RD INTERNATIONAL SCHOOL, ERODE.

COMPUTER INVESTIGATORY PROJECT

NURSERY STORE MANAGEMENT

Guided by: Mrs.KAVITHA.L, PGT (Computer Science)

Done by: KISHORE .P .P XII-B,

AKHIIL DHEEP .B XII-B

CERTIFICATE

CERTIFICATE

This is to certify that the project entitled "**NURSERY STORE MANAGEMENT ''** is a record of bonafide work carried out by K**ISHORE.P.P [XII-B]** and **AKHIIL DHEEP.B [XII-B]**. In partial fulfillment of the requirements in **COMPUTER SCIENCE** prescribed by CBSE for 2022 – 2023 in **RD INTERNATIONAL SCHOOL, GATEPUDUR, ERODE.**

DATE **PRINCIPAL**

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

We wish to express our sincere thanks to our beloved Founder Chairman **Dr. D.SENTHIL KUMAR**, our Chairman **Mr. S. RAGHUL,** our Principal **Mr. R.SHANKAR** and our institution **RD International School, Erode** for guiding and providing facilities towards the successful outcome of this project work.

We wish to express our deep and profound sense of gratitude to our guide teacher **Mrs. KAVITHA.L, PGT (Computer Science)**, for her expert help, valuable guidance, comments and suggestions.

We express my heartfelt gratitude to my parents for constant encouragement while carrying out this project.

We also express our sincere gratitude to one and all who directly or indirectly, have lent their helping hand in this venture.

CONTENTS

CONTENTS

AIM

AIM

Our project is based on developing a software package for Nursery Store Management to create and maintain records easily. The main notion of this project is to have hassle free storage of records and for easy access of records. This software saves time and also it ensures security of records stored in the database.

INTRODUCTION TO PYTHON

INTRODUCTION TO PYTHON

Python is a widely used general-purpose, high level programming language. It was initially designed by ***Guido van Rossum*** in ***1991*** and developed by **Python Software Foundation**. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code. It is used for

- Web development (server-side)
- Software development
- Mathematics
- System scripting

BENEFITS OF PYTHON:

- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.
- Python can be used for rapid prototyping, or for production-ready software development.
- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
- Python has a simple syntax similar to the English language.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.

- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
- Python can be treated in a procedural way, an object-oriented way or a functional way.
- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.
- In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing large collections of Python files.
- Python was designed for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes.

INTRODUCTION TO THE PROJECT

INTRODUCTION TO THE PROJECT

INTRODUCTION

This is a free software designed for easy purchase at Nursery stores. This software is a link between the Manager, Staff and the Customers. It is a user-friendly software provided with security. It allows the manager to update and delete their records. The whole details of the customer can be inserted, deleted and revealed only to the manager so the customer's security is also ensured. The user is also allowed to check, update and manage through this software.

OBJECTIVES OF THE PROJECT

- ★ This software helps the managers of a plant store.
- ★ Managers can compute details of the customers who bought products and expunge the details of the staff who resigned.
- ★ Here the customers can buy by viewing all the plant details.
- ★ And also it has an option on details of the plants according to the selected import location and variety.
- ★ Managers also have options to check their total margin obtained after sales of plants accordingly.

REQUIREMENT

REQUIREMENTS

HARDWARE REQUIRED:

- Processor : (ANY)
- RAM : 512MB+
- Hard disk : ANY HARD DISK WITH 256 GB & ABOVE
- MOTHERBOARD : 1.845 OR 915,995 FOR PENTIUM 0R MSI K9MM-V VIA K8M800+8237R PLUS
- A Suitable PC

  (or)
- MONITOR 14.1 or 15 -17 inch
- Keyboard and mouse
- Printer : (if print is required – [Hard copy])

SOFTWARE REQUIRED:

- Operating system : WINDOWS 7,8,10,11/LINUX/MAC OS
- Python 3.6 with mysql.connector package

PROJECT ANALYSIS

PROJECT ANALYSIS

- The project "**NURSERY STORE MANAGEMENT**" has 8 modules.
    1. Add Plant record
    2. Buy Plants
    3. Search Plants
    4. Staff Details
    5. Sales Record
    6. Available Plants
    7. Total Income
    8. Signing in

- We have initially created an interface between Python and MySQL by installing mysql.connector package to our project interpreter to ease the accessibility.
- We have used various tables to store the user data. The various tables used to store the user data are:

    - Database:
        - Nursery Database
    - Tables:
        - Available_plants
        - Sell_rec
        - SignUp
        - staff_details

CODE

```
import mysql.connector as m

con = m.connect(host="localhost", user="root", password="root",
auth_plugin='mysql_native_password')

# CREATING DATABASE AND TABLE
cur = con.cursor()
cur.execute("create database if not exists nursery")
cur.execute("use nursery")
cur.execute("create table if not exists signup(username
varchar(20),password varchar(20))")

while True:
   ch = int(input("SIGN UP [1] ; LOGIN [2] : "))
   # SIGNUP
   if ch == 1:
       username = input("USERNAME: ")
       pw = input("PASSWORD: ")
       cur.execute("insert into signup values('" + username +
"','" + pw + "')")
       con.commit()
   # LOGIN
   elif ch == 2:
       username = input("USERNAME : ")
       cur.execute("select username from signup where
username='" + username + "'")
       pot = cur.fetchone()
       if pot is not None:
           pw = input("PASSWORD 👁 : ")
           cur.execute("select password from signup where
password='" + pw + "'")
           a = cur.fetchone()
           if a is not None:
               print("""LOGIN SUCCESSFUL""")
print("""=============================================================
============================================================
++++++++++++++++++++++++++++++++++++++++++    MY PLANT
NURSERY     ++++++++++++++++++++++++++++++++++++++++++
============================================================
============================================================""")
               cur.execute(
                   "create table if not exists
Available_Plants(Plant_Name varchar(30) primary key,Variety
varchar("
                   "20),Quantity int(3),Import_Location
varchar(30),Price int(4))")
```

```
cur.execute(
                    "create table if not exists
Sell_rec(CustomerName varchar(20),Phone_Number char(10) unique
key, "
                    "Plant_Name varchar(30),Quantity
int(100),Price int(4),foreign key (Plant_Name) references "
                    "Available_Plants(Plant_Name))")
                cur.execute(
                    "create table if not exists
Staff_details(Name varchar(30), Gender varchar(10),Age int(3),
"
                    "Phone_Number char(10) unique key , Address
varchar(40))")
                con.commit()

                while True:
                    print("""1:Add Plant record
2:Buy Plants
3:Search Plants
4:Staff Details
5:Sales Record
6:Available Plants
7:Total Income after the Latest Reset
8:Exit""")

                    a = int(input("Enter your choice:"))


                    # ADD PLANTS
                    if a == 1:
                        plant = str(input("Plant Name: "))
                        variety = str(input("Variety: "))
                        quantity = int(input("Quantity: "))
                        import_location = str(input("Plant
import location: "))
                        price = int(input("Enter the price: "))

                        cur.execute("select * from
Available_Plants where Plant_Name ='" + plant + "'")
                        row = cur.fetchone()

                        if row is not None:
                            cur.execute("update Available_Plants
set quantity=quantity+'" + str(
                                quantity) + "' where
Plant_Name='" + plant + "'")
                            con.commit()
```

```
print("""++++++++++++++++++++
++SUCCESSFULLY ADDED++
++++++++++++++++++++""")

                        else:
                            cur.execute(
                                "insert into
Available_Plants(Plant_Name,variety,quantity,Import_Location,"
                                "price) values('" + plant +
"','" + variety + "','" + str(
                                    quantity) + "','" +
import_location + "','" + str(price) + "')")
                            con.commit()

                            print("""++++++++++++++++++++
++SUCCESSFULLY ADDED++
++++++++++++++++++++""")


                            # SELL PLANT
                    elif a == 2:

                        print("AVAILABLE PLANTS...")

                        cur.execute("select * from
Available_Plants ")
                        for x in cur:
                            print(x)


                        customer_name = str(input("Enter
customer name:"))
                        phone_no = int(input("Enter phone
number:"))
                        plant = str(input("Enter Plant Name:"))
                        price = int(input("Enter the price:"))
                        n = int(input("Enter quantity:"))

                        cur.execute("select quantity from
available_plants where Plant_Name='" + plant + "'")
                        lk = cur.fetchone()

                        if max(lk) < n:
                            print(n, "Plants are not
available!!!!")
                        else:
                            cur.execute("select Plant_Name from
available_plants where Plant_Name='" + plant + "'")
                            log = cur.fetchone()
                            if log is not None:
```

```
 cur.execute("insert into Sell_rec values('" + customer_name +
"','" + str(
                                    phone_no) + "','" + plant +
"','" + str(n) + "','" + str(price) + "')")
                                cur.execute("update
Available_plants set quantity=quantity-'" + str(
                                    n) + "' where Plant_Name='"
+ plant + "'")
                                con.commit()


                                print("""++++++++++++++++++++
++PLANT HAS BEEN SOLD++
++++++++++++++++++++""")

                            else:
                                print("PLANT IS NOT
AVAILABLE!!!!!!!")

                    # SEARCH PLANTS ON THE BASIS OF GIVEN
OPTIONS
                    elif a == 3:

                        print("""1:Search by name
2:Search by variety
3:Search by Import Location""")

                        search = int(input("Search by?:"))

                        # BY PLANT NAME
                        if search == 1:
                            plant_search = input("Enter Plant to
search:")

                            cur.execute(
                                "select Plant_Name from
available_plants where Plant_Name='" + plant_search + "'")
                            tree = cur.fetchone()

                            if tree is not None:
                                print("""++++++++++++++++++
++PLANT IS IN STOCK++
++++++++++++++++++""")

                            else:
                                print("PLANT IS NOT IN
STOCK!!!!!!!")
```

```
        # BY VARIETY
                        elif search == 2:
                            variety_search = input("Enter
variety to search:")
                            cur.execute("select variety from
available_plants where variety='" + variety_search + "'")
                            poll = cur.fetchall()
                            if poll is not None:
                                print("""++++++++++++++++++++
++PLANT IS IN STOCK++
++++++++++++++++++++""")
                                cur.execute("select * from
available_plants where variety='" + variety_search + "'")

                                for y in cur:
                                    print(y)


                            else:
                                print("PLANTS OF SUCH VARIETY
ARE NOT AVAILABLE!!!!!!!!!")

                        # BY IMPORT LOCATION
                        elif search == 3:
                            location_search = input("Enter
Import location to search:")


                            cur.execute(
                                "select Import_Location from
available_plants where Import_Location='" + location_search +
"'")
                            home = cur.fetchall()


                            if home is not None:
                                print("""++++++++++++++++++++
++PLANT IS IN STOCK++
++++++++++++++++++++""")


                                cur.execute(
                                    "select * from
available_plants where Import_Location='" + location_search +
"'")
                                for z in cur:
                                    print(z)
                            else:
                                print("PLANTS FROM THIS LOCATION
ARE NOT AVAILABLE!!!!!!!")
                        con.commit()
```

OUTPUT

OUTPUT

```
SIGN UP [1] ; LOGIN [2] : 1

USERNAME: Akhiil

PASSWORD: root

SIGN UP [1] ; LOGIN [2] : 2

USERNAME : Akhiil

PASSWORD 👁️ : root

LOGIN SUCCESSFUL

============================================================
==========================================================

++++++++++++++++++++++++++++++++++++++++++++     MY  PLANT
NURSERY      ++++++++++++++++++++++++++++++++++++++++++++

============================================================
==========================================================

1:Add Plant record

2:Buy Plants

3:Search Plants

4:Staff Details

5:Sales Record

6:Available Plants

7:Total Income after the Latest Reset

8:Exit

Enter your choice:1

Plant Name: Oakleaf

Variety: Snowflake

Quantity: 100

Plant import location: North America

Enter the price: 240


++SUCCESSFULLY ADDED++
```

```
1:Add Plant record
2:Buy Plants
3:Search Plants
4:Staff Details
5:Sales Record
6:Available Plants
7:Total Income after the Latest Reset
8:Exit
Enter your choice:1
Plant Name: Palm
Variety: Queen
Quantity: 100
Plant import location: South America
Enter the price: 480
++SUCCESSFULLY ADDED++
1:Add Plant record
2:Buy Plants
3:Search Plants
4:Staff Details
5:Sales Record
6:Available Plants
7:Total Income after the Latest Reset
8:Exit
Enter your choice:2
AVAILABLE PLANTS...
('Oakleaf', 'Snowflake', 100, 'North America', 240)
('Palm', 'Queen', 100, 'South America', 480)
Enter customer name:Kishore
Enter phone number:9444616745
Enter Plant Name:Oakleaf
```

```
Enter the price:240
Enter quantity:3
++++++++++++++++++++
++PLANT HAS BEEN SOLD++
++++++++++++++++++++
1:Add Plant record
2:Buy Plants
3:Search Plants
4:Staff Details
5:Sales Record
6:Available Plants
7:Total Income after the Latest Reset
8:Exit
Enter your choice:5
1:Sales Record
2:Reset Sales Record
Enter your choice:1
('Kishore', '9444616745', 'Oakleaf', 3, 240)
1:Add Plant record
2:Buy Plants
3:Search Plants
4:Staff Details
5:Sales Record
6:Available Plants
7:Total Income after the Latest Reset
8:Exit
Enter your choice:7
(Decimal('240'),)
```

SUGGESTED IMPROVEMENTS

SUGGESTED IMPROVEMENT

- A proper e-commerce integration to the software would help customers in choosing much more products
- The shop could also provide a user buying guide for customers
- Availability of Google, Facebook or Apple sign in to create an account in the program.
- This is an offline program, in the future, it can be developed further as an online application too.
- Online payment of the buying expenses can be done with our intelligence software.
- And to make the usage of the software at ease and inclusion of Graphic User Interface (GUI) for more elegant and interactive software experience.

BIBLIOGRAPHY

BIBLIOGRAPHY

BOOKS REFERRED:

- ➔ Computer Science Textbook for Class XII (NCERT)
- ➔ Computer science With Python - Class XII by Sumita Arora

WEBSITES REFERRED:

- ★ https://www.python.org
- ★ https://stackoverflow.com/questions/766/python-and-mysql
- ★ https://mysqlcode.com/