AMRITA SCHOOL OF ENGINEERING, CHENNAI

337/1A, vallal RCK Nagar, Vengal, Tamil Nadu-601 103

ADVANCED PROGRAMMING

PROJECT REPORT ON

STUDENT REPORT CARD MANAGEMENT SYSTEM

Prepared by:

ROHAN REDDY [CH.EN.U4CSE20009]

HIMESH [CH.EN.U4CSE20021]

CHARAN REDDY [CH.EN.U4CSE20022]

PRIYANKA VARMA [CH.EN.U4CSE20038]

CSE -A, IInd Year

TEAM 15

Submitted to:

Dr. A Padmavathi –

Asst. Professor-[CSE]

Subject Code:

19CSE201

TABLE OF CONTENTS

S.No:	TOPIC	PG.NO:
1.	Acknowledgement	3
2.	Abstract	4
3.	Introduction of project	5
4.	Feasibility Study	6
5.	About the project	7
6.	Software and Hardware requirement	7
7.	Database-MySQL	8
8.	Python source code	9-22
9.	Python Modules Used	23
10.	Working of Student Marks Report Card Python Project and output	23-29
11.	System Design	30
12.	Advantages of the project	31
13.	Limitations of the project	31
14.	Conclusion	32

ACKNOWLEDGEMENT

Any accomplishment requires the efforts of many people and thus our project is no different. We TEAM-15 of class CSE A 2nd Year, would like to express our sincere gratitude to our computer science professor Dr. Padmavati and Dr. Raghupathy, for their vital support, guidance and encouragement, without which this project would not have come forth.

We would also like to extend my gratitude to our chairperson Dr. Prasanna Kumar, principal Dr. Shankar who have always encouraged us. We would like to express our gratitude towards them, for there Constant encouragement, supports and guidance. We also thank all our peers for helping us whenever we asked for.

Last but not the least we all thank our parents and Almighty for their blessings.

ABSTRACT

Our project Student Report Card Management System includes registration of the new students and student marks, storing details info into the system i.e., computerized process.

Our software has the facility to give unique id for every system and stores the data and details of every student in Database (MySQL server). It mainly includes storing and modifying the marks data obtained by each student in the term examinations. It also includes a search facility also- search by name, roll number, student term marks. The data can be retrieved easily.

The interface is very user-friendly. The data are well protected for personal use and makes the data Processing very fast.

INTRODUCTION

Student Marks Report Card Python Project

This is a project based on Repot Card Management System. The program helps us to enter, display or alter the details of different students report marks.

This project STUDENT REPORT CARD SYSTEM includes facilities of registration, search, display, modification, deletion of student information about the marks in different sessions and different term examinations. This software searches the student information on the basis of roll number or by name of the student or father name of the student which is stored in the record in database. The software used for small schools for maintaining their records related to report card and marks of student and cost savings.

Moreover, and most importantly the program helps us to get the results and class wise or session wise report and has the connectivity to the Database.

Data file handling has been effectively used in the program. Database is a collection of interrelated data to serve multiple applications, i.e., database programs create files of information. So, we see that files are worked with most inside the program itself.

DBMS:

The software required for management of data is called DBMS. It has three models

- **Relation model:** It stores information in form of rows(cardinality) and columns(degree).
- **Hierarchical model:** In this type of model, we have multiple records inside a single record.
- **Network model:** In this, the data is represented by collections of record and relationship is represented by associations.

CHARACTERISTICS OF DBMS:

- It reduces the redundancy
- Data sharing
- Data standardization
- Reduction of data in inconsistency

FEASIBILITY STUDY

The objective of feasibility study is to determine whether or not the proposed system is feasible. The feasibility is determined in terms of three aspects.

These are:

- **Technical Feasibility**

In this, one has to test whether the system can be developed using existing technology or not. We have used Python as front end. It is evident that necessary hardware and software are available for development and implementation of proposed system. We acquire the technical knowledge of working in Python language, and then only we have started designing our project.

- **Economic Feasibility**

As a part of this, the costs and benefits associated with the proposed system are compared and the project is economically e-feasible only if tangible and intangible benefits outweigh the cost.

The cost for proposed student's report card management is outweigh the cost and efforts involved in maintaining the registers, files and generations of various report. The system also reduces the administrative and technical staff to do various jobs that single software can do, So, this system is economically feasible.

- **Operational Feasibility**

When it is found that the project is both economic and technical feasible, the next step is to determine whether it is operationally feasible or not. Operational feasibility depends upon human resources for the development and implementation of the system. User involvement is more required. This project also satisfies the Operational feasibility.

ABOUT THE PROJECT

The Main Aim of the project is to create a Python code and Database for **Student Report Card Management System.**

Marks report card is one such project that almost every teacher needs to records the marks obtained by the students. The database of this project is MySQL.

Student Report Card Management System is a Python program that record and calculate all the student’s grade. The system was developed using only Python at the frontend language and MySQL at the backend. The system is very straightforward, the user is freely to view and display the list of all the students. In teacher side the user can add new student, modify the student marks, and show the report card of each student. The system has an auto function that automatically calculate the student grade and gives the total marks and percentage of the student in each session or in each term examinations after entering the individual subject marks. The main objective of the project is to display record of individual student and to search student report cards and to search whole class reports.

Software and Hardware requirement:

#Software Specification:

Operating system:	Windows 11/10/8/7
Platform:	Python IDLE 3.9 or VS Code
Database:	MySQL
Languages:	Python

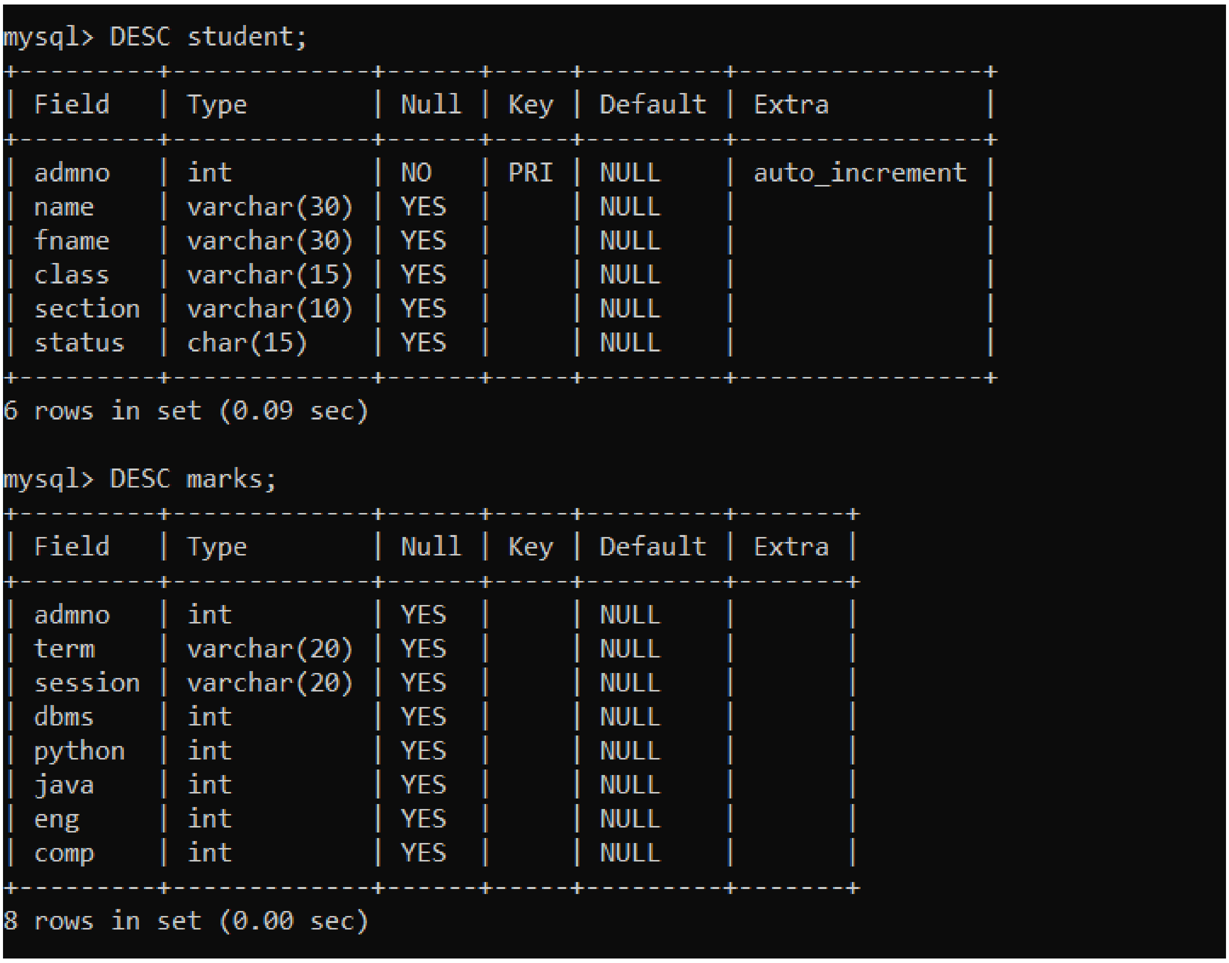
#Hardware specification:

Processor:	Dual core or above
Hard Disk:	40 GB
Ram:	1024MB

DATABASE-MYSQL

Student report card project stores basic information of students and their marks in two separate MySQL tables – **Students and Marks.**

The table structure of the Database Tables:



```
mysql> DESC student;
+---------+-------------+------+-----+---------+----------------+
| Field   | Type        | Null | Key | Default | Extra          |
+---------+-------------+------+-----+---------+----------------+
| admno   | int         | NO   | PRI | NULL    | auto_increment |
| name    | varchar(30) | YES  |     | NULL    |                |
| fname   | varchar(30) | YES  |     | NULL    |                |
| class   | varchar(15) | YES  |     | NULL    |                |
| section | varchar(10) | YES  |     | NULL    |                |
| status  | char(15)    | YES  |     | NULL    |                |
+---------+-------------+------+-----+---------+----------------+
6 rows in set (0.09 sec)

mysql> DESC marks;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| admno   | int         | YES  |     | NULL    |       |
| term    | varchar(20) | YES  |     | NULL    |       |
| session | varchar(20) | YES  |     | NULL    |       |
| dbms    | int         | YES  |     | NULL    |       |
| python  | int         | YES  |     | NULL    |       |
| java    | int         | YES  |     | NULL    |       |
| eng     | int         | YES  |     | NULL    |       |
| comp    | int         | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
8 rows in set (0.00 sec)
```

PYTHON SOURCE CODE

```
#   Project name:   STUDENT REPORT CARD MANAGEMENT SYSTEM
#   Made by:        TEAM 15 CSE-A 2nd Year



import mysql.connector
from prettytable import PrettyTable


# global variable for report
school_name ='Amrita School of Engineering'
school_address ='337/1 A, Valla RCK Nagar, SH 50A, Vengal Village, Tamil
nadu'
school_email = 'info@ch.amrita.edu'
school_phone ='044-27602165'


def clear():
  for _ in range(65):
     print()


def add_student():
    conn = mysql.connector.connect(
        host='localhost', database='report_card', port='3306', user='root',
password='mysql@1111')
    cursor = conn.cursor()
    clear()
    print('Add New Student Screen')
    print('-'*120)
    name = input('Enter student Name : ')
    fname = input('Enter student Father Name  : ')
    clas = input('Enter student Class : ')
    section = input('Enter student section : ')
    sql ='insert into student(name,fname,class,section,status) values (" '+name+'
"," '+fname+' "," '+clas+' "," '+section+' ","active");'
    cursor.execute(sql)
    conn.commit()
    conn.close()
    print('\n\n\n New Student added successfully.....')
    wait=input('\n\n\nPress Enter to continue...')


def add_marks():
    conn = mysql.connector.connect(
        host='localhost', database='report_card', port='3306', user='root',
password='mysql@1111')
    cursor = conn.cursor()
    clear()
    print('Add New marks Screen')
    print('-'*120)
    admno = input('Enter admission NO :')
    term = input('Enter TERM Name : ')
    session = input('Enter session  : ')
    dbms = input('Enter marks in Data Base Management System : ')
    python = input('Enter marks in Python : ')
    java = input('Enter marks in Java : ')
    eng = input('Enter marks in English : ')
    comp = input('Enter marks in Computer Systems and Essentials : ')
    sql = 'insert into marks(admno,term,session,dbms,python,java,eng,comp)
values (' + \

admno+',"'+term+'","'+session+'",'+dbms+','+python+','+java+','+eng+','+comp+
');'
    cursor.execute(sql)
    conn.commit()
    conn.close()
    print('\n\n\n New Marks added successfully.....')
    wait = input('\n\n\nPress Enter to continue...')


def modify_student():
    conn = mysql.connector.connect(
        host='localhost', database='report_card', port='3306', user='root',
password='mysql@1111')
    cursor = conn.cursor()
    clear()
    print('Modify Student Information - Screen')
    print('-'*120)
    admno = input('Enter admission No :')
    print('\n1.   Name  ')
    print('\n2.   Father Name  ')
    print('\n3.   Class  ')
    print('\n4.   Section  ')
    print('\n\n')
    choice = int(input('Enter your choice :'))
    field=''
    if choice ==1:
       field ='name'
    if choice == 2:
        field = 'fname'
    if choice == 3:
        field = 'class'
    if choice == 4:
        field = 'section'
    value =input('Enter new value :')
    sql ='update student set '+field+' ="'+value +'" where admno ='+admno+';'
    cursor.execute(sql)
    conn.commit()
    conn.close()
    print('\n\n\n Student Record Updated.....')
    wait = input('\n\n\nPress Enter to continue......')


def modify_marks():
    conn = mysql.connector.connect(
        host='localhost', database='report_card',port='3306',  user='root',
password='mysql@1111')
    cursor = conn.cursor()
    clear()
    print('Modify Marks - Screen')
    print('-'*120)
    admno = input('Enter admission No :')
    term = input('Enter Term  :')
    session = input('Enter Session  :')
    print('\n1.   Data Base Management System  ')
    print('\n2.   Python  ')
    print('\n3.   Java  ')
    print('\n4.   English  ')
    print('\n5.   Computer System and Essentials  ')
    print('\n\n')
    choice = int(input('Enter your choice :'))
    field = ''
    if choice == 1:
      field = 'dbms'
    if choice == 2:
      field = 'python'
    if choice == 3:
      field = 'java'
    if choice == 4:
      field = 'eng'
    if choice == 5:
      field = 'comp'

    value = input('Enter new value :')
    sql = 'update marks set '+field+' ='+value + ' where admno ='+admno+' AND
term="'+term+'" AND session="'+session+'";'
    cursor.execute(sql)
    conn.commit()
    conn.close()
    print('\n\n\n Marks Updated.....')
    wait = input('\n\n\nPress Enter to continue......')

def search_student(field):
  conn = mysql.connector.connect(
      host='localhost', database='report_card', port='3306', user='root',
password='mysql@1111')
  cursor = conn.cursor()
  sql ='select * from student where '
  msg ='Enter '+field +' :'
  value = input(msg)
  if field=='admno':
     sql = sql + field +'=' +value+';'
  else:
     sql = sql + field +' like "%'+value+'%" or fname like "%'+value +'%";'
  cursor.execute(sql)
  records = cursor.fetchall()
  clear()
  print('Search Result for '+field+' : '+value)
  print('-'*120)
  for record in records:
     print(record)
  conn.close()
  wait = input('\n\n\n Press Enter to continue.....')

def search_marks():
    conn = mysql.connector.connect(
      host='localhost', database='report_card', port='3306', user='root',
password='mysql@1111')
    cursor = conn.cursor()
    admno = input('Enter admission No :')
    session = input('Enter Session  :')
    sql ='select * from marks where admno = '+admno + ' and session
="'+session+'";'
    cursor.execute(sql)
    records = cursor.fetchall()
    clear()
    print('Search Result for Admission No :'+admno +' Session : '+session)
    print('-'*120)
    for record in records:
      print(record)
    conn.close()
    wait = input('\n\n\n Press Enter to continue.....')

def search_menu():
    while True:
      clear()
      print(' S E A R C H    M E N U')
      print('-'*120)
      print("\n1.  Admission No")
      print('\n2.  Name / Father Name')
      print('\n3.  Student Term Marks')
      print('\n4.  back to main')
      print('\n\n')
      choice = int(input('Enter your choice ...: '))
      field=''
      if choice == 1:
        field='admno'
        search_student(field)
      if choice == 2:
        field='name'
        search_student(field)
      if choice == 3:
        search_marks()
      if choice == 4:
        break


def report_single_term():
    conn = mysql.connector.connect(
        host='localhost', database='report_card', port='3306', user='root',
password='mysql@1111')
    cursor = conn.cursor()
    admno = input('Enter admission No :')
    session = input('Enter Session  :')
    term = input('Enter term  :')
    sql ='select s.admno,name,fname,dbms,java,python,eng,comp from  \
          student s,marks m  where s.admno = m.admno and s.admno = '+admno +'
and m.session = "'+session+'" and m.term ="'+term+'";'
    cursor.execute(sql)
    record = cursor.fetchone()
    clear()
    print(school_name)
    print(school_address)
    print('Phone :',school_phone ,' Email :', school_email)
    print('-'*120)
    print('Admno :',record[0],' Name :',record[1], '   Father Name :',record[2])
    print('Session :', session, ' Term :', term)
    print('-'*120)
    print('Subject',' Max_marks','min-marks','marks obtained')
    print('Data Base Management System','100','33',record[3])
    print('Python','100','33',record[4])
    print('Java','100','33',record[5])
    print('English','100','33',record[6])
    print('Computer System and Essentials','100','33',record[7])
    print('-'*120)
    total = record[3]+record[4]+record[5]+record[6]+record[7]
    percentage = total*100/500
    print('Total Marks : ',total,'Percentage Marks :',percentage)
    conn.close()
    wait = input('\n\n\n Press Enter to continue.....')


def report_whole_class():
    conn = mysql.connector.connect(
        host='localhost', database='report_card', port='3306', user='root',
password='mysql@1111')
    cursor = conn.cursor()
    clas = input('Enter Class :')
    section =input ('Enter section :')
    session = input('Enter Session  :')
    term = input('Enter term  :')
    sql ='select s.admno,name,fname,dbms,java,python,eng,comp from  \
          student s, marks m  where s.admno = m.admno  AND s.class="'+clas+'"
AND s.section ="'+section +'" and m.session = "'+session+'" and m.term
="'+term+'";'
    cursor.execute(sql)
    records = cursor.fetchall()
    clear()
    print(school_name)
    print(school_address)
    print('Phone :',school_phone ,' Email :', school_email)
    print('-'*120)
    print('Class Wise Report Card:',clas,'-',section, 'Session : ',session, ' Term :',term)
    print('-'*120)
    t = PrettyTable(['admno', 'Name', 'Father Name', 'dbms', 'python', 'java','Eng','Comp','Total'])
    for idr, name, fname, dbms,python,java,eng,comp in records:
      total = dbms+python+java+eng+comp
      t.add_row([idr, name, fname, dbms, python, java, eng, comp,total])
    print(t)
    conn.close()
    wait = input('\n\n\n Press Enter to continue.....')



def report_whole_session():
    conn = mysql.connector.connect(
        host='localhost', database='report_card',port='3306',  user='root', password='mysql@1111')
    cursor = conn.cursor()
    session = input('Enter Session  :')
    sql = 'select s.admno,name,fname,class, section,term,dbms,java,python,eng,comp from  \
          student s, marks m  where s.admno = m.admno  and m.session = '"+session+"' ORDER BY class,section,term;'
    cursor.execute(sql)
    records = cursor.fetchall()
    clear()
    print(school_name)
    print(school_address)
    print('Phone :', school_phone, ' Email :', school_email,'\n\n')

    print('Whole Session Report Card:', '          Session : ', session)
    print('-'*120)
    t = PrettyTable(['admno', 'Name', 'Father Name','Class','section','Term',
'dbms',
                    'python', 'java', 'Eng', 'Comp', 'Total'])
    for idr, name, fname,clas1,section, term, dbms, python, java, eng, comp in
records:
      total = dbms+python+java+eng+comp
      t.add_row([idr, name, fname,clas1, section, term, dbms, python, java, eng,
comp, total])
    print(t)
    conn.close()
    wait = input('\n\n\n Press Enter to continue.....')



def report_topper_list():
    conn = mysql.connector.connect(
        host='localhost', database='report_card', port='3306', user='root',
password='mysql@1111')
    cursor = conn.cursor()
    session = input('Enter Session  :')
    term = input('Enter Term :')
    clas= input('Enter class  :')
    section= input('Enter section  :')
    sql = 'select s.admno, name, fname,dbms,python,java,eng,comp,
dbms+python+java+eng+comp "Total" from student s, marks m \
          where s.admno = m.admno and class = "'+clas+'" and section =
"'+section+'" and session ="'+session+'" and term="'+term+'" order by total
Desc;'
    cursor.execute(sql)
    records = cursor.fetchall()
    clear()
    print(school_name)
    print(school_address)
    print('Phone :', school_phone, ' Email :', school_email, '\n\n')


    print('T O P P E R S    L I S T \n\n Class :',clas,'          Session : ', session, '
Term :',term)
    #print('-'*120)
    t = PrettyTable(['admno', 'Name', 'Father Name', 'dbms', 'python',
'java','English','Computer', 'Total'])
    for idr, name, fname, dbms, python, java, eng, comp, total in records:
        t.add_row([idr, name, fname, dbms, python, java, eng, comp,total])
    print(t)
    conn.close()
    wait = input('\n\n\n Press Enter to continue.....')



def report_menu():
    while True:
      clear()
      print(' R E P O R T   M E N U ')
      print("\n1.  Single Term report card")
      print('\n2.  Whole class report card')
      print('\n3.  Whole Session report Card ')
      print('\n4.  Class Wise- Toppers')
      print('\n5.  Back to main menu')
      print('\n\n')
      choice = int(input('Enter your choice ...: '))
      if choice == 1:
        report_single_term()
      if choice == 2:
        report_whole_class()
      if choice == 3:
        report_whole_session()
      if choice == 4:
        report_topper_list()
      if choice == 5:
        break



def main_menu():
    while True:
      clear()
      print(' R E P O R T   C A R D   M E N U  ')
      print("\n1.  Add Student")
      print('\n2.  Modify Student Record')
      print('\n3.  Add marks')
```

```
        print('\n4.  Modify Marks')
        print('\n5.  Search Menu')
        print('\n6.  Report Menu')
        print('\n7.  Close application')
        print('\n\n')
        choice = int(input('Enter your choice ...: '))
        if choice == 1:
          add_student()
        if choice == 2:
          modify_student()
        if choice == 3:
          add_marks()
        if choice == 4:
          modify_marks()
        if choice == 5:
          search_menu()
        if choice == 6:
          report_menu()
        if choice == 7:
          break


if __name__ == "__main__":
    main_menu()
```

Python Modules Used

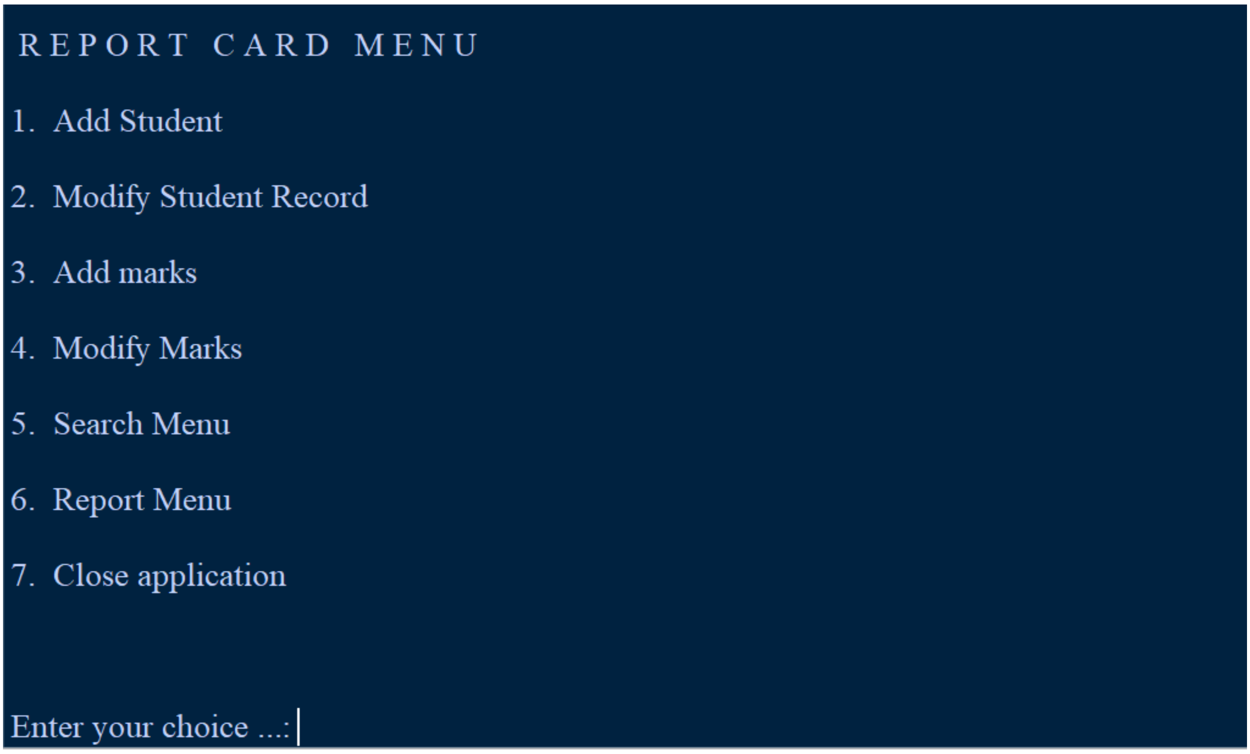
- mysql.connector: To connect Python with MySQL database using its various methods.

- PrettyTable: To display in MySQL select like tables.

Working of Student Marks Report Card Python Project

Whole Project has been divided into 6 major parts and each part of the project has been developed in a Python function. These functions have been called to generate a menu as shown in the output below;

Output of the main report card menu:

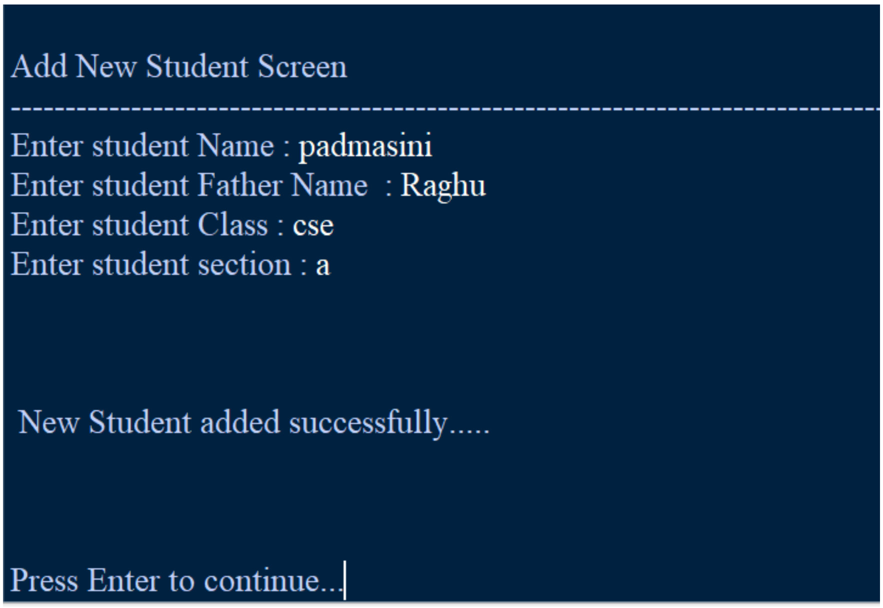


```
REPORT CARD MENU

1.  Add Student

2.  Modify Student Record

3.  Add marks

4.  Modify Marks

5.  Search Menu

6.  Report Menu

7.  Close application



Enter your choice ...:
```

1) Add Student:

When you choose and enter “1” under the menu it will direct to the Add student function which enables the user to input the information of a new student as it directs to the student table in MySQL Database.

Output:



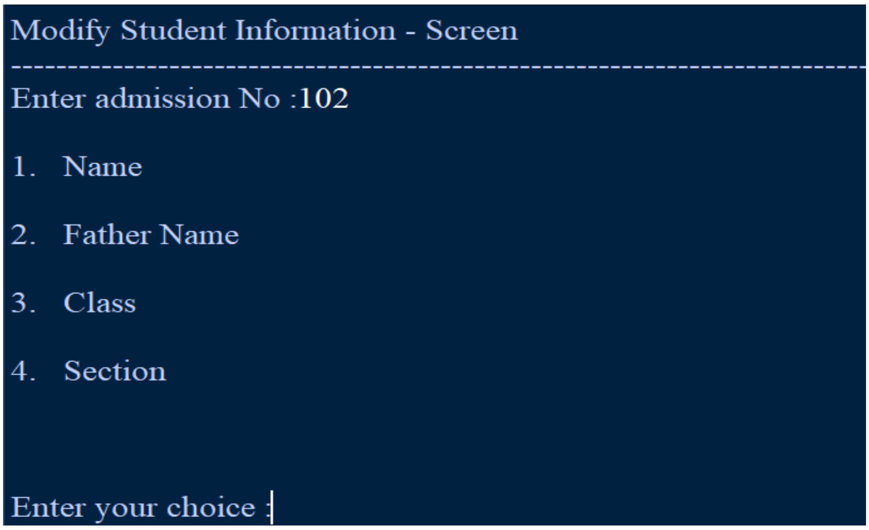
```
Add New Student Screen
------------------------------------------------------------------------------
Enter student Name : padmasini
Enter student Father Name  : Raghu
Enter student Class : cse
Enter student section : a



 New Student added successfully.....



Press Enter to continue...
```

After pressing “Enter” it will again redirect to the main menu of report card

2) Modify Student Record:

This function actually Modifies Current student’s details, First it asks to enter an admission number of a student which the user wants to modify as shown in the output below;

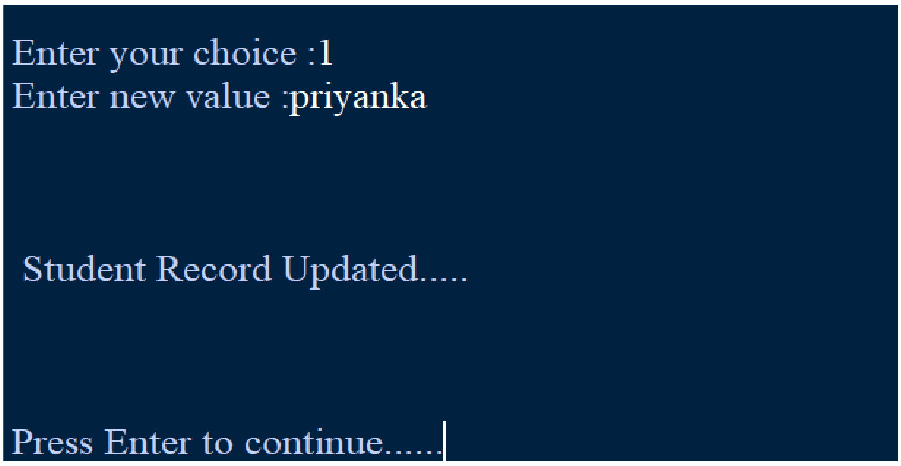
Output:



```
Modify Student Information - Screen
------------------------------------------------------------------------------
Enter admission No :102

1.   Name

2.   Father Name

3.   Class

4.   Section


Enter your choice :
```

After entering the admission number, it again shows which aspect of the student the user wants to modify and the user can modify as they need.

Output:

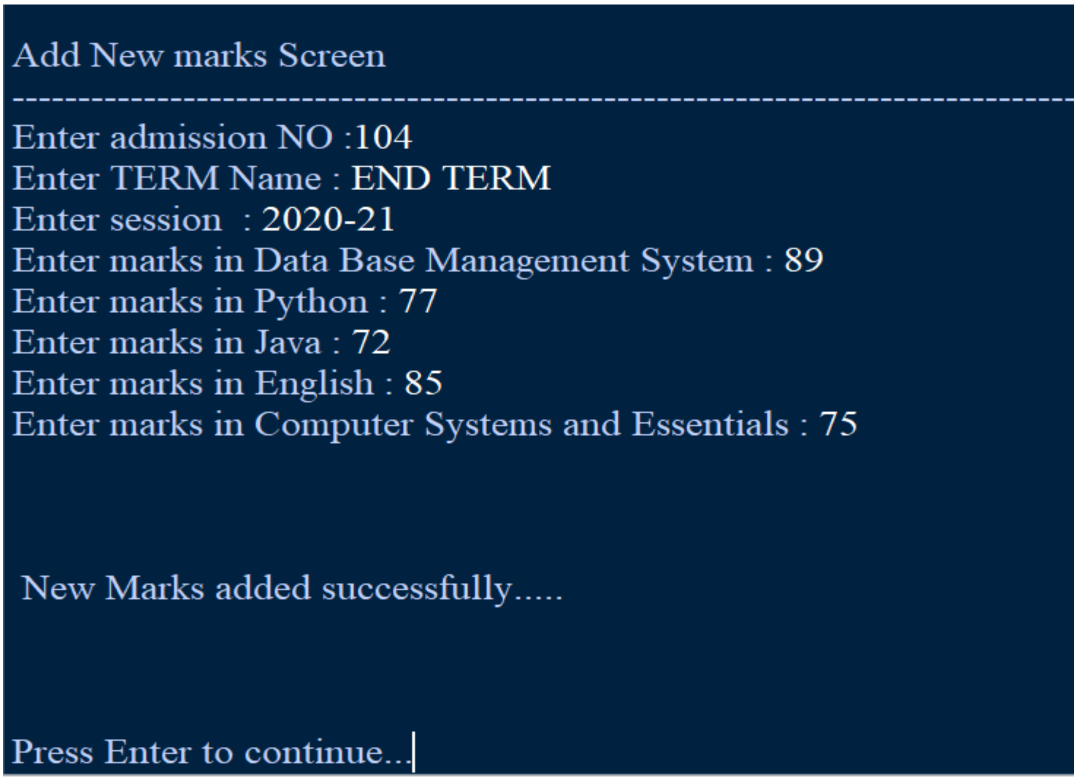


```
Enter your choice :1
Enter new value :priyanka



 Student Record Updated.....



Press Enter to continue......
```

3) <u>Add Marks:</u>

In main menu of report card if you choose number “3” it directs to the function Add Marks where you can add the marks as well as in marks table in the database of a student as per the TERM AND SESSION wise as shown in the output below;

Output:

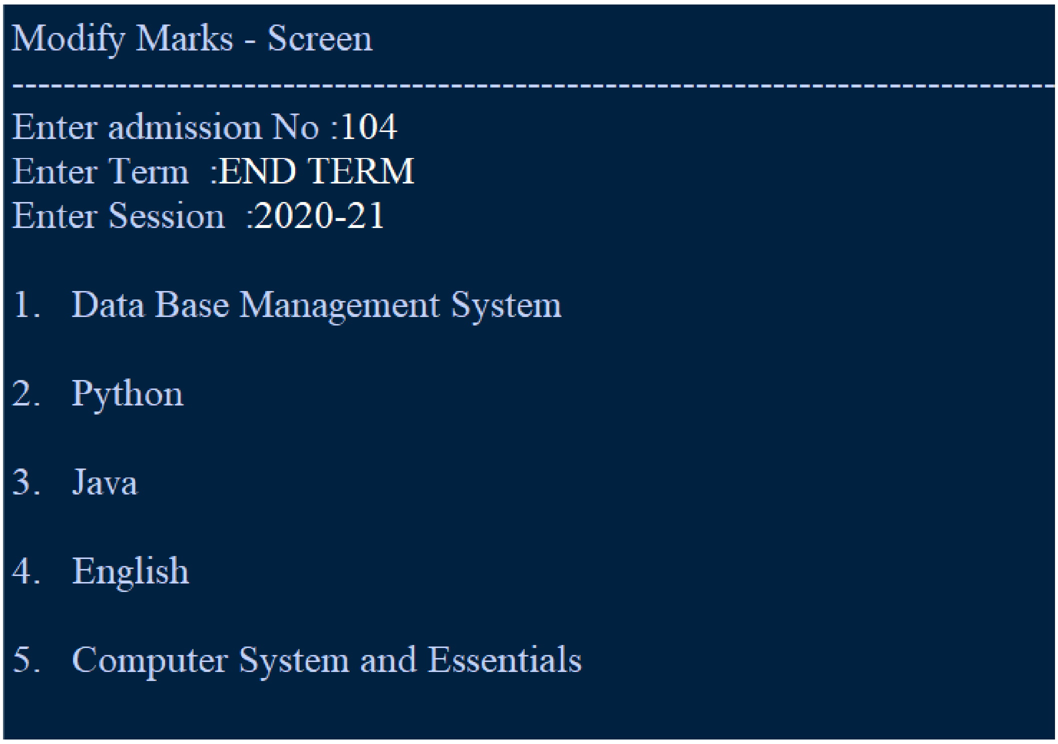


```
Add New marks Screen
------------------------------------------------------------------------------
Enter admission NO :104
Enter TERM Name : END TERM
Enter session  : 2020-21
Enter marks in Data Base Management System : 89
Enter marks in Python : 77
Enter marks in Java : 72
Enter marks in English : 85
Enter marks in Computer Systems and Essentials : 75



 New Marks added successfully.....



Press Enter to continue...
```

4) Modify Marks:

In main menu of report card if you choose number “4” it directs to the function Modify Marks which makes the user to change or modify the existing student marks as well as in the student table in database.

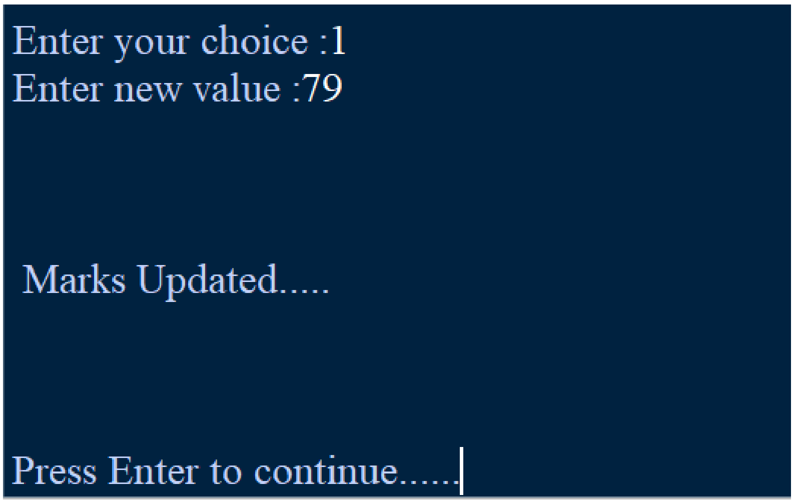
Output:



```
Modify Marks - Screen
--------------------------------------------------------------------------------
Enter admission No :104
Enter Term  :END TERM
Enter Session  :2020-21

1.   Data Base Management System

2.   Python

3.   Java

4.   English

5.   Computer System and Essentials
```

After entering the admission No, Term and session it asks to which subject marks does the user wants to modify after selecting the specific number the user can update new marks of the subject as shown in the output below;

Output:

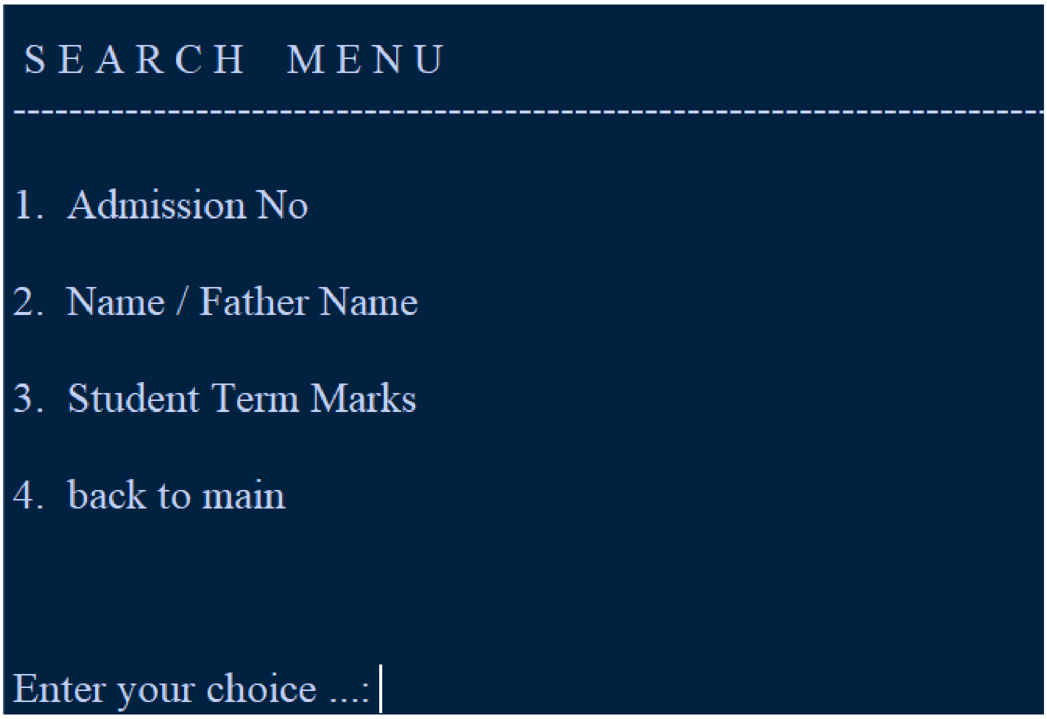


```
Enter your choice :1
Enter new value :79



 Marks Updated.....



Press Enter to continue......
```

5) Search Menu:

In main menu of report card if you choose number “5” it directs to the function Search Menu which makes the user to search the information in the database as shown in the output below;

Output:

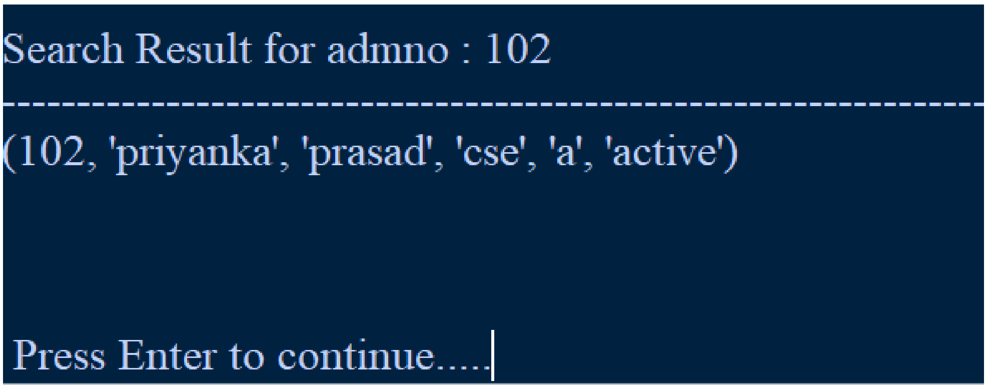


```
SEARCH  MENU
--------------------------------------------------------------------

1.  Admission No

2.  Name / Father Name

3.  Student Term Marks

4.  back to main



Enter your choice ...:
```

This menu provides to search information about the student in either way of choosing admission number or by entering the name or father name of the student.

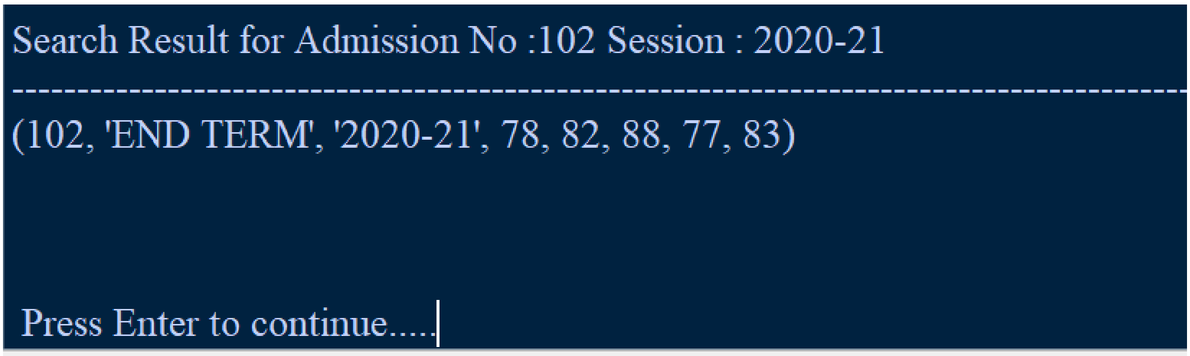
It also can help you to search the information of the marks of the student in the specific Term.

Output 1:



```
Search Result for admno : 102
------------------------------------------------------------
(102, 'priyanka', 'prasad', 'cse', 'a', 'active')



Press Enter to continue.....
```

Output 1.2:

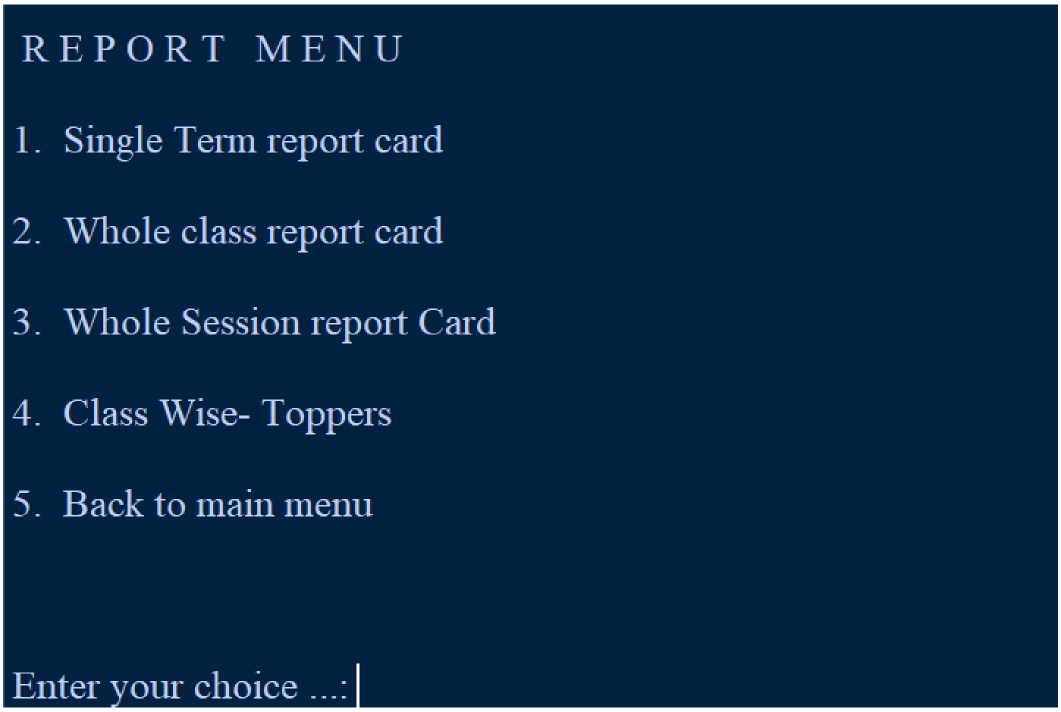


```
Search Result for Admission No :102 Session : 2020-21
--------------------------------------------------------------------------------
(102, 'END TERM', '2020-21', 78, 82, 88, 77, 83)


 Press Enter to continue.....
```

6) Report Menu:

In main menu of report card if you choose number “6” it directs to the function Report Menu which results in providing various types of reports like single student sport, class wise exam reports etc. as shown in the output below;

Output:

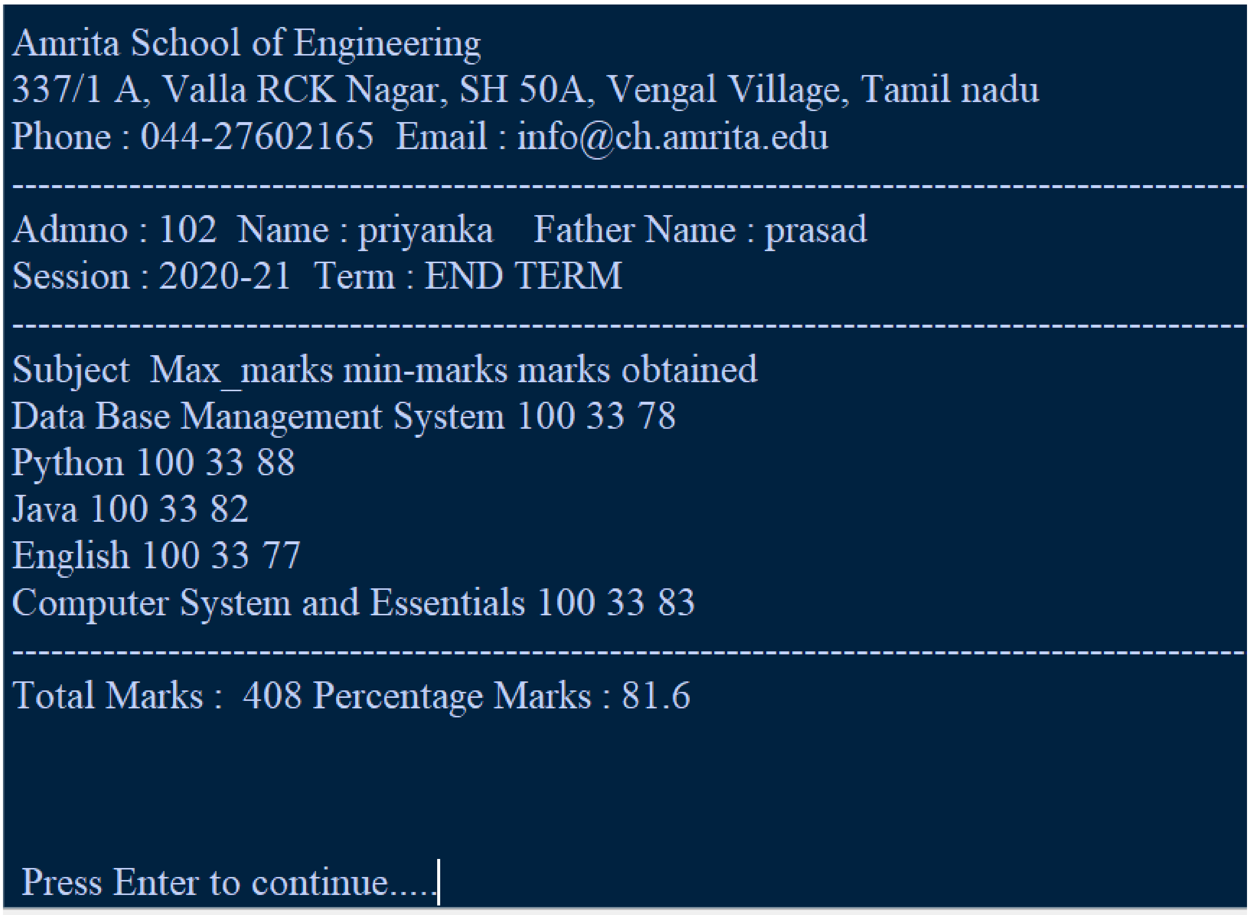


```
 R E P O R T   M E N U

1.  Single Term report card

2.  Whole class report card

3.  Whole Session report Card

4.  Class Wise- Toppers

5.  Back to main menu


Enter your choice ...:
```

As per the user’s choice the function produces the report card of each aspect as shown,

For example, if user choose 1, it gives the Single Term report card as shown in the below output;

Output:



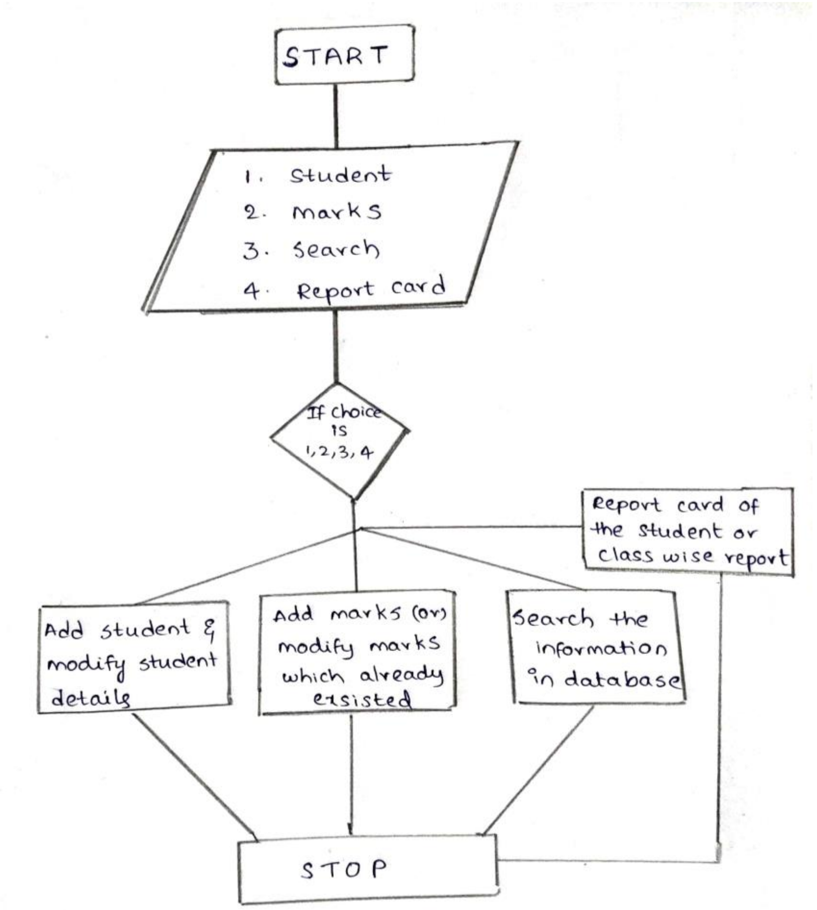
```
Amrita School of Engineering
337/1 A, Valla RCK Nagar, SH 50A, Vengal Village, Tamil nadu
Phone : 044-27602165  Email : info@ch.amrita.edu
------------------------------------------------------------------------------------------------
Admno : 102  Name : priyanka    Father Name : prasad
Session : 2020-21  Term : END TERM
------------------------------------------------------------------------------------------------
Subject  Max_marks min-marks marks obtained
Data Base Management System 100 33 78
Python 100 33 88
Java 100 33 82
English 100 33 77
Computer System and Essentials 100 33 83
------------------------------------------------------------------------------------------------
Total Marks :  408 Percentage Marks : 81.6



 Press Enter to continue.....
```

SYSTEM DESIGN

System design is a solution, a “HOW TO” approach to the creation of a new system. It translates system requirements into ways by which they can be made operational. It is a translational from a user-oriented document to a document-oriented programmers.

For that, it provides the understanding and procedural details necessary for the implementation. Here we use Flowchart to supplement the working of the new system. The system thus made should be reliable, durable and above all should have least possible maintenance costs. It should overcome all the drawbacks of the Old existing system of manual work and most important of all meet the user requirements.

FLOWCHART


START
1. Student
2. marks
3. search
4. Report card
If choice is 1,2,3,4
Report card of the student or class wise report
Add student & modify student details
Add marks (or) modify marks which already exsisted
Search the information in database
STOP

ADVANTAGES OF THE PROJECT

- We can edit the marks of the student easily and generate the report again
- It saves the time of the teacher to calculate the percentage and the grade or rank and pass or fail
- This software reduces paper work.
- It is easy to handle student's record.
- This software saves the time.
- Information of each student stores permanently in the dataset unless the changes are made by the user.

LIMITATIONS OF THE PROJECT

Though the project is complete in all sense still it has some scope of improvement. Few of them as follows:

- This is a command-based Python interface. It can be developed in either GUI (Graphical User Interface) or Web interface using Django.
- Field level and form level validations not implemented.
- Restricted to only five subjects.
- This System can applicable for small institutions.

CONCLUSION

This Project is focused on creating an automated students result management system using MySQL database and Python Language as frontend. This is a computerized examinations results management system for tertiary student’s examination records.

The manual method of students’ academic result processing was found to be tedious, especially when carried out for a large number of students, this makes the entire process time-consuming and error prone. The system designed is meant to register students as soon as they have paid their departmental registration and only then will they be able to view their results. The system presents a single platform that will be used to manage the processing of all examination records within the institution.