print('\t\t\t 3. Main Menu')
        p = int(input('\t\tEnter Your Choice : '))
        if p==1 :
            create_database()
        if p==2 :
            list_database()
        if p==3 :
            break

def create_database():
    mydb = mysql.connector.connect(host='localhost', user='root', passwd='apoorv2103', database='stock')
    mycursor=mydb.cursor()
    print('Creating PRODUCT Table ')
    sql = "CREATE TABLE product (\
           pcode int(4) PRIMARY KEY,\
           pname char(30) NOT NULL,\
           pprice float(8,2),\
           pqty int(4),\
           pcat char(30));"
    mycursor.execute(sql)
    print("Creating ORDER Table ")
    sql = "CREATE TABLE orders (\
           orderid int(4) PRIMARY KEY, \
           orderdate DATE, \
           pcode char(30) NOT NULL, \
           pprice float(8,2), \
           pqty int(4), \
           supplier char(50), \
           pcat char(30));"
    mycursor.execute(sql)
    print("ORDER table created ")

    print('Creating Sales Table ')
    sql = "CREATE TABLE sales (\
           salesid int(4) PRIMARY KEY, \
           salesdate DATE, \
           pcode char(30) references product(pcode), \
           pprice float(8,2), \
           pqty int(4), \
           Total double(8,2)\
           );"
    mycursor.execute(sql)
    print( 'SALES table created ')
    sql = "CREATE TABLE user (\
           uid char(15) PRIMARY KEY, \
           uname char(30) NOT NULL, \
           upwd char(30));"
    mycursor.execute(sql)
    print( 'USER table created' )
```

```
def list_database():
    mydb = mysql.connector.connect(host="localhost", user="root", passwd="apoorv2103", database='stock')
    mycursor = mydb.cursor()
    sql='show tables;'
    mycursor.execute(sql)
    for i in mycursor:
        print(i)

def clrscr():
    print('\n'*5)

print('\n')
print('       Welcome to The Stock Management System - A Project by Apoorv Thite')
print('\n')
while True :
    print('\t\t\t STOCK MANAGEMENT')
    print('\t\t\t *****************\n')
    print('\t\t\t 1. PRODUCT MANAGEMENT')
    print('\t\t\t 2. PURCHASE MANAGEMENT')
    print('\t\t\t 3. SALES MANAGEMENT')
    print('\t\t\t 4. USER MANAGEMENT')
    print('\t\t\t 5. DATABASE SETUP')
    print('\t\t\t 6. Exit\n')
    n=int(input('   Enter Your Choice : '))
    if n==1:
        product_mgmt()
    if n==2:
        os.system('cls')
        purchase_mgmt()
    if n==3:
        sales_mgmt()
    if n==4:
        user_mgmt()
    if n==5:
        db_mgmt()
    if n==6:
        break
print('\n')
```

OUTPUT

```
    Welcome to The Stock Management System - A Project by Apoorv Thite


                        STOCK MANAGEMENT
                        ****************

                        1. PRODUCT MANAGEMENT
                        2. PURCHASE MANAGEMENT
                        3. SALES MANAGEMENT
                        4. USER MANAGEMENT
                        5. DATABASE SETUP
                        6. Exit

Enter Your Choice : 1
                        1. Add New Product
                        2. List Product
                        3. Update Product
                        4. Delete Product
                        5. Main Menu
                Enter Your Choice : 1


            Enter Product Code : 1578
            Enter Product Name : File
            Enter Product Quantity : 10
            Enter Product Unit Price : 50
            Enter Product Category : Book
```

```
                        1. Add New Product
                        2. List Product
                        3. Update Product
                        4. Delete Product
                        5. Main Menu
                Enter Your Choice : 2


                        1. List All Product
                        2. List Product Code Wise
                        3. List Product Categry Wise
                        4. Main Menu
                Enter Your Choice : 1




                                Product Details
    ------------------------------------------------------------------
    code  name    price  quantity   category
    ------------------------------------------------------------------
    1234    Hat     5.0     5       Random
    1543    Car     70.0    5       Toy
    1578    File    50.0    10      Book
    2131    Ball    20.0    5       Sports
    2211    Bat     1.0     1       3
    ------------------------------------------------------------------
```

```
                        STOCK MANAGEMENT
                        ****************

                        1. PRODUCT MANAGEMENT
                        2. PURCHASE MANAGEMENT
                        3. SALES MANAGEMENT
                        4. USER MANAGEMENT
                        5. DATABASE SETUP
                        6. Exit

   Enter Your Choice : 2
                        1. Add Order
                        2. List Order
                        3. Main Menu
                Enter Your Choice : 1
Enter Product Code : 4986
Enter Product Quantity : 5
Enter Product Unit Price : 20
Enter Product Category : Toy
Enter Supplier Details : Pune
                        1. Add Order
                        2. List Order
                        3. Main Menu
                Enter Your Choice : 2




                                ORDER DETAILS
-------------------------------------------------------------------------------------------
orderid    Date    Product Code    Price    Quantity    Supplier    Category
-------------------------------------------------------------------------------------------
2149    2022-01-31      4986      20.0   5          Pune          Toy
-------------------------------------------------------------------------------------------
```

```
    Welcome to The Stock Management System - A Project by Apoorv Thite


                        STOCK MANAGEMENT
                        ****************

                        1. PRODUCT MANAGEMENT
                        2. PURCHASE MANAGEMENT
                        3. SALES MANAGEMENT
                        4. USER MANAGEMENT
                        5. DATABASE SETUP
                        6. Exit

   Enter Your Choice : 3
                        1. Sale Items
                        2. List Sales
                        3. Main Menu
                Enter Your Choice : 1
Enter Product Code : 1234
(1234, 'Hat', 5.0, 5, 'Random')
Enter No. of Quantity : 2
Collect Rs.  10
```

```
                        STOCK MANAGEMENT
                        ****************

                        1. PRODUCT MANAGEMENT
                        2. PURCHASE MANAGEMENT
                        3. SALES MANAGEMENT
                        4. USER MANAGEMENT
                        5. DATABASE SETUP
                        6. Exit

   Enter Your Choice : 4
                        1. Add User
                        2. List User
                        3. Main Menu
                Enter Your Choice : 1
Enter emaid id : 21
Enter Name : Apoorv
Enter Password :apoorvT
1  user created
                        1. Add User
                        2. List User
                        3. Main Menu
                Enter Your Choice : 2




                        USER DETAILS
                -------------------------
                UID     Name
                -------------------------
                21      Apoorv
                -------------------------
```

```
                        STOCK MANAGEMENT
                        ****************

                        1. PRODUCT MANAGEMENT
                        2. PURCHASE MANAGEMENT
                        3. SALES MANAGEMENT
                        4. USER MANAGEMENT
                        5. DATABASE SETUP
                        6. Exit

   Enter Your Choice : 5
                        1. Database Creation
                        2. List Database
                        3. Main Menu
                Enter Your Choice : 2
('orders',)
('product',)
('sales',)
('user',)
```

BIBLIOGRAPHY

- wikipedia.com
- slideshare.com
- pythonworld.in
- Sumita Arora Book for IP Class 12
- NCERT IP Class 12 Textbook
- w3schools.com