

# Vishal Bharti Senior Secondary School



CENTRAL BOARD OF SECONDARY  
EDUCATION

**PROJECT FILE**                   **(RESULT MANAGEMENT SYSTEM)**

**CLASS XII EXAMINATION**

**SESSION - 2022-23**

**SUBMITTED BY:** Khushi Dua

**HOD(COMPUTER):**Mrs. Nomita Sharma

**CLASS:** XII-A

**ROLL NO:**

# ACKNOWLEDGEMENT

*I wish to express my deep sense of gratitude and indebtedness to our learned teacher **MRS. NOMITA SHARMA**, PGT COMPUTER SCIENCE, **VISHAL BHARTI SENIOR SECONDARY SCHOOL** for her invaluable help, advice and guidance in the preparation of this project.*

*I am also greatly indebted to our principal **MRS. KIRAN DHALL** and school authorities for providing me with the facilities and requisite laboratory conditions for making this project file.*

*I also extend my thanks to a number of teachers ,my classmates and friends who helped me to complete this project file successfully.*

**KHUSHI DUA**

# CERTIFICATE

*This is to certify that KHUSHI DUA, student of Class XII, VISHAL BHARTI SENIOR SECONDARY SCHOOL has successfully completed the PROJECT FILE during the academic year 2022-23 towards partial fulfillment of credit for the Computer Science practical evaluation of CBSE and submitted satisfactory project, as compiled in the following pages, under my supervision.*

*Head of the Department  
Signature*

*Principal  
Seal and Signature*

# **CONTENTS**

S.NO.	TOPIC
1.	<b>CERTIFICATE</b>
2.	<b>ACKNOWLEDGEMENT</b>
3.	<b>INTRODUCTION</b>
4.	<b>REQUIREMENTS</b>
5.	<b>FRONT END AND BACK END</b>
6.	<b>CODING</b>
7.	<b>OUTPUT</b>
8.	<b>BIBLIOGRAPHY</b>

---

**INTERNAL EXAMINER  
SIGNATURE**

---

**EXTERNAL EXAMINER  
SIGNATURE**

# **INTRODUCTION TO MY PROJECT**

**This document aims at defining overall software requirement for STUDENT RESULT MANAGEMENT SYSTEM .**

**Efforts have been made to define the requirements exhaustively and accurately. The final product will be having only features/functionalities mentioned in this document and assumptions for any additional functionality/feature should not be made by any of the parties in developing/testing /implementing /using this product.**

## **PURPOSE**

**This specification document describes the capabilities that will be provided by the software application STUDENT RESULT MANAGEMENT SYSTEM. It also states the various constraints by which the system will abide. This intended audience for this document are the development team, testing team and end users of the product.**

## **SCOPE**

**The application will manage the information about various students enrolled in this course in different years, the subject offered during different semesters of the course, the marks obtained by the various students in various subjects in different semesters.**

**The application will greatly simplify and speed up the result preparation and management process.**

# REQUIREMENTS

## HARDWARE REQUIREMENTS

- ❖ Computer , for coding and typing the required documents of the project.
- ❖ Printer , to print the required documents of the project.
- ❖ Compact Drive.
- ❖ Processor: Pentium Quad Core.
- ❖ RAM: 64 MB.
- ❖ Hard Disk: 20 GB.

## SOFTWARE REQUIREMENTS

- ❖ Operating system : Window 7 or above.
- ❖ Python 3 : for execution of programme.
- ❖ MySQL: for storing data in the database.
- ❖ Python-MySQL Connector: for database connectivity.
- ❖ Microsoft Word for documentation.

# **FRONT-END AND BACK-END USED**

## **FRONT END:**

Front End is the development environment where we write the program code to communicate with the system. I have taken Python to develop the project. Using IDLE programs were developed and tested.

## **BACK END:**

Back End refers to the database which is working in behind of front end. It is used to store data. I have taken MYSQL as Back-End to store my data of result of students. Both softwares used for developing project work is specified by CBSE and freely available as free and open source.

# CODING

```
import mysql.connector as mysql  
connection=mysql.connect(host="localhost",user="root",pass  
wd="2005")  
if connection.is_connected():  
    print("Connection established")  
else:  
    print("Connection Error! Kindly check.")  
  
cursor=connection.cursor()  
cursor.execute("Create database Result")  
cursor.execute("Use Result")  
cursor.execute("Create table personalrecord (RollNo  
Integer,CandidateName Varchar(50), MotherName  
Varchar(50),FatherName varchar(50),DOB Date)")  
cursor.execute("Create table examrecord(Stream  
varchar(30),TotalMarks integer,Grade char(2),RollNo Integer)  
")  
  
ch=1  
while ch!=0:  
    print("-----Menu-----")  
    print("Enter 1 to add record")  
    print("Enter 2 to show record")  
    print("Enter 3 to edit record")  
    print("Enter 4 to show list")  
    print("Enter 0 to exit")  
    print("-----")  
    ch=int(input("Enter your choice:"))
```

```

if ch==1:
    RollNo=int(input("Enter Roll No.:"))
    Name=input("Enter candidate name:")
    FatherName=input("Enter Father's Name:")
    MotherName=input("Enter Mother's Name:")
    DOB=input("Enter Candidate's DOB:")
    query="Insert into personalrecord
values({},'{}','{}','{}','{}')".format(RollNo,Name,FatherName
,MotherName,DOB)
    cursor.execute(query)
    Stream=input("Enter your stream:")
    TotalMarks=float(input("Enter total Marks:"))
    Grade=input("Enter ur grade:")
    Query="Insert into examrecord
values('{}',{},{},{})".format(Stream,TotalMarks,Grade,Roll
No)
    cursor.execute(Query)
    connection.commit()

elif ch==2:
    Candidatename=input("Enter Candidate Name:")
    Query="Select * from personalrecord,examrecord where
CandidateName='{}' and
personalrecord.RollNo=examrecord.RollNo".format(Candidat
ename)
    cursor.execute(Query)
    data=cursor.fetchall()
    print("RollNo CandidateName FatherName MotherName
DOB Stream TotalMarks Grade RollNo")
    for i in data:
        print(i,end="\n")

```

```

print("\n")

elif ch==3:
    RollNo=int(input("Enter RollNo of candidate:"))

    ans="y"
    while ans!="n":

        print("-----Menu-----")
        print("Enter a to edit Candidate's name")
        print("Enter b to edit Candidate Father's name")
        print("Enter c to edit Candidate Mother's name")
        print("Enter d to edit DOB")
        print("Enter e to edit TotalMarks:")
        print("Enter f to edit Grade:")
        print("Enter g to edit full data")
        print("Enter n to exit")

        print("-----")
        ans=input("Enter your choice:")

        if ans=="a":
            Name=input("Enter candidate correct name:")
            Query="Update personalrecord set
candidatename='{}' where RollNo={}".format(Name,RollNo)
            cursor.execute(Query)

        elif ans=="b":

```

```
FatherName=input("Enter Candidate Father Correct  
Name:")
```

```
Query="Update personalrecord set FatherName='{}'  
where RollNo={}".format(FatherName,RollNo)  
cursor.execute(Query)
```

```
elif ans=="c":
```

```
MotherName=input("Enter Candidate Mother  
Correct Name:")
```

```
Query="Update personalrecord set  
MotherName='{}' where  
RollNo={}".format(MotherName,RollNo)  
cursor.execute(Query)
```

```
elif ans=="d":
```

```
DOB=input("Enter correct DOB:")
```

```
Query="Update personalrecord set DOB='{}' where  
RollNo={}".format(DOB,RollNo)  
cursor.execute(Query)
```

```
elif ans=="e":
```

```
TotalMarks=float(input("Enter correct  
TotalMarks:"))
```

```
Query="Update examrecord set TotalMarks={}  
where RollNo={}".format(TotalMarks,RollNo)  
cursor.execute(Query)
```

```
connection.commit()
```

```
query="Update examrecord set Grade='{}' where  
RollNo={}".format(Grade,RollNo)
```

```
cursor.execute(query)
```

```
connection.commit()
```

```
elif ans=="f":  
    Grade=input("Enter candidate's grade:")  
    Query="Update examrecord set Grade='{}' where  
RollNo={}".format(Grade,RollNo)  
    cursor.execute(Query)  
    connection.commit()  
  
elif ans=="g":  
    Name=input("Enter candidate name:")  
    FatherName=input("Enter Father's Name:")  
    MotherName=input("Enter Mother's Name:")  
    DOB=input("Enter Candidate's DOB:")  
    Query="Delete from personalrecord where  
RollNo={}".format(RollNo)  
    cursor.execute(Query)  
    connection.commit()  
    Query="Insert into personalrecord  
values('{}','{}','{}','{}','{}','{}')".format(RollNo,Name,FatherN  
ame,MotherName,DOB)  
    cursor.execute(Query)  
    connection.commit()  
    Stream=input("Enter your stream:")  
    TotalMarks=float(input("Enter total Marks:"))  
    Grade=input("Enter candidate's Grade:")  
    Query="Delete from examrecord where  
RollNo={}".format(RollNo)  
    cursor.execute(Query)  
    connection.commit()  
    Query="Insert into examrecord  
values('{}','{}','{}','{}')".format(Stream,TotalMarks,Grade,Roll  
No)  
    cursor.execute(Query)
```

```
connection.commit()

elif ans=="n":
    print("Your edit is done successfully")

else:
    print("Invalid choice chosen")

elif ch==4:
    cursor.execute("Select * from
personalrecord,examrecord where
personalrecord.RollNo=examrecord.RollNo")
    data=cursor.fetchall()
    for i in data:
        print(i,end="\n")
        print("\n")
else:
    print("Your Result Analysis is completed")
```

```

import mysql.connector as mysql
connection=mysql.connect(host="localhost",user="root",passwd="2005")
if connection.is_connected():
    print("Connection established")
else:
    print("Connection Error! Kindly check.")

cursor=connection.cursor()
cursor.execute("Create database Result")
cursor.execute("Use Result")
cursor.execute("Create table personalrecord (RollNo Integer,CandidateName Varchar(50), MotherName Varchar(50),FatherName varchar(50),DOB Date)")
cursor.execute("Create table examrecord(Stream varchar(30),TotalMarks integer,Grade char(2),RollNo Integer) ")

ch=1
while ch!=0:
    print("=====Menu====")
    print("Enter 1 to add record")
    print("Enter 2 to show record")
    print("Enter 3 to edit record")
    print("Enter 4 to show list")
    print("Enter 0 to exit")
    print("=====")
    ch=int(input("Enter your choice:"))

    if ch==1:
        RollNo=int(input("Enter Roll No."))
        Name=input("Enter candidate name:")
        FatherName=input("Enter Father's Name:")
        MotherName=input("Enter Mother's Name:")
        DOB=input("Enter Candidate's DOB:")
        query="Insert into personalrecord values({},{},{},{},{})".format(RollNo,Name,FatherName,MotherName,DOB)
        cursor.execute(query)
        Stream=input("Enter your stream:")
        TotalMarks=float(input("Enter total Marks:"))
        Grade=input("Enter ur grade:")
        Query="Insert into examrecord values({},{},{},{},{})".format(Stream,TotalMarks,Grade,RollNo)
        cursor.execute(Query)
        connection.commit()

    elif ch==2:
        CandidateName=input("Enter Candidate Name:")
        CandidateName=input("Enter Candidate Name:")
        Query="Select * from personalrecord,examrecord where CandidateName='{}' and personalrecord.RollNo=examrecord.RollNo".format(CandidateName)
        cursor.execute(Query)
        data=cursor.fetchall()
        print("RollNo CandidateName FatherName MotherName DOB Stream TotalMarks Grade RollNo")
        for i in data:
            print(i,end="\n")
        print("\n")

    elif ch==3:
        RollNo=int(input("Enter RollNo of candidate:"))

        ans="y"
        while ans!="n":
            print("=====Menu====")
            print("Enter a to edit Candidate's name")
            print("Enter b to edit Candidate Father's name")
            print("Enter c to edit Candidate Mother's name")
            print("Enter d to edit DOB")
            print("Enter e to edit TotalMarks:")
            print("Enter f to edit Grade:")
            print("Enter g to edit full data")
            print("Enter n to exit")
            print("=====")
            ans=input("Enter your choice:")

            if ans=="a":
                Name=input("Enter candidate correct name:")
                Query="Update personalrecord set candidatename='{}' where RollNo={}".format(Name,RollNo)
                cursor.execute(Query)

            elif ans=="b":
                FatherName=input("Enter Candidate Father Correct Name:")
                Query="Update personalrecord set FatherName='{}' where RollNo={}".format(FatherName,RollNo)
                cursor.execute(Query)

            elif ans=="c":
                MotherName=input("Enter Candidate Mother Correct Name:")
                Query="Update personalrecord set MotherName='{}' where RollNo={}".format(MotherName,RollNo)
                cursor.execute(Query)

```

```

    elif ans=="d":
        DOB=input("Enter correct DOB:")
        Query="Update personalrecord set DOB='{}' where RollNo={}".format(DOB,RollNo)
        cursor.execute(Query)

    elif ans=="e":
        TotalMarks=float(input("Enter correct TotalMarks:"))
        Query="Update examrecord set TotalMarks={} where RollNo={}".format(TotalMarks,RollNo)
        cursor.execute(Query)
        connection.commit()
        query="Update examrecord set Grade='{}' where RollNo={}".format(Grade,RollNo)
        cursor.execute(query)
        connection.commit()

    elif ans=="f":
        Grade=input("Enter candidate's grade:")
        Query="Update examrecord set Grade='{}' where RollNo={}".format(Grade,RollNo)
        cursor.execute(Query)
        connection.commit()

    elif ans=="g":
        Name=input("Enter candidate name:")
        FatherName=input("Enter Father's Name:")
        MotherName=input("Enter Mother's Name:")
        DOB=input("Enter Candidate's DOB:")
        Query="Delete from personalrecord where RollNo={}".format(RollNo)
        cursor.execute(Query)
        connection.commit()
        Query="Insert into personalrecord values('{}','{}','{}','{}','{}')".format(RollNo,Name,FatherName,MotherName,DOB)
        cursor.execute(Query)
        connection.commit()
        Stream=input("Enter your stream:")
        TotalMarks=float(input("Enter total Marks:"))
        Grade=input("Enter candidate's Grade:")
        Query="Delete from examrecord where RollNo={}".format(RollNo)
        cursor.execute(Query)
        connection.commit()
        Query="Insert into examrecord values('{},{}','{}','{}')".format(Stream,TotalMarks,Grade,RollNo)
        cursor.execute(Query)
        connection.commit()

    elif ans=="n":
        print("Your edit is done successfully")

    else:
        print("Invalid choice chosen")

elif ch==4:
    cursor.execute("Select * from personalrecord,examrecord where personalrecord.RollNo=examrecord.RollNo")
    data=cursor.fetchall()
    for i in data:
        print(i,end="\n")
        print("\n")
else:
    print("Your Result Analysis is completed")

```

# OUTPUT

```
>>>
= RESTART: C:\Users\user\AppData\Local\Programs\Python\Python310\Result_Analysis_Khushi_Dua.py
Connection established
=====Menu=====
Enter 1 to add record
Enter 2 to show record
Enter 3 to edit record
Enter 4 to show list
Enter 0 to exit
=====
Enter your choice:1
Enter Roll No.:1
Enter candidate name:Abc
Enter Father's Name:xyz
Enter Mother's Name:mno
Enter Candidate's DOB:2006-01-03
Enter your stream:Science
Enter total Marks:100
Enter ur grade:A
=====Menu=====
Enter 1 to add record
Enter 2 to show record
Enter 3 to edit record
Enter 4 to show list
Enter 0 to exit
=====
Enter your choice:2
Enter Candidate Name:abc
RollNo CandidateName FatherName MotherName DOB Stream TotalMarks Grade RollNo
(1, 'Abc', 'xyz', 'mno', datetime.date(2006, 1, 3), 'Science', 100, 'A', 1)

=====Menu=====
Enter 1 to add record
Enter 2 to show record
Enter 3 to edit record
Enter 4 to show list
Enter 0 to exit
=====
Enter your choice:1
Enter Roll No.:2
Enter Roll No.:2
Enter candidate name:Xyz
Enter Father's Name:Pqr
Enter Mother's Name:Uvw
Enter Candidate's DOB:2005-01-08
Enter your stream:Science
Enter total Marks:99
Enter ur grade:A
=====Menu=====
Enter 1 to add record
Enter 2 to show record
Enter 3 to edit record
Enter 4 to show list
Enter 0 to exit
=====
Enter your choice:3
Enter RollNo of candidate:1
=====Menu=====
Enter a to edit Candidate's name
Enter b to edit Candidate Father's name
Enter c to edit Candidate Mother's name
Enter d to edit DOB
Enter e to edit TotalMarks:
Enter f to edit Grade:
Enter g to edit full data
Enter n to exit
=====
Enter your choice:a
Enter candidate correct name:uvw
=====Menu=====
Enter a to edit Candidate's name
Enter b to edit Candidate Father's name
Enter c to edit Candidate Mother's name
Enter d to edit DOB
Enter e to edit TotalMarks:
Enter f to edit Grade:
Enter g to edit full data
Enter n to exit
=====
Enter your choice:b
Enter Candidate Father Correct Name:pqr
=====Menu=====
Enter a to edit Candidate's name
```

```

Enter Candidate Father Correct Name:pqr
=====Menu=====
Enter a to edit Candidate's name
Enter b to edit Candidate Father's name
Enter c to edit Candidate Mother's name
Enter d to edit DOB
Enter e to edit TotalMarks:
Enter f to edit Grade:
Enter g to edit full data
Enter n to exit
=====
Enter your choice:c
Enter Candidate Mother Correct Name:abc
=====Menu=====
Enter a to edit Candidate's name
Enter b to edit Candidate Father's name
Enter c to edit Candidate Mother's name
Enter d to edit DOB
Enter e to edit TotalMarks:
Enter f to edit Grade:
Enter g to edit full data
Enter n to exit
=====
Enter your choice:d
Enter correct DOB:2005-03-01
=====Menu=====
Enter a to edit Candidate's name
Enter b to edit Candidate Father's name
Enter c to edit Candidate Mother's name
Enter d to edit DOB
Enter e to edit TotalMarks:
Enter f to edit Grade:
Enter g to edit full data
Enter n to exit
=====
Enter your choice:e
Enter correct TotalMarks:99
=====Menu=====
Enter a to edit Candidate's name
Enter b to edit Candidate Father's name
Enter c to edit Candidate Mother's name
Enter d to edit DOB
Enter e to edit TotalMarks:
Enter f to edit Grade:
Enter g to edit full data
Enter n to exit
=====
Enter your choice:f
Enter candidate's grade:A+
=====Menu=====
Enter a to edit Candidate's name
Enter b to edit Candidate Father's name
Enter c to edit Candidate Mother's name
Enter d to edit DOB
Enter e to edit TotalMarks:
Enter f to edit Grade:
Enter g to edit full data
Enter n to exit
=====
Enter your choice:n
Your edit is done successfully
=====Menu=====
Enter 1 to add record
Enter 2 to show record
Enter 3 to edit record
Enter 4 to show list
Enter 0 to exit
=====
Enter your choice:4
(1, 'uvw', 'abc', 'pqr', datetime.date(2005, 3, 1), 'Science', 99, 'A+', 1)

(2, 'Xyz', 'Pqr', 'Uvw', datetime.date(2005, 1, 8), 'Science', 99, 'A', 2)

=====Menu=====
Enter 1 to add record

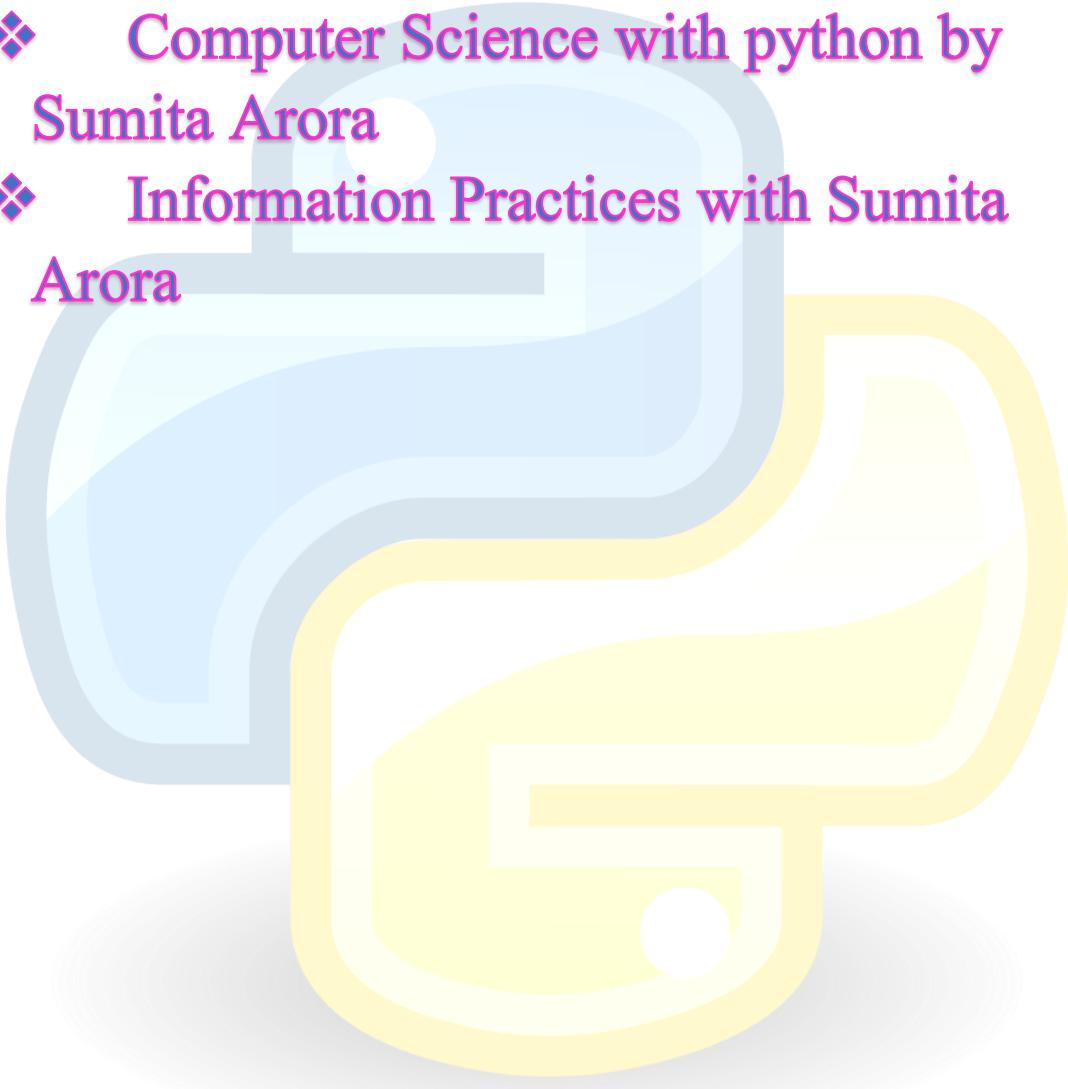
```

```
-----  
Enter 2 to show record  
Enter 3 to edit record  
Enter 4 to show list  
Enter 0 to exit  
=====  
Enter your choice:3  
Enter RollNo of candidate:2  
=====Menu=====  
Enter a to edit Candidate's name  
Enter b to edit Candidate Father's name  
Enter c to edit Candidate Mother's name  
Enter d to edit DOB  
Enter e to edit TotalMarks:  
Enter f to edit Grade:  
Enter g to edit full data  
Enter n to exit  
=====  
Enter your choice:g  
Enter candidate name:uvw  
Enter Father's Name:xyz  
Enter Mother's Name:pqr  
Enter Candidate's DOB:2005-08-09  
Enter your stream:Science  
Enter total Marks:100  
Enter candidate's Grade:A  
=====Menu=====  
Enter a to edit Candidate's name  
Enter b to edit Candidate Father's name  
Enter c to edit Candidate Mother's name  
Enter d to edit DOB  
Enter e to edit TotalMarks:  
Enter f to edit Grade:  
Enter g to edit full data  
Enter n to exit  
=====  
Enter your choice:n  
Your edit is done successfully  
=====Menu=====
```

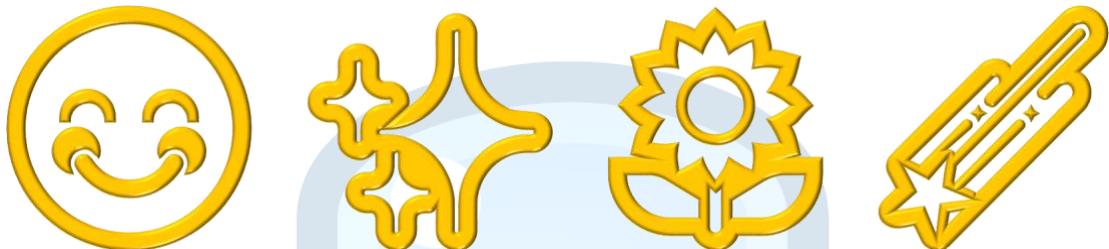
```
Enter 1 to add record  
Enter 2 to show record  
Enter 3 to edit record  
Enter 4 to show list  
Enter 0 to exit  
=====  
Enter your choice:4  
(1, 'uvw', 'abc', 'pqr', datetime.date(2005, 3, 1), 'Science', 99, 'A+', 1)  
  
(2, 'uvw', 'xyz', 'pqr', datetime.date(2005, 8, 9), 'Science', 100, 'A', 2)  
  
=====Menu=====  
Enter 1 to add record  
Enter 2 to show record  
Enter 3 to edit record  
Enter 4 to show list  
Enter 0 to exit  
=====  
Enter your choice:0  
Your Result Analysis is completed
```

# BIBLIOGRAPHY

- ❖ Computer Science with python by Sumita Arora
- ❖ Information Practices with Sumita Arora



# THANK YOU



*Remarks :*

*Internal Examiner  
Signature*

*External Examiner  
Signature*