INDEX

INTRODUCTION

The **SCHOOL MANAGEMENT SYSTEM** is an integrated software which is used to maintain the details of the school. This project contains the modules to deal the details of students and employees working in the school. This software is used to store, edit, search and delete the details of the students as well as employees. Additionally it handles School Rules for students.

OBJECTIVES OF THE PROJECT

The objective of this project is to let the students apply the programming knowledge into a real- world situation/problem and exposed the students how programming skills helps in developing a good software.

- Write programs utilizing modern software tools.
- Apply object oriented programming principles effectively when developing small to medium sized projects.
- Write effective procedural code to solve small to medium sized problems.
- Students will demonstrate a breadth of knowledge in computer science, as exemplified in the areas of systems, theory and software development.
- Students will demonstrate ability to conduct a research or applied Computer Science project, requiring writing and presentation skills which exemplify scholarly style in computer science.

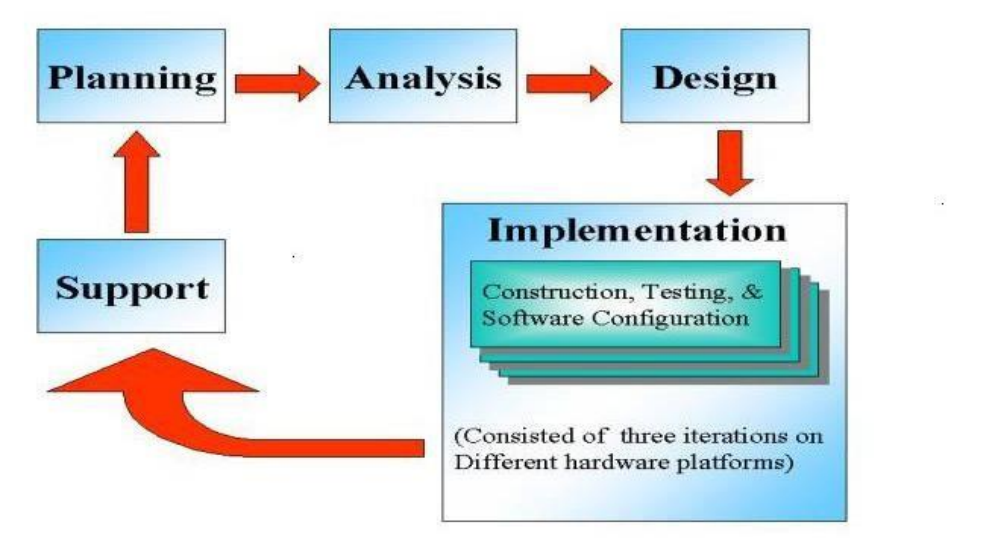
PROPOSED SYSTEM

Today one cannot afford to rely on the fallible human beings of be really wants to stand against today's merciless competition where not to wise saying **"to err is human"** no longer valid, it's outdated to rationalize your mistake. So, to keep pace with time, to bring about the best result without malfunctioning and greater efficiency so to replace the unending heaps of flies with a much sophisticated hard disk of the computer.

One has to use the data management software. Software has been an ascent in atomization various organizations. Many software products working are now in markets, which have helped in making the organizations work easier and efficiently. Data management initially had to maintain a lot of ledgers and a lot of paper work has to be done but now software product on this organization has made their work faster and easier. Now only this software has to be loaded on the computer and work can be done.

This prevents a lot of time and money. The work becomes fully automated and any information regarding the organization can be obtained by clicking the button. Moreover, now it's an age of computers of and automating such an organization gives the better look.

SYSTEM DEVELOPMENT LIFE CYCLE (SDLC)



The systems development life cycle is a project management technique that divides complex projects into smaller, more easily managed segments or phases. Segmenting projects allows managers to verify the successful completion of project phases before allocating resources to subsequent phases.

Software development projects typically include initiation, planning, design, development, testing, implementation, and maintenance phases. However, the phases may be divided differently depending on the organization involved.

For example, initial project activities might be designated as request, requirements-definition, and planning phases, or initiation, concept development, and planning phases. End users of the system under development should be involved in reviewing the output of each phase to ensure the system is being built to deliver the needed functionality.

PHASES OF SYSTEM DEVELOPMENT LIFE CYCLE

INITIATION PHASE

The Initiation Phase begins when a business sponsor identifies a need or an opportunity.

The purpose of the Initiation Phase is to:

- Identify and validate an opportunity to improve business accomplishments of the organization or a deficiency related to a business need.
- Identify significant assumptions and constraints on solutions to that need.
- Recommend the exploration of alternative concepts and methods to satisfy the need including questioning the need for technology, i.e., will a change in the business process offer a solution?
- Assure executive business and executive technical sponsorship. The Sponsor designates a Project Manager and the business need is documented in a Concept Proposal. The Concept Proposal includes information about the business process and the relationship to the Agency/Organization.
- Infrastructure and the Strategic Plan. A successful Concept Proposal results in a Project Management Charter which outlines the authority of the project manager to begin the project.

Careful oversight is required to ensure projects support strategic business objectives and resources are effectively implemented into an organization's enterprise architecture. The initiation phase begins when an opportunity to add, improve, or correct a system is identified and formally requested through the presentation of a business case. The business case should, at a minimum, describe a proposal's purpose, identify expected benefits, and explain how the proposed system supports one of the organization's business strategies. The business case should also identify alternative solutions and detail as many informational, functional, and network requirements as possible.

SYSTEM CONCEPT DEVELOPMENT PHASE

The System Concept Development Phase begins after a business need or opportunity is validated by the Agency/Organization Program Leadership and the Agency/Organization CIO.

The purpose of the System Concept Development Phase is to:

- Determine the feasibility and appropriateness of the alternatives.
- Identify system interfaces.
- Identify basic functional and data requirements to satisfy the business need.
- Establish system boundaries; identify goals, objectives, critical success factors, and performance measures.
- Evaluate costs and benefits of alternative approaches to satisfy the basic functional requirements
- Assess project risks
- Identify and initiate risk mitigation actions, and Develop high-level technical architecture, process models, data models, and a concept of operations. This phase explores potential technical solutions within the context of the business need.
- It may include several trade-off decisions such as the decision to use COTS software products as opposed to developing custom software or reusing software components, or the decision to use an incremental delivery versus a complete, onetime deployment.
- Construction of executable prototypes is encouraged to evaluate technology to support the business process. The System Boundary Document serves as an important reference document to support the Information Technology Project Request (ITPR) process. The ITPR must be approved by the State CIO before the project can move forward.

PICTORIAL REPRESENTATION OF SDLC:

PLANNING PHASE

The planning phase is the most critical step in completing development, acquisition, and maintenance projects. Careful planning, particularly in the early stages of a project, is necessary to coordinate activities and manage project risks effectively. The depth and formality of project plans should be commensurate with the characteristics and risks of a given project. Project plans refine the information gathered during the initiation phase by further identifying the specific activities and resources required to complete a project.

A critical part of a project manager's job is to coordinate discussions between user, audit, security, design, development, and network personnel to identify and document as many functional, security, and network requirements as possible. During this phase, a plan is developed that documents the approach to be used and includes a discussion of methods, tools, tasks, resources, project schedules, and user input. Personnel assignments, costs, project schedule, and target dates are established.

A Project Management Plan is created with components related to acquisition planning, configuration management planning, quality assurance planning, concept of operations, system security, verification and validation, and systems engineering management planning.

REQUIREMENTS ANALYSIS PHASE

This phase formally defines the detailed functional user requirements using high-level requirements identified in the Initiation, System Concept, and Planning phases. It also delineates the requirements in terms of data, system

performance, security, and maintainability requirements for the system. The requirements are defined in this phase to a level of detail sufficient for systems design to proceed. They need to be measurable, testable, and relate to the business need or opportunity identified in the Initiation Phase. The requirements that will be used to determine acceptance of the system are captured in the Test and Evaluation Master Plan.

The purposes of this phase are to:

- Further define and refine the functional and data requirements and document them in the Requirements Document,
- Complete business process reengineering of the functions to be supported (i.e., verify what information drives the business process, what information is generated, who generates it, where does the information go, and who processes it),
- Develop detailed data and process models (system inputs, outputs, and the process.
- Develop the test and evaluation requirements that will be used to determine acceptable system performance.

DESIGN PHASE

The design phase involves converting the informational, functional, and network requirements identified during the initiation and planning phases into unified design specifications that developers use to script programs during the development phase. Program designs are constructed in various ways. Using a top-down approach, designers first identify and link major program components and interfaces, then expand design layouts as they identify and link smaller subsystems and connections. Using a bottom-up approach, designers first identify and link minor program components and interfaces, then expand design layouts as they identify and link larger systems and connections. Contemporary design techniques often use prototyping tools that build mock-up designs of items such as application screens, database layouts, and system architectures. End users, designers, developers, database managers, and network administrators should review and refine the prototyped designs in an iterative process until they agree on an acceptable design. Audit, security, and quality assurance personnel should be involved in the review and approval process. During this phase, the system is designed to satisfy the functional requirements identified in the previous phase. Since problems in the design phase could be very expensive to solve in the later stage of the software development, a variety of elements are considered in the design to mitigate risk. These include:

- Identifying potential risks and defining mitigating design features.
- Performing a security risk assessment.

- Developing a conversion plan to migrate current data to the new system.
- Determining the operating environment.
- Defining major subsystems and their inputs and outputs.
- Allocating processes to resources.
- Preparing detailed logic specifications for each software module. The result is a draft System Design Document which captures the preliminary design for the system.
- Everything requiring user input or approval is documented and reviewed by the user. Once these documents have been approved by the Agency CIO and Business Sponsor, the final System Design Document is created to serve as the Critical/Detailed Design for the system.
- This document receives a rigorous review by Agency technical and functional representatives to ensure that it satisfies the business requirements. Concurrent with the development of the system design, the Agency Project Manager begins development of the Implementation Plan, Operations and Maintenance Manual, and the Training Plan.

DEVELOPMENT PHASE

The development phase involves converting design specifications into executable programs. Effective development standards include requirements that programmers and other project participants discuss design specifications before programming begins. The procedures help ensure programmers clearly understand program designs and functional requirements. Programmers use various techniques to develop computer programs. The large transaction oriented programs associated with financial institutions have traditionally been developed using procedural programming techniques. Procedural programming involves the line-by-line scripting of logical instructions that are combined to form a program. Effective completion of the previous stages is a key factor in the success of the Development phase. The Development phase consists of:

- Translating the detailed requirements and design into system components.
- Testing individual elements (units) for usability.
- Preparing for integration and testing of the IT system.

INTEGRATION AND TEST PHASE

- Subsystem integration, system, security, and user acceptance testing is conducted during the integration and test phase. The user, with those responsible for quality assurance, validates that the functional requirements, as defined in the functional requirements document, are satisfied by the developed or modified system. OIT Security staff assesses

the system security and issue a security certification and accreditation prior to installation/implementation.

Multiple levels of testing are performed, including:

- Testing at the development facility by the contractor and possibly supported by end users

- Testing as a deployed system with end users working together with contract personnel

- Operational testing by the end user alone performing all functions. Requirements are traced throughout testing, a final Independent Verification & Validation evaluation is performed and all documentation is reviewed and accepted prior to acceptance of the system.

IMPLEMENTATION PHASE

This phase is initiated after the system has been tested and accepted by the user. In this phase, the system is installed to support the intended business functions. System performance is compared to performance objectives established during the planning phase. Implementation includes user notification, user training, installation of hardware, installation of software onto production computers, and integration of the system into daily work processes. This phase continues until the system is operating in production in accordance with the defined user requirements.

OPERATIONS AND MAINTENANCE PHASE

The system operation is ongoing. The system is monitored for continued performance in accordance with user requirements and needed system modifications are incorporated. Operations continue as long as the system can be effectively adapted to respond to the organization's needs. When modifications or changes are identified, the system may reenter the planning phase.

The purpose of this phase is to:

- Operate, maintain, and enhance the system.
- Certify that the system can process sensitive information.
- Conduct periodic assessments of the system to ensure the functional requirements continue to be satisfied.
- Determine when the system needs to be modernized, replaced, or retired.

CODING

SchoolManagement1.py - C:/Users/pc/Desktop/PracticePythonCoding/Exam/SchoolManagement1.py (3.10.1)

File Edit Format Run Options Window Help

```
import mysql.connector
#database=apple table=emp
def addrecord():
    conn=mysql.connector.connect(host="localhost",user="root",passwd="VikasKumar",database="apple")
    if conn.is_connected()==False:
        print("CONNECTION ERROR")
        exit()
    cursor=conn.cursor()
    eid=int(input("Enter Employee ID    :"))
    ename=input("enter employee name    :")
    edob=input("Enter date of birth     :")
    edept=input("Enter Department       :")
    edesig=input("Enter Designation     :")
    esal=float(input("Enter salary        :"))
    q="insert into emp values({},'{}','{}','{}','{}',{})".format(eid,ename,edob,edept,edesig,esal)
    cursor.execute(q)
    print("RECORD ADDED SUCCESSFULLY")
    conn.commit()
    cursor.close()
    conn.close()

def searchrecord():
    conn=mysql.connector.connect(host='localhost',user='root',passwd='VikasKumar',database='apple')
    if conn.is_connected()==False:
        print("Connection Error!!!")
        exit()
    cursor=conn.cursor()
    eid=int(input("Enter the EMPLOYEE ID of the employee you want to fetch:"))
    q="select * from emp where eid={}".format(eid)
    cursor.execute(q)
    row=cursor.fetchone()
    n=cursor.rowcount
    if n==1:
        print("DETAILS OF THE EMPLOYEE WITH EMPLOYEE ID ",eid,"ARE:")
        print("=====================================================")
        print("EMPLOYEE ID         ",row[0])
        print("EMPLOYEE NAME       ",row[1])
        print("EMPLOYEE DOB        ",row[2])
        print("EMPLOYEE DEPARTMENT ",row[3])
        print("EMPLOYEE DESIGNATION",row[4])
        print("EMPLOYEE SALARY     ",row[5])
```

SchoolManagement1.py - C:/Users/pc/Desktop/PracticePythonCoding/Exam/SchoolManagement1.py (3.10.1)

File Edit Format Run Options Window Help

```
        print("EMPLOYEE SALARY     ",row[5])
        print("============================================================")
    else:
        print("\n\n")
        print("============================================================")
        print("SORRY!!!RECORD NOT FOUND")
        print("============================================================")
        print("\n\n")
    conn.commit()
    cursor.close()
    conn.close()

def deleterecord():
    conn=mysql.connector.connect(host='localhost',user='root',passwd='VikasKumar',database='apple')
    if conn.is_connected==False:
        print("Conection FAILED!!!!")
        exit()
    cursor=conn.cursor()
    eid=int(input("ENTER THE EMPLOYEE ID OF THE EMPLOYEE YOU WANT TO DELETE:"))
    q=("select * from emp where eid={}").format(eid)
    cursor.execute(q)
    row=cursor.fetchone()
    n=cursor.rowcount
    if n==1:
        print("DETAILS OF THE EMPLOYEE DELETED WITH EMPLOYEE EID",eid,"ARE:")
        print("============================================================")
        print("EMPLOYEE EID                :",row[0])
        print("EMPLOYEE NAME               :",row[1])
        print("EMPLOYEE DATE OF BIRTH      :",row[2])
        print("EMPLOYEE DEPARTMENT         :",row[3])
        print("EMPLOYEE DESIGNATION        :",row[4])
        print("EMPLOYEE SALARY             ;",row[5])
        q2=("delete from emp where eid={}").format(eid)
        cursor.execute(q2)
        conn.commit()
    else:
        print("\n\n")
        print("============================================================")
        print("SORRY!!!RECORD NOT FOUND")
        print("============================================================")
        print("\n\n")
```

SchoolManagement1.py - C:/Users/pc/Desktop/PracticePythonCoding/Exam/SchoolManagement1.py (3.10.1)

File Edit Format Run Options Window Help

```
    else:
        print("\n\n")
        print("================================================================")
        print("SORRY!!!RECORD NOT FOUND")
        print("================================================================")
        print("\n\n")


    cursor.close()
    conn.close()

def updaterecord():
    conn=mysql.connector.connect(host="localhost",user="root",passwd="VikasKumar",database="apple")
    if conn.is_connected==False:
        print("CONNECTION ERROR!!!!")
        exit()
    cursor=conn.cursor()
    eid=int(input("ENTER THE EMPLOYEE EID OF THE PERSON YOU WANT TO EDIT"))
    q=("select * from emp where eid={}").format(eid)
    cursor.execute(q)
    row=cursor.fetchone()
    n=cursor.rowcount
    if n==1:
        print("DETAILS OF THE EMPLOYEE WITH EMPLOYEE ID:",eid,"are:")
        print("=====================================================")
        print("EMPLOYEE ID         :",row[0])
        print("EMPLOYEE NAME       :",row[1])
        print("EMPLOYEE DateOfBirth:",row[2])
        print("EMPLOYEE DESIGNATION:",row[3])
        print("EMPLOYEE DEPARTMENT :",row[4])
        print("EMPLOYEE SALARY     :",row[5])
        print("=====================================================")
        print("\n\n")
        print("ENTER NEW DETAILS FOR EMPLOYEE ID",eid)
        ename=input("ENTER EMPLOYEE NAME          :")
        edob=input("ENTER Date Of Birth           :")
        edept=input("Enter DEPARTMENT             :")
        edesig=input("ENTER DESIGNATION           :")
        esal=input("ENTER SALARY                  :")
        q2="update emp set ename='{}',edob='{}',edept='{}',edesig='{}',esal={} where eid={}".format(ename,edob,edesig,edept,esal,eid)
        cursor.execute(q2)
```

SchoolManagement1.py - C:/Users/pc/Desktop/PracticePythonCoding/Exam/SchoolManagement1.py (3.10.1)

File Edit Format Run Options Window Help

```
        cursor.execute(q2)
        print("RECORD UPDATED SUCCESSFULLY")
        conn.commit()
    else:
        print("\n\n")
        print("============================================================")
        print("SORRY!!!RECORD NOT FOUND")
        print("============================================================")
        print("\n\n")
    cursor.close()
    conn.close()

def display():
    conn=mysql.connector.connect(host='localhost',\
                                 user='root',\
                                 passwd="VikasKumar",\
                                 database='apple')
    if conn.is_connected==False:
        print("CONNECTION ERROR!!!!1")
        exit()
    cursor=conn.cursor()

    cursor.execute("select * from emp ")
    data=cursor.fetchall()
    count=cursor.rowcount
    print("NO. OF RECORDS FETCHED:",count)
    for row in data:
        print("============================================================")
        print("EMPLOYEE ID             :",row[0])
        print("EMPLOYEE NAME           :",row[1])
        print("EMPLOYEE DATE OF BIRTH :",row[2])
        print("EMPLOYEE DEPARTMENT     :",row[3])
        print("EMPLOYEE DESIGNATION    :",row[4])
        print("EMPLOYEE SALARY         :",row[5])
        print("============================================================")
    conn.commit()
    cursor.close()
    conn.close()

def ethics():
    print('''^^^^^^^^^^^^^^^SCHOOL RULES:\n
```

SchoolManagement1.py - C:/Users/pc/Desktop/PracticePythonCoding/Exam/SchoolManagement1.py (3.10.1)

File Edit Format Run Options Window Help

```
    conn.commit()
    cursor.close()
    conn.close()

def ethics():
    print('''^^^^^^^^^^^^^^^^SCHOOL RULES:\n
1.Keep your classroom neat and clean
2.Always be properly dressed as per uniform code
3.Show due respect to all school employees.\n''')

def studrecord():
    conn=mysql.connector.connect(host="localhost",user="root",passwd="VikasKumar",database="apple")
    if conn.is_connected()==False:
        print("CONNECTION ERROR")
        exit()
    cursor=conn.cursor()
    adm=int(input("Enter Admission No.  :"))
    sname=input("enter student name     :")
    sdob=input("Enter date of birth     :")
    sclass=input("Enter class           :")
    sec=input("Enter section            :")
    q="insert into student values({},'{}','{}','{}','{}')".format(adm,sname,sdob,sclass,sec)
    cursor.execute(q)
    print("RECORD ADDED SUCCESSFULLY")
    conn.commit()
    cursor.close()
    conn.close()

def studsearch():
    conn=mysql.connector.connect(host='localhost',user='root',passwd='VikasKumar',database='apple')
    if conn.is_connected()==False:
        print("Connection Error!!!")
        exit()
    cursor=conn.cursor()
    adm=int(input("Enter the STUDENT ADMISSION NO. In order to search:"))
    q="select * from student where A_no={}".format(adm)
    cursor.execute(q)
    row=cursor.fetchone()
    n=cursor.rowcount
    if n==1:
        print("DETAILS OF THE STUDENT WITH ADMISSION NO. ",adm,"ARE:")
```

SchoolManagement1.py - C:/Users/pc/Desktop/PracticePythonCoding/Exam/SchoolManagement1.py (3.10.1)

File Edit Format Run Options Window Help

```
        print("DETAILS OF THE STUDENT WITH ADMISSION NO. ",adm,"ARE:")
        print("=====================================================")
        print("ADMISSION NO.      ",row[0])
        print("STUDENT NAME       ",row[1])
        print("STUDENT  DOB       ",row[2])
        print("STUDENT CLASS      ",row[3])
        print("STUDENT SECTION    ",row[4])
        print("=====================================================")
    else:
        print("\n\n")
        print("=====================================================")
        print("SORRY!!!RECORD NOT FOUND")
        print("=====================================================")
        print("\n\n")
    conn.commit()
    cursor.close()
    conn.close()

def studremove():
    conn=mysql.connector.connect(host='localhost',user='root',passwd='VikasKumar',database='apple')
    if conn.is_connected==False:
        print("Conection FAILED!!!!")
        exit()
    cursor=conn.cursor()
    adm=int(input("ENTER THE ADMISSION NO. OF THE STUDENT YOU WANT TO DELETE:"))
    q=("select * from student where A_no={}").format(adm)
    cursor.execute(q)
    row=cursor.fetchone()
    n=cursor.rowcount
    if n==1:
        print("DETAILS OF THE STUDENT DELETED WITH ADMISSION NO.",adm,"ARE:")
        print("============================================================")
        print("ADMISSION NO.               :",row[0])
        print("STUDENT  NAME               :",row[1])
        print("STUDENT DATE OF BIRTH       :",row[2])
        print("STUDENT CLASS               :",row[3])
        print("STUDENT SECTION             :",row[4])
        q2=("delete from student where A_no={}").format(adm)
        cursor.execute(q2)
        conn.commit()
    else:
```

SchoolManagement1.py - C:/Users/pc/Desktop/PracticePythonCoding/Exam/SchoolManagement1.py (3.10.1)

File Edit Format Run Options Window Help

```
    else:
        print("\n\n")
        print("=================================================================")
        print("SORRY!!!RECORD NOT FOUND")
        print("=================================================================")
        print("\n\n")

    cursor.close()
    conn.close()




while True:
    print("=================================================================")
    print("==============SCHOOL MANAGEMENT==================================")
    print("=================================================================")
    print("=================================================================")
    print('''Please choose an option: \n
    ~~~~~~~~~~~~~~~~~~~~~FOR STAFF(Employee):
    ================================================================
    1.FOR ADDING A RECORD  \n
    2.FOR DISPLAYING RECORD(S)  \n
    3.SEARCHING A RECORD   \n
    4.FOR DELETING A RECORD   \n
    5.FOR UPDATING A RECORD  \n
    ================================================================
    ================================================================
    ~~~~~~~~~~~~~~~~~~~~~FOR STUDENTS:
    7.FOR ADDING STUDENT RECORD \n
    8.FOR SEARCHING STUDENT RECORD \n
    9.FOR REMOVING STUDENT RECORD \n
    ================================================================
    ================================================================
    ~~CODE OF ETHICS FOR ALL STUDENTS:\n
    10.SCHOOL RULES  \n
    11.EXIT\n''')
```

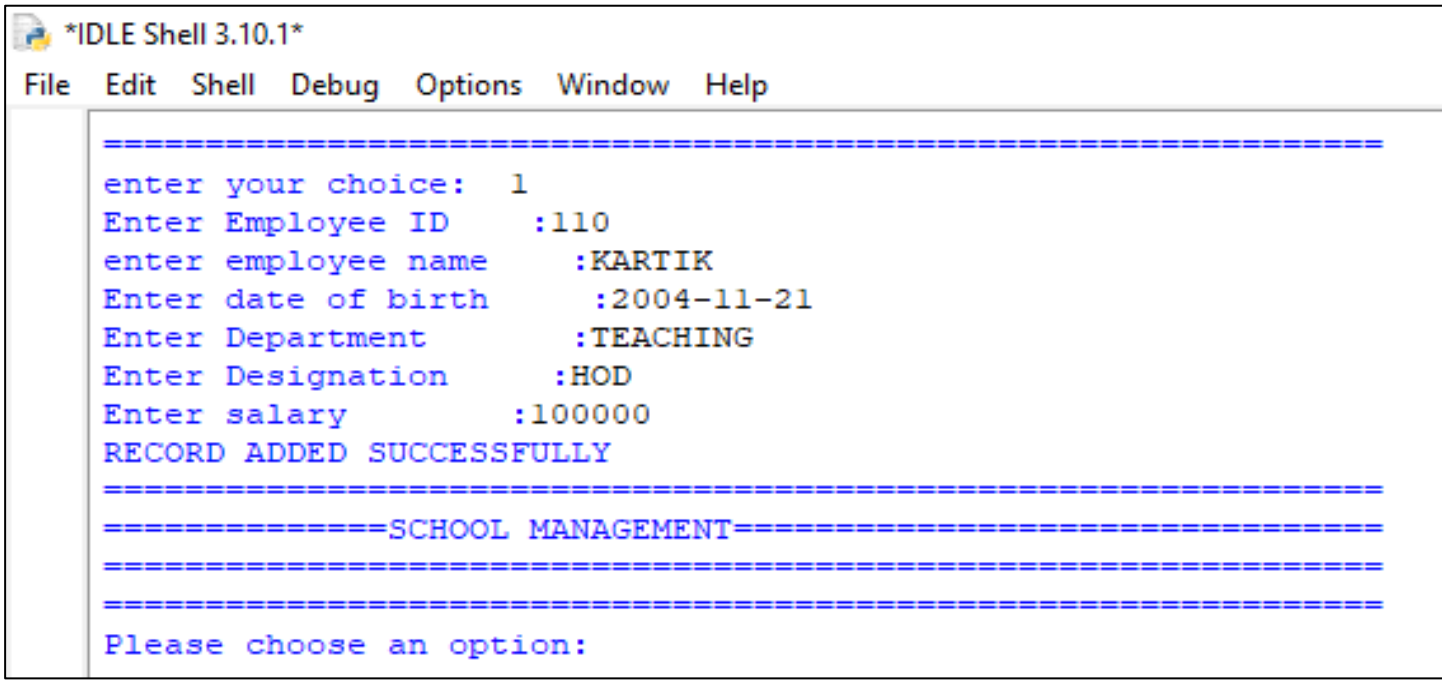
SchoolManagement1.py - C:/Users/pc/Desktop/PracticePythonCoding/Exam/SchoolManagement1.py (3.10.1)

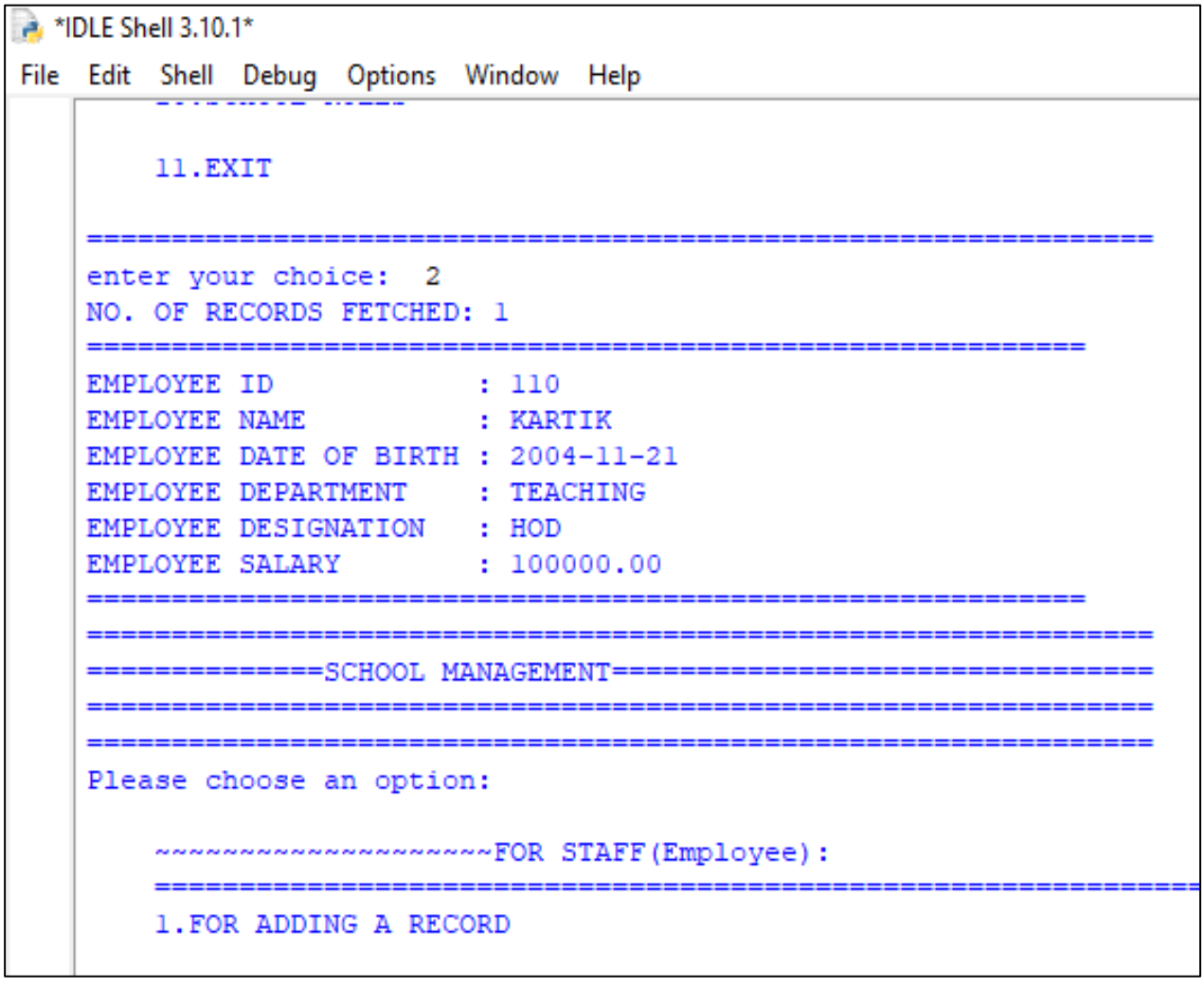
File Edit Format Run Options Window Help

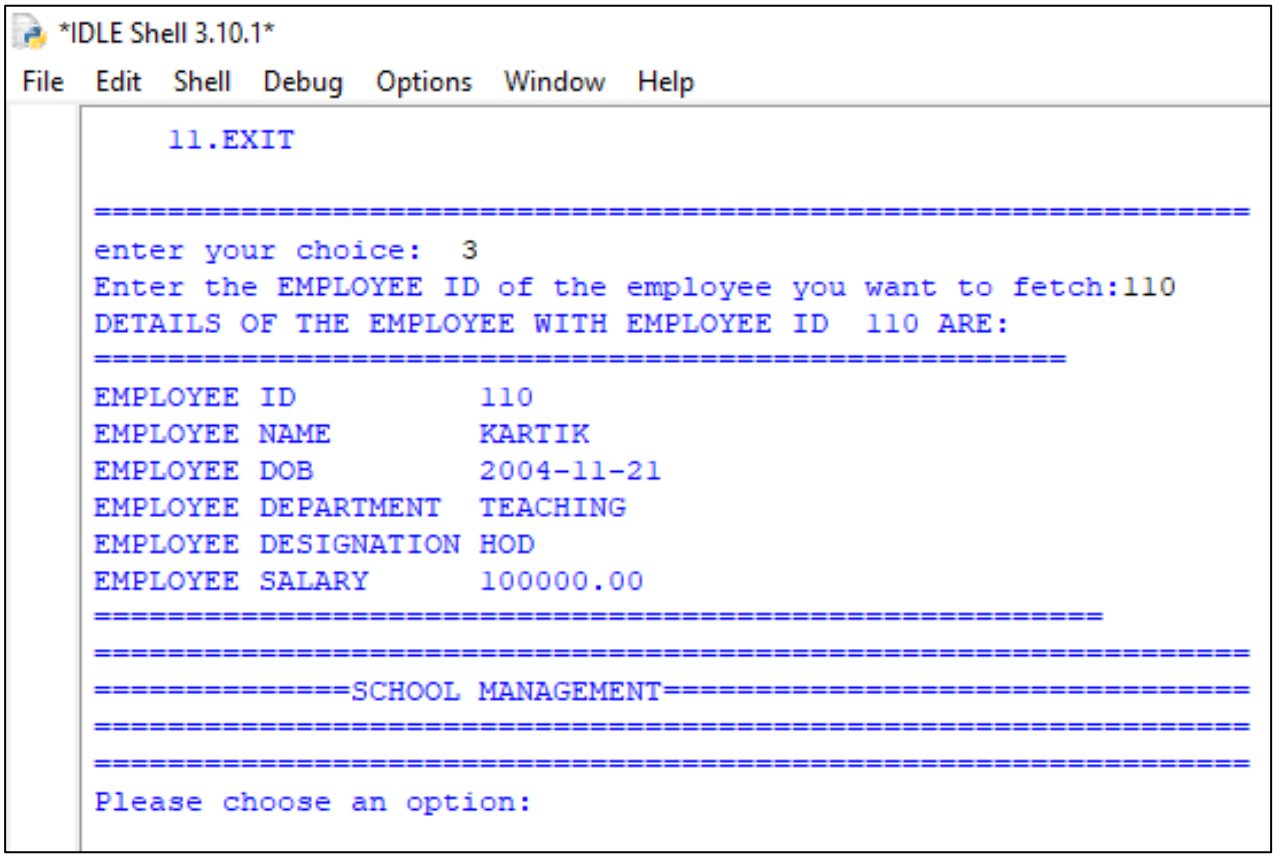
```
    ================================================================================
    ~~~~~~~~~~~~~~~~~~~~~~FOR STUDENTS:
    7.FOR ADDING STUDENT RECORD \n
    8.FOR SEARCHING STUDENT RECORD \n
    9.FOR REMOVING STUDENT RECORD \n
    ================================================================================
    ================================================================================
    ~~CODE OF ETHICS FOR ALL STUDENTS:\n
    10.SCHOOL RULES  \n
    11.EXIT\n''')




    print("================================================================================")
    choice=int(input("enter your choice:  "))
    if choice==1:
        addrecord()
    elif choice==2:
        display()
    elif choice==3:
        searchrecord()
    elif choice==4:
        deleterecord()
    elif choice==5:
        updaterecord()
    elif choice==7:
        studrecord()
    elif choice==8:
        studsearch()
    elif choice==9:
        studremove()
    elif choice==10:
        ethics()
    elif choice==11:
        print("Thanks for using our software:")
        break
    else:
        print("wrong choice entered!!!PLZ CHOOOSE AGAIN")
```

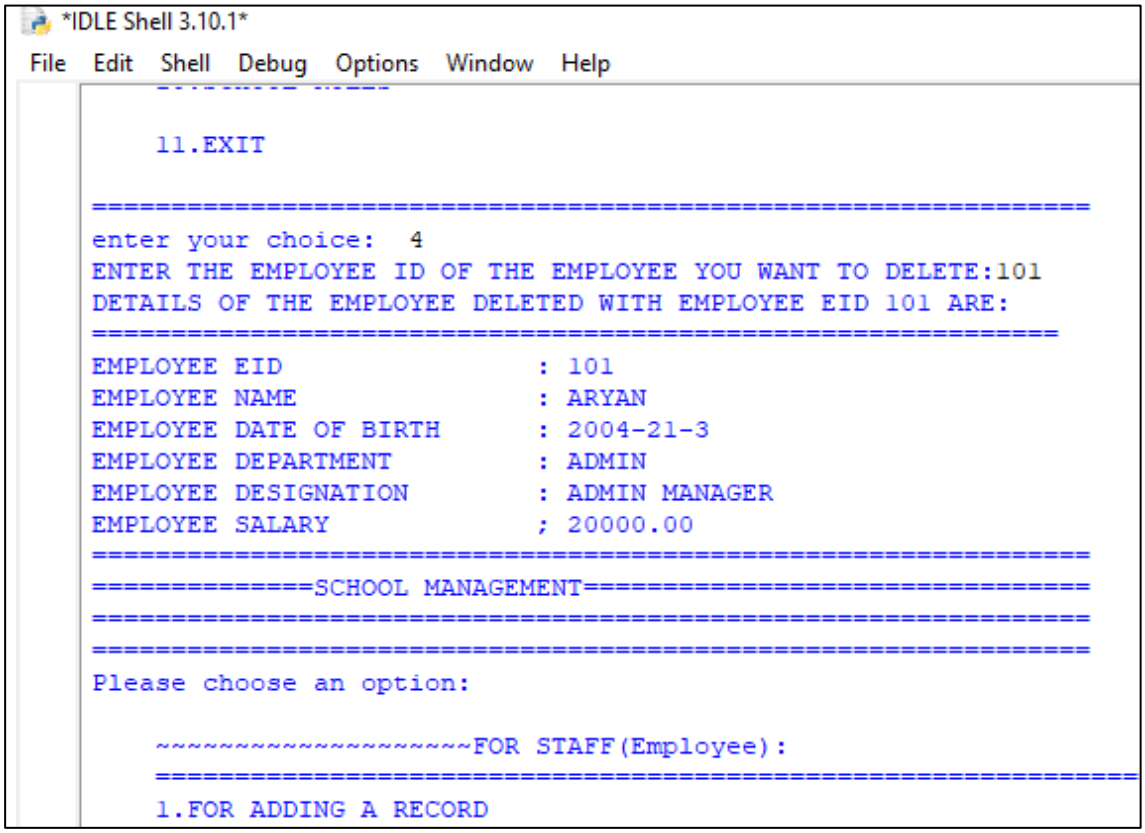
OUTPUT

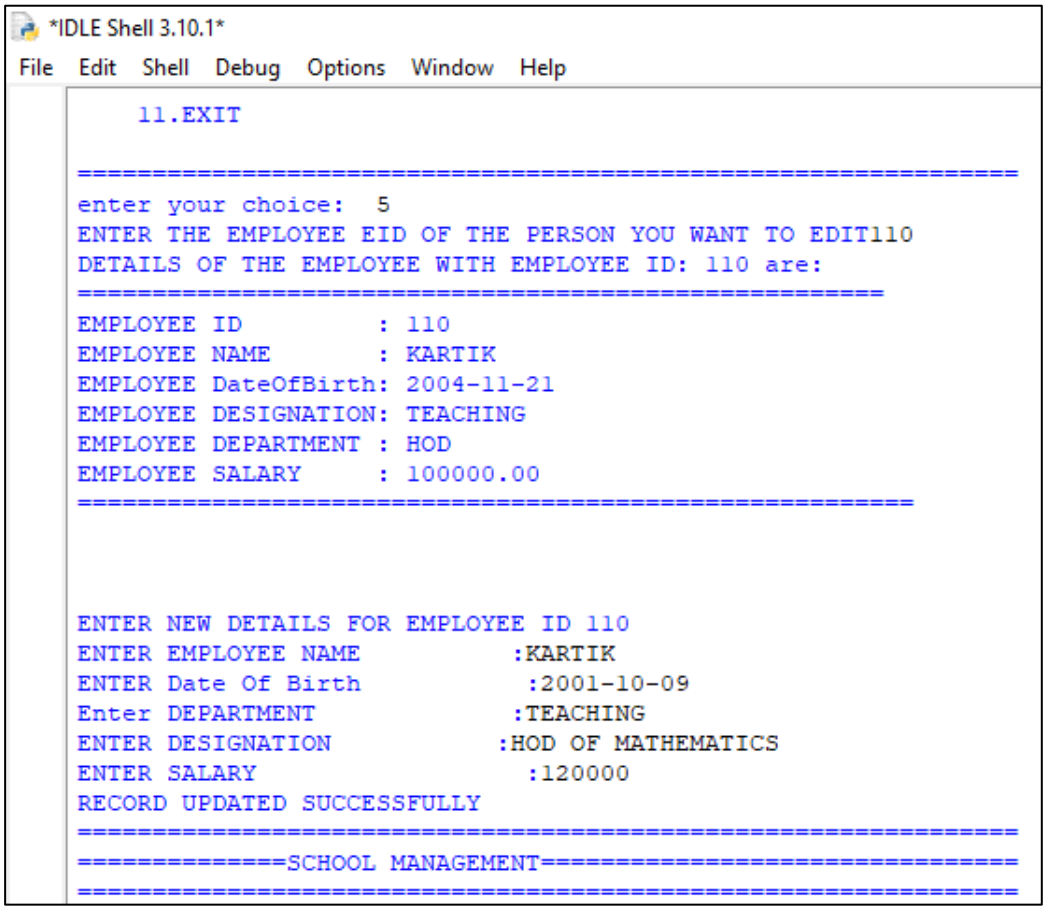
```
= RESTART: C:/Users/pc/Desktop/PracticePythonCoding/Exam/SchoolManagement1.py
================================================================
==============SCHOOL MANAGEMENT=================================
================================================================
================================================================
Please choose an option:

    ~~~~~~~~~~~~~~~~~~~~~FOR STAFF(Employee):
    ================================================================================
    1.FOR ADDING A RECORD

    2.FOR DISPLAYING RECORD(S)

    3.SEARCHING A RECORD

    4.FOR DELETING A RECORD

    5.FOR UPDATING A RECORD

    ================================================================================
    ================================================================================
    ~~~~~~~~~~~~~~~~~~~~~FOR STUDENTS:
    7.FOR ADDING STUDENT RECORD

    8.FOR SEARCHING STUDENT RECORD

    9.FOR REMOVING STUDENT RECORD

    ================================================================================
    ================================================================================
    ~~CODE OF ETHICS FOR ALL STUDENTS:

    10.SCHOOL RULES

    11.EXIT

================================================================
enter your choice:
```

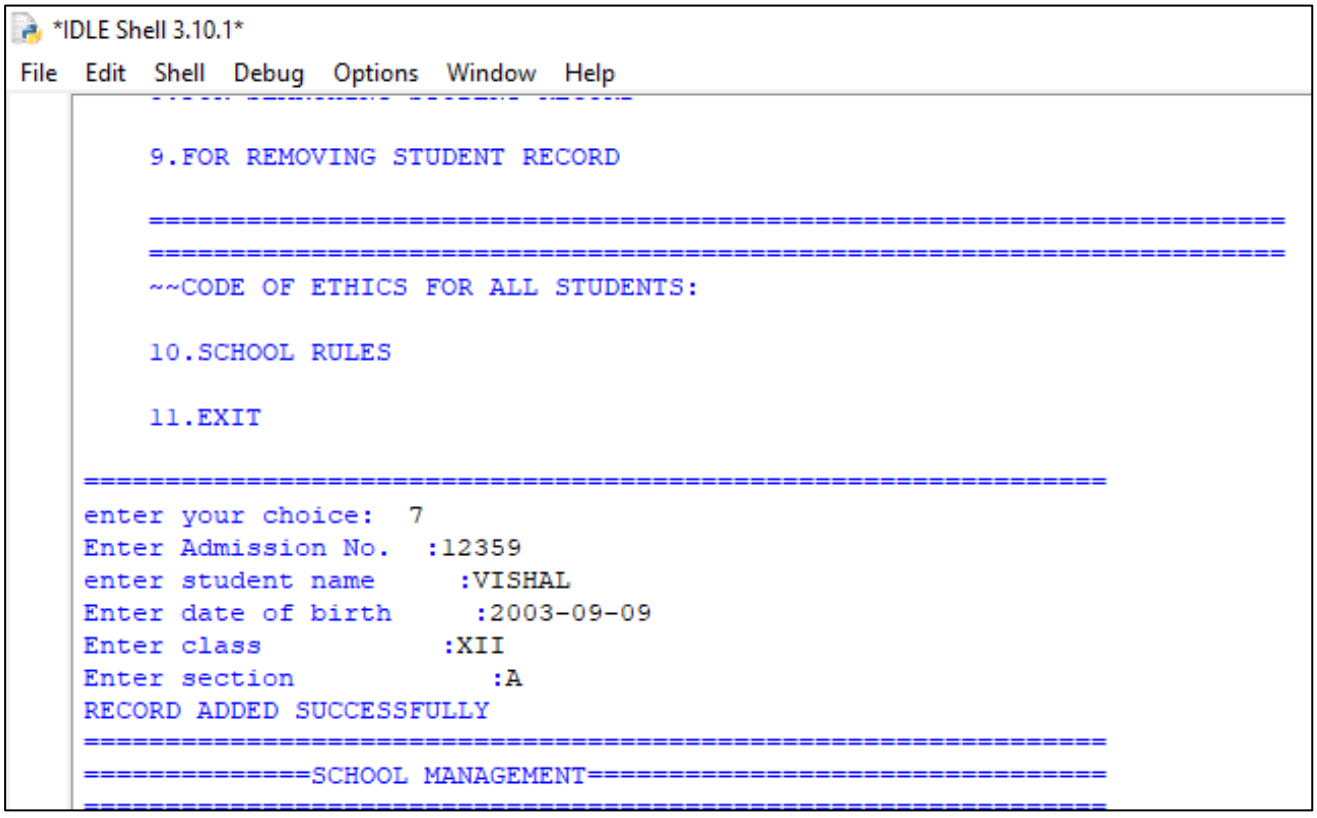

IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
==========
enter your choice: 1
Enter Employee ID :110
enter employee name :KARTIK
Enter date of birth :2004-11-21
Enter Department :TEACHING
Enter Designation :HOD
Enter salary :100000
RECORD ADDED SUCCESSFULLY
==========
==========SCHOOL MANAGEMENT==========
==========
==========
Please choose an option:

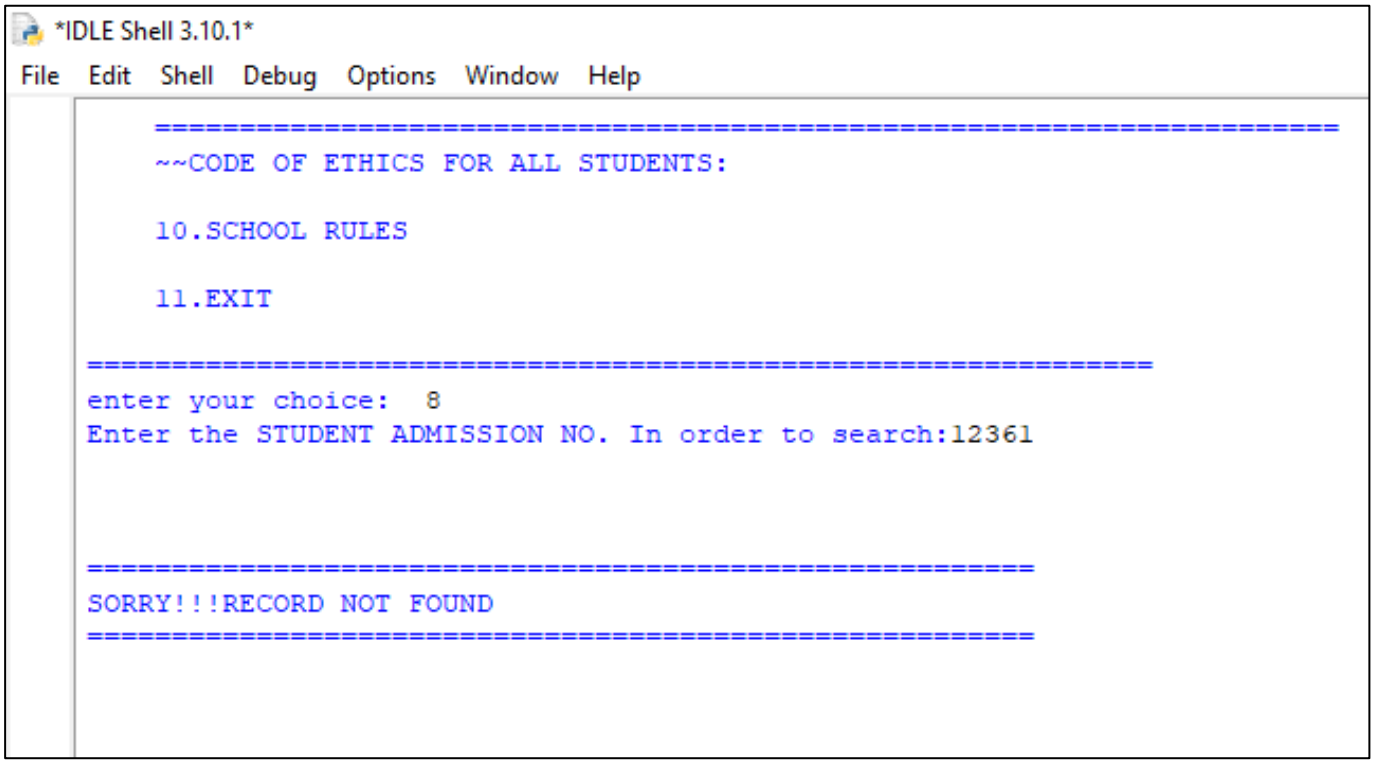

IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
11.EXIT
==========
enter your choice: 2
NO. OF RECORDS FETCHED: 1
==========
EMPLOYEE ID : 110
EMPLOYEE NAME : KARTIK
EMPLOYEE DATE OF BIRTH : 2004-11-21
EMPLOYEE DEPARTMENT : TEACHING
EMPLOYEE DESIGNATION : HOD
EMPLOYEE SALARY : 100000.00
==========
==========
==========SCHOOL MANAGEMENT==========
==========
==========
Please choose an option:
~~~~~~~~~~~~~~~~~~~~FOR STAFF(Employee):
==========
1.FOR ADDING A RECORD
~~~~~~~~~~~~~~~~~~~~

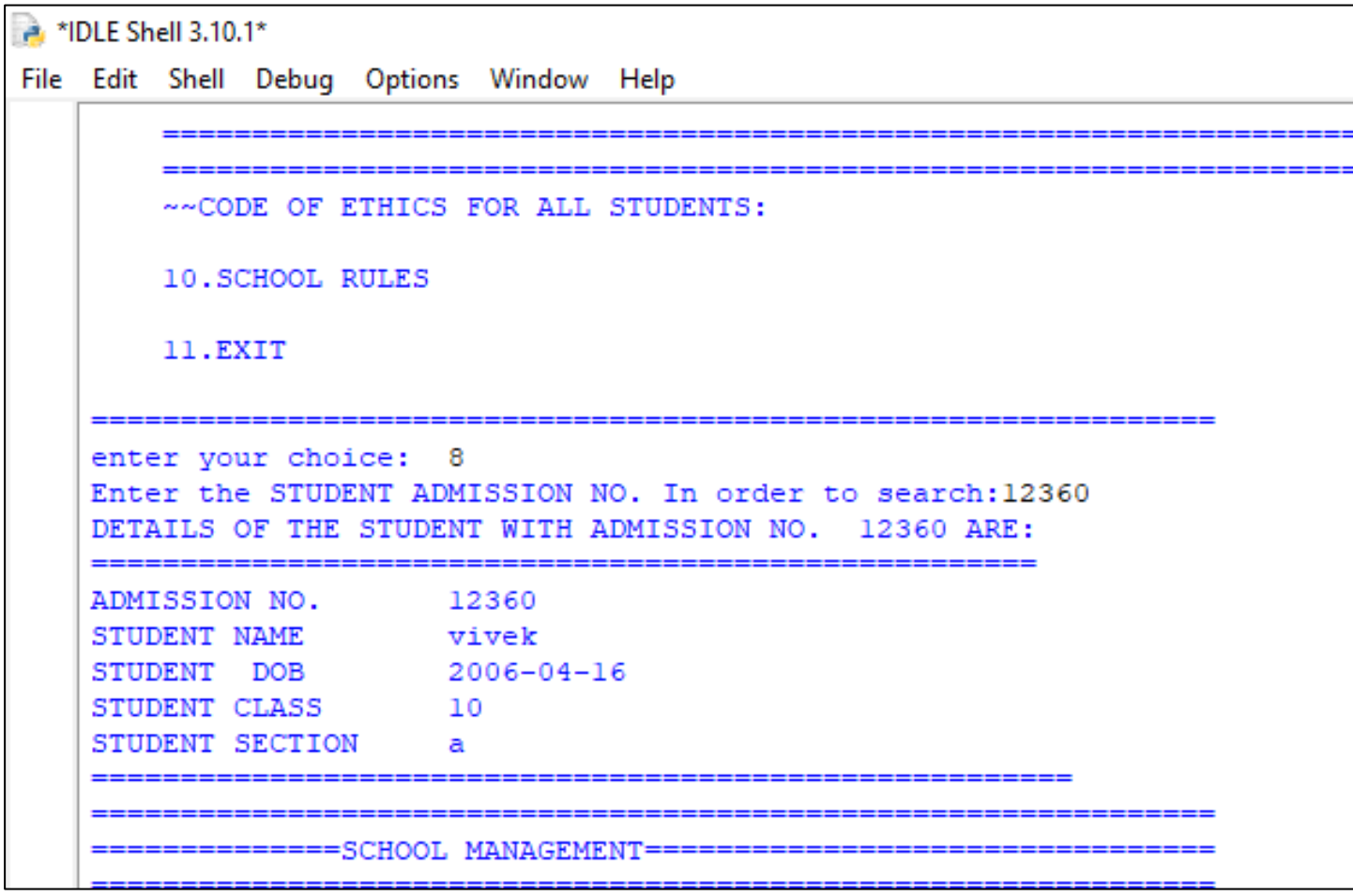

IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
11.EXIT
==========
enter your choice: 3
Enter the EMPLOYEE ID of the employee you want to fetch:110
DETAILS OF THE EMPLOYEE WITH EMPLOYEE ID 110 ARE:
==========
EMPLOYEE ID 110
EMPLOYEE NAME KARTIK
EMPLOYEE DOB 2004-11-21
EMPLOYEE DEPARTMENT TEACHING
EMPLOYEE DESIGNATION HOD
EMPLOYEE SALARY 100000.00
==========
==========
==========SCHOOL MANAGEMENT==========
==========
==========
Please choose an option:

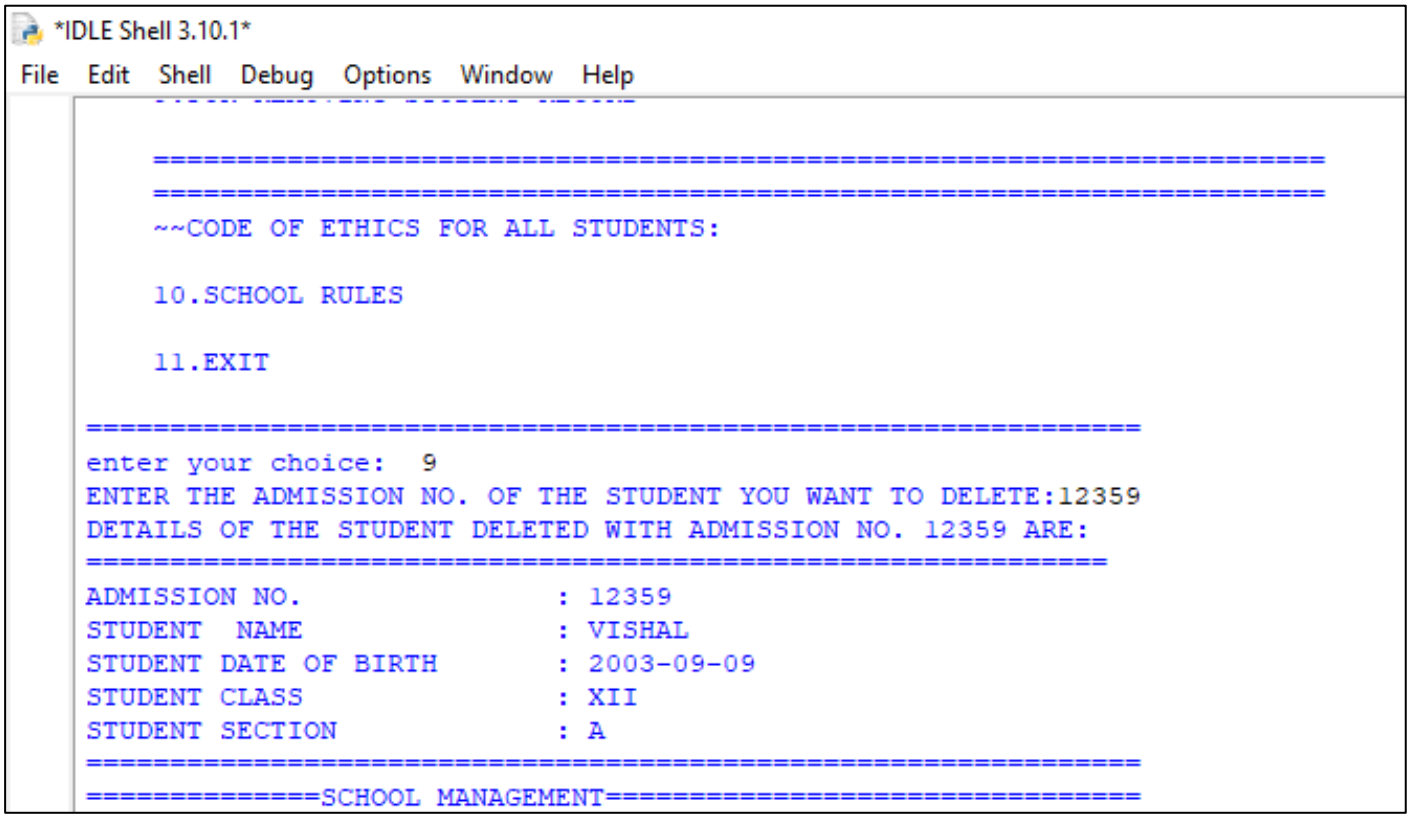

IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
11.EXIT
==========
enter your choice: 4
ENTER THE EMPLOYEE ID OF THE EMPLOYEE YOU WANT TO DELETE:101
DETAILS OF THE EMPLOYEE DELETED WITH EMPLOYEE EID 101 ARE:
==========
EMPLOYEE EID : 101
EMPLOYEE NAME : ARYAN
EMPLOYEE DATE OF BIRTH : 2004-21-3
EMPLOYEE DEPARTMENT : ADMIN
EMPLOYEE DESIGNATION : ADMIN MANAGER
EMPLOYEE SALARY ; 20000.00
==========
==========SCHOOL MANAGEMENT==========
==========
==========
Please choose an option:
~~~~~~~~~~~~~~~~~~~~FOR STAFF(Employee):
==========
1.FOR ADDING A RECORD
~~~~~~~~~~~~~~~~~~~~

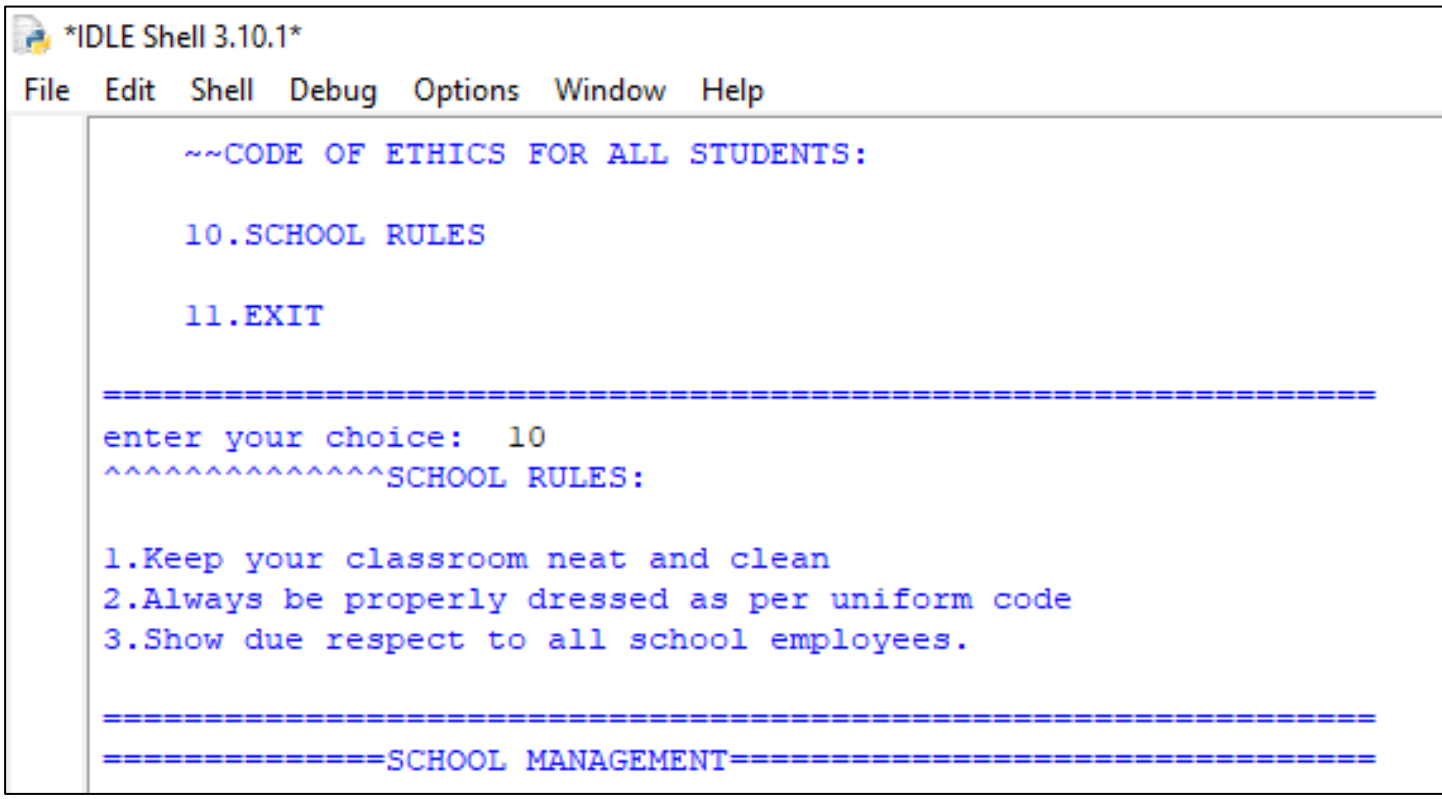

IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
11.EXIT
==========
enter your choice: 5
ENTER THE EMPLOYEE EID OF THE PERSON YOU WANT TO EDIT110
DETAILS OF THE EMPLOYEE WITH EMPLOYEE ID: 110 are:
==========
EMPLOYEE ID : 110
EMPLOYEE NAME : KARTIK
EMPLOYEE DateOfBirth: 2004-11-21
EMPLOYEE DESIGNATION: TEACHING
EMPLOYEE DEPARTMENT : HOD
EMPLOYEE SALARY : 100000.00
==========
ENTER NEW DETAILS FOR EMPLOYEE ID 110
ENTER EMPLOYEE NAME :KARTIK
ENTER Date Of Birth :2001-10-09
Enter DEPARTMENT :TEACHING
ENTER DESIGNATION :HOD OF MATHEMATICS
ENTER SALARY :120000
RECORD UPDATED SUCCESSFULLY
==========
==========SCHOOL MANAGEMENT==========
==========


IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
9.FOR REMOVING STUDENT RECORD
==========
==========
~~CODE OF ETHICS FOR ALL STUDENTS:
10.SCHOOL RULES
11.EXIT
==========
enter your choice: 7
Enter Admission No. :12359
enter student name :VISHAL
Enter date of birth :2003-09-09
Enter class :XII
Enter section :A
RECORD ADDED SUCCESSFULLY
==========
==========SCHOOL MANAGEMENT==========
==========


IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
==========
~~CODE OF ETHICS FOR ALL STUDENTS:
10.SCHOOL RULES
11.EXIT
==========
enter your choice: 8
Enter the STUDENT ADMISSION NO. In order to search:12361
==========
SORRY!!!RECORD NOT FOUND
==========


IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
~~CODE OF ETHICS FOR ALL STUDENTS:
10.SCHOOL RULES
11.EXIT
==========
enter your choice: 8
Enter the STUDENT ADMISSION NO. In order to search:12360
DETAILS OF THE STUDENT WITH ADMISSION NO. 12360 ARE:
==========
ADMISSION NO. 12360
STUDENT NAME vivek
STUDENT DOB 2006-04-16
STUDENT CLASS 10
STUDENT SECTION a
==========
==========SCHOOL MANAGEMENT==========


IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
==========
==========
~~CODE OF ETHICS FOR ALL STUDENTS:
10.SCHOOL RULES
11.EXIT
==========
enter your choice: 9
ENTER THE ADMISSION NO. OF THE STUDENT YOU WANT TO DELETE:12359
DETAILS OF THE STUDENT DELETED WITH ADMISSION NO. 12359 ARE:
==========
ADMISSION NO. : 12359
STUDENT NAME : VISHAL
STUDENT DATE OF BIRTH : 2003-09-09
STUDENT CLASS : XII
STUDENT SECTION : A
==========
==========SCHOOL MANAGEMENT==========


IDLE Shell 3.10.1
File Edit Shell Debug Options Window Help
~~CODE OF ETHICS FOR ALL STUDENTS:
10.SCHOOL RULES
11.EXIT
==========
enter your choice: 10
^^^^^^^^^^^^^^SCHOOL RULES:
1.Keep your classroom neat and clean
2.Always be properly dressed as per uniform code
3.Show due respect to all school employees.
==========
==========SCHOOL MANAGEMENT==========

```
    ==========================================================================
    ==========================================================================
    ~~CODE OF ETHICS FOR ALL STUDENTS:

    10.SCHOOL RULES

    11.EXIT

==================================================================
enter your choice:  11
Thanks for using our software:
>>>
```

HARDWARE AND SOFTWARE REQUIREMENTS

I.OPERATING SYSTEM : WINDOWS 7 AND ABOVE

II. PROCESSOR : PENTIUM(ANY) OR AMD ATHALON(3800+- 4200+ DUAL CORE)

III. MOTHERBOARD :1.845 OR 915,995 FOR PENTIUM 0R MSI K9MM-V VIA K8M800+8237R

PLUS

CHIPSET FOR AMD ATHALON

IV. RAM : 512MB+

V. Hard disk : SATA 40 GB OR ABOVE

VI. CD/DVD r/w multi drive combo : (If back up required)

VII. FLOPPY DRIVE 1.44 MB : (If Backup required)

VIII. MONITOR 14.1 or 15 -17 inch

IX. Key board and mouse

SOFTWARE REQUIREMENTS:

I. **Windows OS**

II. **Python , MySql**

BIBLIOGRAPHY

- ***Computer science With Python - Class XI & XII ,By Sumita Arora.***
- ***SULTAN CHAND PUBLICATIONS CS CLASS XI AND XII***
- ***https://python4csip.com/***

***

THANK YOU