PM SHRI KENDRIYA VIDYALAYA DL MEERUT

COMPUTER SCIENCE (083)

PROJECT REPORT-STUDENT MANAGEMENT SYSTEM

SESSION-2023-24

SUBMITTED BY: Divyay Agarwal

Declaration

I,**Divyay Agarwal,**student of Class XII-Science at PM Shri Kendriya Vidyalaya DL Meerut hereby declare that the project work entitled "**Student Management system**" submitted for the Senior Secondary Supplementary Examination 2023-24 of Computer Science (Code-083) has been prepared by me.

NAME: Divyay Agarwal
ROLL NO:

PM SHRI KENDRIYA VIDYALAYA DL MEERUT

CERTIFICATE

This is to certify that the project report entitled **"STUDENT MANAGEMENT SYSTEM"** has been prepared and submitted by **Divyay Agarwal**, of Class XII-A ,**PM Shri Kendriya Vidyalaya DL Meerut** for the Senior Secondary Supplementary Examination 2023-24 of Computer Science (Code-083).

Signature of teacher

Name: Mr.Sandeep Kumar
Designation: PGT-CS

ACKNOWLEDGEMENT

I would like to express a deep sense of thanks & gratitude to our Project guide **Mr Sandeep Kumar, PGT – Computer Sc., PM Shri KVDL Meerut** for guiding us immensely through the course of the project. His constructive advice & constant motivation have been responsible for the successful completion of this project.

We would also like to thank our parents for their constant motivation & support. We want to thank our class mates for their timely help & support for completion of this project. Last but not the least We would like to thanks all those who had helped directly and indirectly towards the completion of this project.

NAME: Divyay Agarwal

ROLL NO:

CONTENTS

PYTHON BASICS

PYTHON is a general purpose interpreted, simple, interactive, object-oriented, and high-level programming language.It was developed by Guido von Rossum in 1991. It is widely used for applications such as web development, machine learning, artificial intelligence etc. It is an interpreted language as its programs are executed by an interpreter.

Python features: It's some of the key features include:

Easy to Read and Write: Python has a simple syntax that is easy to read and write and can be easily interpreted. It uses indentation and whitespace to define code blocks due to which it is easy to work on.

Database supporter: It supports Relational database systems such as MySQL etc. Through mysql.coonector module, we can easily interface python with MySQL.

Object-Oriented: Python supports object-oriented programming. It allows you to define classes, create objects etc.

Large Standard Library: Python comes with a comprehensive standard library that provides a wide range of modules and functions for various tasks, such as file handling, networking, database access, and more.

Cross-Platform Language: Python is a cross-platform language, meaning that Python programs can run on different operating systems, including Windows, macOS, Linux, and more. This makes it highly portable and versatile.

It also supports SQL cursors.

MySQL Basics

MySQL is a popular open-source and freely available relational database management system (RDBMS) that uses Structured Query Language(SQL) for creating, storing, maintaining and accessing data,stored in the form of databases and their respective tables. It is known for its reliability, ease of use and other features.

Features of MySQL: Some of it's features are:

Easy to Use: MySQL is user-friendly and has a simple interface, making it easy for developers and administrators to work with.

Database handling: It can handle large-scale databases and high-traffic websites easily.

Cross-Platform Compatibility: It is platform-independent and can run on various operating systems such as Windows, macOS, Linux, and Unix etc.

Data Security: It provides security features to protect your data. A user needs to enter password to access his databases.

Python Connectivity: It can be easily interfaced with python.

INTRODUCTION OF THE PROJECT

The **STUDENT MANAGEMENT SYSTEM** is a Python-based software application designed to streamline the entire student record management process. This project would address the current challenges and provide a unified platform for managing student information. Also, This project underscores the significance of software solutions in education, where efficient record-keeping can lead to improved administrative processes and, ultimately, better support for students' academic journeys.

The primary reason behind choosing this topic is to create a database like program which could help schools in managing student data and could easily replace the outdated student record management system.

For example: XYZ High School, a reputable educational institution, is facing problem with an outdated and cumbersome student record management system. The current system, primarily reliant on manual paperwork and decentralized data storage, is causing numerous administrative hurdles and inefficiencies due to which it is creating many difficulties. So, to overcome this problem, the school is looking for an effective solution that can replace this outdated system.

Modules used in the project

- Mysql.connector

Table used - student 1

Database used - school

HARDWARE/SOFTWARE CONFIGURATIONS

Hardware configuration:

- x64 based PC with Intel core i3 processor and RAM: 4 GB

Software configuration:

OS: Microsoft windows 10

- Python 3.12.0 for front end development
- MySQL 8.0.35 for back end data storage and management
- MSWord for Documentation

project name: - Student Management System

Source code:-

```
import mysql.connector

mydb = mysql.connector.connect(
    host="localhost",user="root",
     passwd="Div81624",
     database="student")
mycursor = mydb.cursor()

def main_menu():
      ch = 'y'
      while ch == 'y':
          print("Student management System")
          print("1. Records management")
          print("2. Result management")
          choice = input("enter your choice:")
          if choice == '1':
              adrec()
          elif choice == '2':
              acdrec()
          else:
              print("wrong input.")
          ch = input("Do you want to
                     continue?(y/n)")
  def adrec():
      n = "y"
      while n == "y":
```

```
print("Record management")
print("1.To create a new table")
print("2.To show existing tables")
print("3.To describe structure")
print("4.To add the record of a new
          student")
print("5.To delete a record")
print("6.To view record of a student")
print("7.To alter a record")
print("8.To view all records")
print("9.To QUIT")
ch = input("enter your choice:")
if ch == '1':
    create_table()
elif ch == '2':
    show_table()
elif ch == '3':
    desc_table()
elif ch == '4':
    newStudent()
elif ch == '5':
    deleteSturec()
elif ch == '6':
    viewrec()
elif ch == '7':
    updateStudent()
elif ch == '8':
   displayStudent()
elif ch == '9':
   print("EXITING")
```

```
            n = input("Do you want to continue
          to main menu?(y/n)")
            if n == "y" or "Y":
               main_menu()
            if n == "n" or "N":
                break
def create_table():
    createTable = """CREATE TABLE IF NOT EXISTS
student (SROLL_NO VARCHAR(5),
        SNAME VARCHAR(50),
        FNAME VARCHAR(50),
        MNAME VARCHAR(50),
        PHONE CHAR(10),
        ADDRESS VARCHAR(100),
        SCLASS VARCHAR(5),
        SSECTION VARCHAR(5),
        SADMISSION_NO VARCHAR(10) PRIMARY KEY)"""
    mycursor.execute(createTable)
    mycursor.execute("COMMIT")
    print("Table created Successfully")

def show_table():
    mycursor.execute("SHOW TABLES")
    for x in mycursor:
        print(x)
    if mycursor.execute == "" :
        print("Error")

def desc_table():
    mycursor.execute("DESCRIBE student")
    for x in mycursor:
```

```
        print(x)
    if mycursor.execute == "" :
        print("Error")

def newStudent():
    sroll_no = input("ENTER ROLL_NO : ")
    sname = input("\n ENTER STUDENT'S NAME : ")
    fname = input(" ENTER FATHER'S NAME : ")
    mname = input(" ENTER MOTHER'S NAME : ")
    phone = input(" ENTER CONTACT NO. : ")
    address = input(" ENTER ADDRESS : ")
    sclass = input(" ENTER CLASS : ")
    ssection = input(" ENTER SECTION : ")
    sadmission_no = input(" ENTER ADMISSION_NO   :
    ")
    sql = "INSERT INTO student
(sroll_no,sname,fname,mname,phone,address,sclass,
ssection,sadmission_no)
VALUES(%s,%s,%s,%s,%s,%s,%s,%s,%s)"
    values = (sroll_no, sname, fname, mname,
phone, address, sclass, ssection, sadmission_no)
    mycursor.execute(sql, values)
    mycursor.execute("COMMIT")

def deleteSturec():
     adm_no=input("enter the admission number of
student.")
     sql="DELETE FROM student WHERE
sadmission_no=%s"
     mycursor.execute(sql,(adm_no,))
     mycursor.execute("commit")
```

```
        print("record deleted")

    def viewrec():
        adm_no = input("Enter the admission no. of
    student: ")
        sql = "SELECT * FROM STUDENT WHERE
    sadmission_no = %s"
        mycursor.execute(sql, (adm_no,))
        data = mycursor.fetchone()
        if data is not None:
            for x in data:
                print(x)
        else:
            print("No record")

    def displayStudent():
        mycursor.execute("SELECT * FROM student")
        data = mycursor.fetchall()
        for x in data:
            print(x)



    def updateStudent():
        admission_no = input("ENTER ADMISSION NO :")
        sql = "SELECT * FROM student WHERE
    sadmission_no= %s"
        mycursor.execute(sql, (admission_no,))
        data = mycursor.fetchall()

        print("PRESS 1 FOR NAME")
        print("PRESS 2 FOR CLASS")
```

```
    print("PRESS 3 FOR ROLL NO")
    print("PRESS 4 FOR OTHER OPTIONS")
    choice = int(input("Enter Your Choice"))
    if choice == 1:
        name = input("ENTER NAME OF THE
STUDENT  :")
        sql = "UPDATE student SET sname= %s WHERE
sadmission_no =%s"
        mycursor.execute(sql, (name,
admission_no))
        mycursor.execute("COMMIT")
        print("NAME UPDATED")
    elif choice == 2:
        std = input("ENTER CLASS OF THE
STUDENT   :")
        sql = "UPDATE student SET sclass= %s
WHERE sadmission_no=%s"
        mycursor.execute(sql, (std, admission_no))
        mycursor.execute("COMMIT")
        print("CLASS UPDATED")
    elif choice == 3:
      roll_no = int(input("ENTER ROLL NO OF THE
STUDENT:"))
      sql="UPDATE student  SET  sroll_no=    %s
     WHERE sadmission_no = %s"
      mycursor.execute(sql,(roll_no,admission_no))
      mycursor.execute("COMMIT")
      print("ROLL NO UPDATED")
    elif choice==4:
      mycursor.execute("DESCRIBE STUDENT")
      for x in mycursor:
```

```
            print(x)
        c=input("Enter the field to be changed")
        newinfo=input("Enter updated information.")
        admission_no = input("Enter your admission
no.")
        sql="UPDATE student SET %s=%s WHERE
sadmission_no = %s"
  mycursor.execute(sql,(c,newinfo,admission_no))
        mycursor.execute("commit")
        print("RECORD UPDATED")
def acdrec():
    print("RESULT MANAGEMENT")
    print("1. To create table")
    print("2. Describe structure")
    print("3. To add student record")
    print("4. To delete student record")
    print("5. To view report card of a student.")
    print("6. To view overall report.")
    print("7. TO QUIT")

    while True:
        choice = input("Enter your choice: ")

        if choice == "1":
            create_r_table()
        elif choice == "2":
            desc_r_table()
        elif choice == "3":
            add_r_record()
        elif choice == "4":
            del_r_record()
```

```
        elif choice == "5":
            r_card_one_stu()
        elif choice == "6":
            r_card_all()
        elif choice == "7":
            print("EXITING")
            n = input("Do you want to continue to
the main menu? (y/n): ")
            if n == "y" or "Y":
                main_menu()
            if n == "n" or "N":
                 break

def create_r_table():
    create_table = """CREATE TABLE IF NOT EXISTS
MARKS(  SADMISSION_NO VARCHAR(10) PRIMARY KEY,
        SNAME VARCHAR(20),
        HINDI INT,
        ENGLISH INT,
        MATH INT,
        SCIENCE INT,
        SOCIAL INT,
        COMPUTER INT,
        TOTAL INT,
        AVERAGE DECIMAL )"""
    mycursor.execute(create_table)
    mycursor.execute("COMMIT")
    print("Table created successfully.")

def desc_r_table():
    mycursor.execute("DESCRIBE MARKS")
```

```
    for x in mycursor:
        print(x)
def add_r_record():
    admission_no = input("ENTER ADMISSION NO OF
THE STUDENT: ")
    name = input("ENTER NAME OF THE STUDENT")
    hindi = int(input("ENTER MARKS OF HINDI: "))
    english = int(input("ENTER MARKS OF ENGLISH:
"))
    math = int(input("ENTER MARKS OF MATH: "))
    science = int(input("ENTER MARKS OF SCIENCE:
"))
    social = int(input("ENTER MARKS OF SOCIAL: "))
    computer = int(input("ENTER MARKS OF COMPUTER:
"))
    total = hindi + english + math + science +
social + computer
    average = total / 6

    sql = "INSERT INTO MARKS(SADMISSION_NO,SNAME,
HINDI, ENGLISH, MATH, SCIENCE, SOCIAL, COMPUTER,
TOTAL, AVERAGE)
VALUES(%s,%s,%s,%s,%s,%s,%s,%s,%s,%s)"
    values = (admission_no,name, hindi, english,
math, science, social, computer, total, average)

    mycursor.execute(sql, values)
    mycursor.execute("COMMIT")
    print("\nMarks of the Student Entered
Successfully!")
```

```
def del_r_record():
    adm_no = input("Enter the admission number: ")
    sql = "DELETE FROM MARKS WHERE SADMISSION_NO
= %s"
    mycursor.execute(sql, (adm_no,))
    mycursor.execute("COMMIT")
    print("Record deleted successfully")

def r_card_all():
    mycursor.execute("SELECT * FROM MARKS")
    data = mycursor.fetchall()
    c = {}
    for x in data:
        c["AdmNo."] = x[0]
        c["Name:"] = x[1]
        c["Hindi:"] = x[2]
        c["English:"] = x[3]
        c["Maths:"] = x[4]
        c["Science:"] = x[5]
        c["S.ST"] = x[6]
        c["Computer"] = x[7]
        c["Total"] = x[8]
        c["Average"] = x[9]
        print(c)

def r_card_one_stu():
    admission_no = input("ENTER ADMISSION NO OF
THE STUDENT: ")
    sql = "SELECT * FROM MARKS WHERE
SADMISSION_NO = %s"
    mycursor.execute(sql, (admission_no,))
```

```
    data = mycursor.fetchall()
    if data is not None:
        for x in data:
            print(x)
    else:
        print("No record")
# Call the main menu function to start the
program
main_menu()
```

Output Screens

```
Student management System
1. Records management
2. Result management
enter your choice:

Record management
1.To create a new table
2.To show existing tables
3.To describe structure
4.To add the record of a new student
5.To delete a record
6.To view record of a student
7.To alter a record
8.To view all records
9.To QUIT
enter your choice:1
Table created Successfully
enter your choice:2
('marks',)
('student',)
enter your choice:3
('SROLL_NO', 'varchar(5)', 'YES', '', None, '')
('SNAME', 'varchar(50)', 'YES', '', None, '')
('FNAME', 'varchar(50)', 'YES', '', None, '')
('MNAME', 'varchar(50)', 'YES', '', None, '')
('PHONE', 'char(10)', 'YES', '', None, '')
('ADDRESS', 'varchar(100)', 'YES', '', None, '')
('SCLASS', 'varchar(5)', 'YES', '', None, '')
('SSECTION', 'varchar(5)', 'YES', '', None, '')
('SADMISSION_NO', 'varchar(10)', 'NO', 'PRI', None, '')
enter your choice:4
ENTER ROLL_NO : 11

 ENTER STUDENT'S NAME : Divyay
 ENTER FATHER'S NAME : Mr.Sandeep
 ENTER MOTHER'S NAME : Ms.Jyoti
 ENTER CONTACT NO. : 7895XXXX
 ENTER ADDRESS : B-60,Chankyapuri
 ENTER CLASS : 12th
 ENTER SECTION : A
 ENTER ADMISSION_NO   :   86XX
```

```
enter your choice:6
Enter the admission no. of student: 8644
11
Divyay
Mr.Sandeep Kumar
Ms.Jyoti
7895XXXX
Lakhmi Vihar
12
A
8644
enter your choice:7
ENTER ADMISSION NO :8644
PRESS 1 FOR NAME
PRESS 2 FOR CLASS
PRESS 3 FOR ROLL NO
PRESS 4 FOR OTHER OPTIONS
Enter Your Choice
enter your choice:8
('11', 'Divyay', 'Mr.Sandeep Kumar', 'Ms.Jyoti ', '7895XXXX', 'Lakhmi Vihar', '12', 'A', '8644')
('11', 'Divyay', 'Mr.Sandeep', 'Ms.Jyoti', '7895XXXX', 'B-60,Chankyapuri', '12th', 'A', '86XX')
```

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sakila             |
| school             |
| student            |
| sys                |
| world              |
+--------------------+
8 rows in set (0.00 sec)
```

```
mysql> select * from student;
+----------+-------+-------+-------+-------+-------------+--------+----------+--------------+
| SROLL_NO | SNAME | FNAME | MNAME | PHONE | ADDRESS     | SCLASS | SSECTION | SADMISSION_NO |
+----------+-------+-------+-------+-------+-------------+--------+----------+--------------+
| 11       | Div   | sand  | jyot  | 2345  | lakhmi vihar | 12     | A        | 8644          |
+----------+-------+-------+-------+-------+-------------+--------+----------+--------------+
1 row in set (0.00 sec)
```

```
mysql> show tables;
+-------------------+
| Tables_in_student |
+-------------------+
| marks             |
| student           |
+-------------------+
2 rows in set (0.00 sec)
```

References

We referred to these books and links for our project entitled “Student Management System”

1. Computer Science with Python-class 12 by Preeti Arora
2. https://youtube.com

Other than this, the suggestions and learnings of our computer teacher helped us in making our project.