

**PM SHRI KENDRIYA VIDYALAYA  
BERHAMPUR**



**COMPUTER SCIENCE PROJECT**

**2023-2024**

**PROJECT TOPIC :**

**STUDENT MANAGEMENT SYSTEM**

**SUBMITTED BY :**

**NAME: ANKITA BEHERA**

**CLASS: XII A**

**CBSE ROLL NUMBER:**

**SUBMITTED TO:**

**SAROJ KANTA MISRA PGT, CS**

# **CERTIFICATE**

This is to certify that **ANKITA BEHERA** of class: **XII A** of **PM SHREE KENDRIYA VIDYALAYA BERHAMPUR** has done her Project on **STUDENT MANAGEMENT SYSTEM** under my supervision. She has Taken interest and has shown at most Sincerity in completion of this project.

I certify this project up to my expectation & as per guidelines issued by **CBSE, NEW DELHI**.

Internal Examiner

External Examiner

Principal

**ACKNOWLEDGEMENT**

**It is with pleasure that I acknowledge my Sincere gratitude to our teacher, MR. SAROJ KANTA MISRA who taught and Undertook the responsibility of teaching The subject computer science. I have Been greatly benefited from his classes.**

**I am especially indebted to our Principal MR. SHIVA PRIYA DASH who has always been A source of encouragement and support And without whose inspiration this project Would not have been a successful I would Like to place on record heartfelt thanks to Him.**

**Finally, I would like to express my Sincere appreciation for all the other Students for my batch their friendship & the fine time that we all shared Together.**

# **HARDWARES AND SOFTWARES** **REQUIRED**

## **HARDWARES**

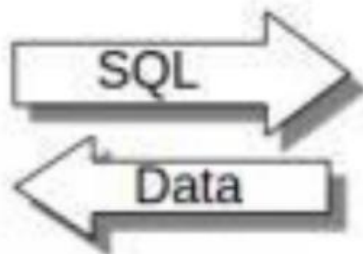
1. Desktop Computer / Laptop
2. Mobile Phone

## **SOFTWARES**

1. Python (latest version)
2. MySQL
3. Python Connector Module



Python



Database System

# **CONTENTS**

<b>S.No.</b>	<b>TOPIC</b>	<b>PAGE NO</b>
<b>1</b>	<b>Certificate</b>	<b>2</b>
<b>2</b>	<b>Acknowledgement</b>	<b>3</b>
<b>3</b>	<b>Hardware's and Software's required</b>	<b>4</b>
<b>4</b>	<b>Introduction</b>	<b>6</b>
<b>5</b>	<b>Python source code</b>	<b>13</b>
<b>6</b>	<b>MySQL Database</b>	<b>29</b>
<b>7</b>	<b>Outputs</b>	<b>33</b>
<b>8</b>	<b>References</b>	<b>39</b>

# **INTRODUCTION**

The real meaning of educational administration has undergone a radical transformation in recent years, with an increasing reliance on technology to streamline student management processes. The provided Python script embodies a Student Management System that leverages the capabilities of MySQL as the backend database. This system caters to the diverse needs of administrators and users, offering functionalities ranging from student details search to updates, views, additions, and deletions.

## **## Evolution of Student Management Systems**

Traditionally, student management involved intricate manual processes, making it susceptible to errors, delays, and inefficiencies. The emergence of digital solutions has ushered in an era of efficiency and accuracy, allowing educational institutions to manage student data seamlessly. Our project aligns with this evolution, providing an interactive and dynamic Student Management System that caters to the specific requirements of administrators and users alike.

## **## Objectives of the Project**

### **1. Administrative Empowerment**

The primary objective of the Student Management System is to empower administrators with tools that facilitate efficient student data management. The system encompasses key administrative functionalities, including:

### ### Student Search

The ``search_student()`` function allows administrators to search for student details based on the student's name. This ensures quick access to comprehensive information, such as admission number, name, gender, class, section, phone number, email ID, and stream.

### ### Update Student Details

The ``update_details()`` function provides administrators with a versatile interface to update various student details, such as name, gender, class, section, phone number, email ID, and stream. The system's flexibility ensures adaptability to changing student information.

### ### View Student Details

The ``view_Marks()`` function facilitates the display of student details, with a particular focus on showcasing student marks. This functionality caters to the academic aspect of student management, allowing administrators to assess and communicate student performance.

### **###Add Student**

The ``add_student()`` function streamlines the process of adding new students to the system, ensuring that vital details are captured efficiently. This contributes to the completeness and accuracy of the student database.

### **### Delete Student**

The ``delete_student()`` function enables administrators to remove student records from the system, providing a mechanism for data cleanup and management.

## **2. User-Friendly Interface**

The project recognizes the importance of user experience for both administrators and end-users. Two distinct panels, namely the Admin Panel and the User Panel, cater to the unique needs of these user groups.



#### **#### Admin Panel**

The ``admin()`` function serves as the gateway to the Admin Panel, offering a menu-driven interface for administrators to access and utilize the various functionalities provided by the system.

#### **#### User Panel**

The ``user()`` function represents the User Panel, providing a simplified interface for end-users to search for their details, view academic information, and exit the system.

### **## Key Code Functionalities Integrated**

#### **### Search and Display Operations**

##### **#### Search Student**

The ``search_student()`` function utilizes SQL queries to search for student details based on the provided name, returning a detailed display of relevant information.

##### **#### View Student Details**

The ``view_Marks()`` function employs SQL queries to fetch and display student details, with a specific focus on academic performance.

### **### Update Operations**

#### **#### Update Student Details**

The ``update_details()`` function integrates user input and SQL queries to selectively update student information, offering granular control over the data modification process.

### **### Data Manipulation Operations**

#### **#### Add Student**

The ``add_student()`` function combines user input and SQL queries to insert new student records into the system, ensuring data completeness.

#### **#### Delete Student**

The ``delete_student()`` function utilizes SQL queries to delete student records based on the provided name, contributing to data management and cleanup.

### **### User and Admin Panels**

#### **#### Admin Panel**

The ``admin()`` function orchestrates the Admin Panel, providing administrators with a menu-driven interface to navigate through the system's functionalities.

#### #### User Panel

The ``user()`` function represents the User Panel, offering end-users a simplified menu to search for their details, view academic information, and exit the system.

### ## Integration of MySQL and Python

#### ### MySQL – The Relational Database Management System

MySQL serves as the backend database for the Student Management System, storing and managing student-related information. The structured schema ensures efficient storage, retrieval, and manipulation of data.

#### ### Python – The Programming Language

Python, with its versatility and readability, powers the logic and user interface of the Student Management System. The script employs Python's database connectivity features to seamlessly integrate with MySQL, providing a cohesive and interactive experience for users and administrators.

## **## Significance of the Project**

The significance of our Student Management System transcends its technical intricacies. In an era where educational institutions are grappling with vast amounts of student data, a robust and user-friendly system is indispensable. Our project addresses this need, offering a solution that not only streamlines administrative tasks but also enhances user experience and contributes to the overall efficiency of student management processes.

In the subsequent sections of this project report, we will delve into the detailed functionalities of the system, explore the database schema, discuss the integration of Python and MySQL, and shed light on the future enhancements and scalability of the Student Management System. Through this comprehensive exploration, we aim to showcase not only the technical prowess of the implemented solution but also the thoughtfulness invested in creating a system that aligns seamlessly with the evolving landscape of educational administration.

# Python Source Code

Import mysql.connector as con

Print("-----  
")

#To search students from student  
details

#table.

Def search\_student():

Print("\*\*\*\*\*STUDENT  
DETAILS\*\*\*\*\*")

K=input("Enter name to search student Name : ")

D=con.connect(host="localhost",user="root",password="admin",database="STUDENT\_MANAGEMENT\_SYSTEM")

C=d.cursor()

c.execute("select\*from STUDENT\_DETAILS where  
NAME like '%{}%'.format(k))

a=c.fetchall()

if len(a)>=1:

for i in a:

print("Addmission no:",i[0])

print("Name is:",i[1])

```
print("Sex:",i[2])
print("Class=",i[3])
print("Sec:",i[4])
print("Phone number is=",i[5])
print("Email_id:",i[6])
print("Stream is:",i[7])
print("-----")
```

else:

```
print("Student Details Not Found")
```

```
d.commit()
```

**#To update details of students from student details table.**

```
Def update_details():
```

```
D=con.connect(host="localhost",user="root",passwor
```

```
d="admin",database="STUDENT_MANAGEMENT_SYSTM")
```

```
C=d.cursor()
```

```
Print("1. Update Name")
```

```
Print("2. Update Gender")
```

```
Print("3. Update class")
```

```
Print("4. Update Section")
```

```
Print("5. Update Phone no.")
```

```
Print("6. Update E-mail id")
```

```
Print("7. Update Stream")
```

```
Opt=int(input("Enter your choice to update"))
```

```
If opt==1:
```

```
Print("-----")
```

```
Print("You are inside updating name.")
```

```
Q=input("Enter ADDMISSION_NUMBER whose  
data you want to update: ")
```

```
L=input("Enter your updated name: ")
```



```
    c.execute("update STUDENT_DETAILS set  
NAME='{}' where  
ADMISSION_NUMBER={}".format(l,q))  
    d.commit()
```

```
elif opt==2:
```

```
    print("-----")  
    print("You are inside updating Gender.")  
    p=input("Enter name whose data you want to  
update: ")  
    l=input("Enter your updated Gender: ")  
    c.execute("update STUDENT_DETAILS set SEX='{}'  
where name='{}'".format(l,p))  
    d.commit()
```

```
elif opt==3:
```

```
    print("-----")  
    print("You are inside updating class.")
```

```
m=input("Enter name whose data you want to  
update: ")
```

```
n=input("Enter your updated class: ")
```

```
c.execute("update STUDENT_DETAILS set  
CLASS={} where name='{}'.format(n,m))
```

```
d.commit()
```

```
elif opt==4:
```

```
print("-----")
```

```
print("You are inside updating section.")
```

```
y=input("Enter name whose data you want to  
update: ")
```

```
e=input("Enter your updated section: ")
```

```
c.execute("update STUDENT_DETAILS set SEC='{}'  
where name='{}'.format(e,y))
```

```
d.commit()
```

```
elif opt==5:
```

```
print("-----")
```

```
print("You are inside updating phonenumber.")
```

```
h=input("Enter name whose data you want to  
update: ")
```

```
r=input("Enter your updated phonenumber: ")
```

```
c.execute("update STUDENT_DETAILS set  
PHONE_NUMBER={} where name='{}'.format(r,h))
```

```
d.commit()
```

```
elif opt==6:
```

```
print("-----")
```

```
print("You are inside updating email_id.")
```

```
j=input("Enter name whose data you want to  
update: ")
```

```
k=input("Enter your updated email_id : ")
```

```
c.execute("update STUDENT_DETAILS set  
EMAIL_ID='{}' where name='{}'.format(k,j))
```

```
d.commit()
```

```
elif opt==7:
```

```
print("-----")
```

```
print("You are inside updating stream.")
```

```
f=input("Enter name whose data you want to update: ")
```

```
z=input("Enter your updated stream : ")
```

```
c.execute("update STUDENT_DETAILS set STREAM='{}' where NAME='{}'.format(z,f))
```

```
d.commit()
```

```
else:
```

```
update_details()
```

**#To display the marks of students from student details table .**

```
Def view_Marks():
```

```
Print("***** VIEW STUDENT DETAILS*****")
```

```
K=input("Enter name to view student details : ")
```

```
D=con.connect(host="localhost",user="root",password="admin",database="student_management_system")
```

```
C=d.cursor()
c.execute("select*from STUDENT_DETAILS where
NAME like '%{}%'".format(k))
s=c.fetchall()
for i in s:
    print("MARKS:",i[8])

    print("-----")

d.commit()
```

**#To insert details to student details table**

```
Def add_student():
```

```
    Print("..... ADD STUDENT
DETAILS.....")
```

```
N=input("Enter student name : ")
```

```
A=int(input("Enter student admission number: "))
```

```
R=input("Enter gender of student: ")
```

```
I=int(input("Enter class of student:"))
```

**P=input("Enter student section : ")**

**T=int(input("Enter student phone number:"))**

**W=input("Enter student stream: ")**

**U=input("Enter your mail id:")**

**M=int(input("Enter your marks:"))**

**D=con.connect(host="localhost",user="root",password="admin",database="STUDENT\_MANAGEMENT\_SYSTEM")**

**C=d.cursor()**

**Sq="insert into STUDENT\_DETAILS  
values( {}, {}, {}, {}, {}, {}, {}, {}, {})".format(a,n,r,i,p,t,u,  
w,m)**

**c.execute(sq)**

**d.commit()**

**print("student data added successfully")**

**#To delete items from student details table**

**Def delete\_student():**

**Print("\*\*\*\*\* DELETE STUDENT  
DETAILS\*\*\*\*\*")**

**K=input("Enter name to DELETE student details : ")**

```
D=con.connect(host="localhost",user="root",password="admin",database="STUDENT_MANAGEMENT_SYSTEM")
```

```
C=d.cursor()
```

```
c.execute("delete from STUDENT_DETAILS where name='{}'.format(k))
```

```
print(".....Data deleted successfully.....")
```

```
d.commit()
```

```
def admin():
```

```
while True:
```

```
    print("*****WELCOME TO STUDENT MANAGEMENT *****")
```

```
    print("-----  
----")
```

```
        print("1.search student details")
```

```
        print("2.update student details")
```

```
        print("3.view student details")
```

```
print("4.add student details")
print("5.delete student details")
print("6.exit")
ch=int(input("your choice: "))
if ch==1:
    search_student()
elif ch==2:
    update_details()
elif ch==3:
    view_Marks()
elif ch==4:
    add_student()
elif ch==5:
    delete_student()
elif ch==6:
    break
else :
    print("invalid input")
```



```
def user():  
    while True:  
        print("*****WELCOME TO STUDENT  
MANAGEMENT *****")  
        print("1.search your Details")  
        print("2.view Details")  
        print("3.exit")  
        ch=int(input("what is your choice: "))  
        if ch==1:  
            search()  
        elif ch==2:  
            view_Marks()  
        elif ch==3:  
            break  
        else:  
            print("invalid input")
```

#To search students from student details  
table.

**Def search():**

```
Print("*****STUDENT  
DETAILS*****")
```

```
K=input("Enter name to search student details: ")
```

```
D=con.connect(host="localhost",user="root",password="admin",database="STUDENT_MANAGEMENT_SYSTEM")
```

```
C=d.cursor()
```

```
c.execute("select*from STUDENT_DETAILS where  
NAME like '%{}%'".format(k))
```

```
a=c.fetchall()
```

```
if len(a)>=1:
```

```
for i in a:
```

```
print("Addmission no:",i[0])
```

```
print("Name is:",i[1])
```

```
print("Sex:",i[2])
```

```
print("Class=",i[3])
```

```
print("Sec:",i[4])
```

```
print("Phone number is=",i[5])
```

```
print("Email_id:",i[6])
```

```
print("Stream is:",i[7])
print("-----")
else:
    print("Student Details Not Found")
d.commit()
```

**#To display the marks of students from student details table .**

**Def view\_Marks():**

```
Print("***** VIEW STUDENT
DETAILS*****")
```

```
K=input("Enter name to view student details : ")
```

```
D=con.connect(host="localhost",user="root",password="admin",database="student_management_system")
```

```
C=d.cursor()
```

```
c.execute("select*from STUDENT_DETAILS where
NAME like '%{}%'".format(k))
```

```
s=c.fetchall()
```

```
for i in s:
```

```
print("MARKS:",i[8])
```

```
print("-----")
```

```
d.commit()
```

```
while True:
```

```
    print("*****WELCOME TO STUDENT  
MANAGEMENT*****")
```

```
    print("1.admin")
```

```
    print("2.user")
```

```
    print("3.exit")
```

```
    ch=int(input("login through: "))
```

```
    if ch==1:
```

```
        admin()
```

```
    elif ch==2:
```

```
        user()
```

```
    elif ch==3:
```

```
        break
```

```
    else:
```

```
        print("invalid input")
```

# MySQL DATABASES

# STUDENT MANAGEMENT SYSTEM DATABASE AND STUDENT DETAILS TABLE:-

```

Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 15
Server version: 8.0.34 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use STUDENT_MANAGEMENT_SYSTEM;
Database changed
mysql> SELECT*FROM STUDENT_DETAILS;

```

ADMISSION_NUMBER	NAME	SEX	CLASS	SEC	PHONE_NUMBER	EMAIL_ID	STREAM	MARKS
10001	SAMEER BEHERA	MALE	12	A	546788776	sam123@gmail.com	SCIENCE	567
10002	HARDIK BHISITH	FEMALE	11	B	456565675	sheetal69@gmail.com	ARTS	543
10003	PRIYANKA MAHAPATRA	FEMALE	12	A	898859855	priya@gmail.com	SCIENCE	585
10004	PUJA BEHERA	FEMALE	12	A	354365765	sweta@gmail.com	SCIENCE	567
10005	ASHISH KUMAR SATAPATHY	MALE	12	A	354444445	ashis@gmail.com	SCIENCE	500
10006	ANSHUMAN KUMAR	MALE	12	A	896875758	ans@gmail.com	SCIENCE	585
10007	DIVYA RANI PARIDA	FEMALE	12	A	987323212	divya56@gmail.com	SCIENCE	576
10008	ANIKET MISHRA	MALE	12	B	911222333	anik32@gmail.com	SCIENCE	598
10009	ANISHA BEHERA	FEMALE	12	A	465387257	anisha68@gmail.com	ARTS	576
10010	ANKITA BEHERA	FEMALE	12	A	898877675	mahi789@gmail.com	SCIENCE	599
10011	PRIYA ROY	FEMALE	11	C	492857396	priya76@gmail.com	ARTS	543
10013	MANISHA RANI	FEMALE	12	A	432888666	manisha@gmail.com	SCIENCE	555
10014	ANUPAMA	FEMALE	11	B	263547587	anu846@gmail.com	ARTS	522
10015	SALONI	FEMALE	12	A	162534786	saloni66@gmail.com	SCIENCE	544

## ADD STUDENT FROM STUDENT \_DETAILS TABLE:

*#Added a new student in student details table named PRITI SHARMA*

```

mysql> SELECT*FROM STUDENT_DETAILS;

```

ADMISSION_NUMBER	NAME	SEX	CLAS
10001	SAMEER BEHERA	MALE	
10002	HARDIK BHISITH	FEMALE	
10003	PRIYANKA MAHAPATRA	FEMALE	
10004	PUJA BEHERA	FEMALE	
10005	ASHISH KUMAR SATAPATHY	MALE	
10006	ANSHUMAN KUMAR	MALE	
10007	DIVYA RANI PARIDA	FEMALE	
10008	ANIKET MISHRA	MALE	
10009	ANISHA BEHERA	FEMALE	
10010	ANKITA BEHERA	FEMALE	
10011	SHREYA SAHU	FEMALE	
10013	MANISHA RANI	FEMALE	
10014	ANUPAMA	FEMALE	
10015	SALONI	FEMALE	
10016	M. KOMAL	FEMALE	
10017	ARADHIYA ROY	FEMALE	

16 rows in set (0.00 sec)

```

Client: Ubuntu
10004 PUJA BEHERA FEMALE
10005 ASHISH KUMAR SATAPATHY MALE
10006 ANSHUMAN KUMAR MALE
10007 DIVYA RANI PARIDA FEMALE
10008 ANIKET MISHRA MALE
10009 ANISHA BEHERA FEMALE
10010 ANKITA BEHERA FEMALE
10011 SHREYA SAHU FEMALE
10013 MANISHA RANI FEMALE
10014 ANUPAMA FEMALE
10015 SALONI FEMALE
10016 M. KOMAL FEMALE
10017 ARADHIYA ROY FEMALE
54354 PRITI SHARMA FEMALE

```

## Updated Student Class From STUDENT\_DETAILS Table:

PRIYANKA MAHAPATRA	FEMALE	12	A
PUJA BEHERA	FEMALE	12	A
ASHISH KUMAR SATAPATHY	MALE	12	A
ANSHUMAN KUMAR	MALE	12	A
DIVYA RANI PARIDA	FEMALE	12	A
ANIKET MISHRA	MALE	12	B
ANISHA BEHERA	FEMALE	12	A
ANKITA BEHERA	FEMALE	12	A
PRIYA ROY	FEMALE	11	C
MANISHA RANI	FEMALE	12	A
ANUPAMA	FEMALE	11	B
SALONI	FEMALE	12	A
M.KOMAL	FEMALE	12	C

```
mysql> SELECT * FROM STUDENT_DETAILS;
```

ADMISSION_NUMBER	NAME	SEX	CLASS	SEC
10001	SAMEER BEHERA	MALE	12	A
10002	HARDIK BHISITH	FEMALE	11	B
10003	PRIYANKA MAHAPATRA	FEMALE	12	A
10004	PUJA BEHERA	FEMALE	12	A
10005	ASHISH KUMAR SATAPATHY	MALE	12	A
10006	ANSHUMAN KUMAR	MALE	12	A
10007	DIVYA RANI PARIDA	FEMALE	12	A
10008	ANIKET MISHRA	MALE	12	B
10009	ANISHA BEHERA	FEMALE	12	A
10010	ANKITA BEHERA	FEMALE	12	A
10011	SHREYA SAHU	FEMALE	11	C
10013	MANISHA RANI	FEMALE	12	A
10014	ANUPAMA	FEMALE	11	B
10015	SALONI	FEMALE	11	A
10016	M.KOMAL	FEMALE	12	C
10017	ARADHIYA ROY	FEMALE	12	A

```
16 rows in set (0.00 sec)

mysql>
```

*#updated the class of student name SALONI from class 12 to class 11.*

## Updated Student Stream From STUDENT\_DETAILS Table:

ADMISSION_NUMBER	NAME	SEX	CLASS	SEC	PHONE_NUMBER	EMAIL_ID	STREAM	MARKS
10003	PRIYANKA MAHAPATRA	FEMALE	12	A	898859855	priya@gmail.com	SCIENCE	585
10004	PUJA BEHERA	FEMALE	12	A	354365765	sweta@gmail.com	SCIENCE	567
10005	ASHISH KUMAR SATAPATHY	MALE	12	A	354444445	ashis@gmail.com	SCIENCE	500
10006	ANSHUMAN KUMAR	MALE	12	A	896875758	ans@gmail.com	SCIENCE	585
10007	DIVYA RANI PARIDA	FEMALE	12	A	987323212	divya56@gmail.com	SCIENCE	576
10008	ANIKET MISHRA	MALE	12	B	911222333	ani32@gmail.com	SCIENCE	598
10009	ANISHA BEHERA	FEMALE	12	A	465387257	anisha68@gmail.com	ARTS	576
10010	ANKITA BEHERA	FEMALE	12	A	898877675	mahi789@gmail.com	SCIENCE	599
10011	PRIYA ROY	FEMALE	11	C	492857396	priya76@gmail.com	ARTS	543
10013	MANISHA RANI	FEMALE	12	A	432888666	manisha@gmail.com	SCIENCE	555
10014	ANUPAMA	FEMALE	11	B	263547587	anu846@gmail.com	ARTS	522
10015	SALONI	FEMALE	12	A	162534786	saloni66@gmail.com	SCIENCE	544
10016	M.KOMAL	FEMALE	12	C	334433445	koma114@gmail.com	ARTS	566

ADMISSION_NUMBER	NAME	SEX	CLASS	SEC	PHONE_NUMBER	EMAIL_ID	STREAM	MARKS
10001	SAMEER BEHERA	MALE	12	A	546788776	sam123@gmail.com	SCIENCE	567
10002	HARDIK BHISITH	FEMALE	11	B	456565675	sheeta169@gmail.com	ARTS	543
10003	PRIYANKA MAHAPATRA	FEMALE	12	A	898859855	priya@gmail.com	SCIENCE	585
10004	PUJA BEHERA	FEMALE	12	A	354365765	sweta@gmail.com	SCIENCE	567
10005	ASHISH KUMAR SATAPATHY	MALE	12	A	354444445	ashis@gmail.com	SCIENCE	500
10006	ANSHUMAN KUMAR	MALE	12	A	896875758	ans@gmail.com	SCIENCE	585
10007	DIVYA RANI PARIDA	FEMALE	12	A	987323212	divya56@gmail.com	SCIENCE	576
10008	ANIKET MISHRA	MALE	12	B	911222333	ani32@gmail.com	SCIENCE	598
10009	ANISHA BEHERA	FEMALE	12	A	465387257	anisha68@gmail.com	ARTS	576
10010	ANKITA BEHERA	FEMALE	12	A	898877675	mahi789@gmail.com	SCIENCE	599
10011	SHREYA SAHU	FEMALE	11	C	492857396	priya76@gmail.com	ARTS	543
10013	MANISHA RANI	FEMALE	12	A	432888666	manisha@gmail.com	SCIENCE	555
10014	ANUPAMA	FEMALE	11	B	263547587	anu846@gmail.com	ARTS	522
10015	SALONI	FEMALE	11	A	456987123	khushi2@gmail.com	ARTS	544
10016	M.KOMAL	FEMALE	12	C	334433445	koma114@gmail.com	ARTS	566
10017	ARADHIYA ROY	FEMALE	12	A	987876345	aryu@gmail.com	SCIENCE	567

*#updated the Stream of student name SALONI from SCIENCE to ARTS.*

## DELETE STUDENT FROM STUDENT \_DETAILS TABLE:

ADMISSION_NUMBER	NAME	SEX	CLASS	SEC	PHONE_NUMBER	EMAIL_ID	STREAM	MARKS
10004	PUJA BEHERA	FEMALE	12	A	354365765	sweta@gmail.com	SCIENCE	567
10005	ASHISH KUMAR SATAPATHY	MALE	12	A	354444445	ashis@gmail.com	SCIENCE	500
10006	ANSHUMAN KUMAR	MALE	12	A	896875758	ans@gmail.com	SCIENCE	585
10007	DIVYA RANI PARIDA	FEMALE	12	A	987323212	divya56@gmail.com	SCIENCE	576
10008	ANIKET MISHRA	MALE	12	B	911222333	ani32@gmail.com	SCIENCE	598
10009	ANISHA BEHERA	FEMALE	12	A	465387257	anisha68@gmail.com	ARTS	576
10010	ANKITA BEHERA	FEMALE	12	A	898877675	mahi789@gmail.com	SCIENCE	599
10011	SHREYA SAHU	FEMALE	11	C	492857396	priya76@gmail.com	ARTS	543
10013	MANISHA RANI	FEMALE	12	A	432888666	manisha@gmail.com	SCIENCE	555
10014	ANUPAMA	FEMALE	11	B	263547587	anu846@gmail.com	ARTS	522
10015	SALONI	FEMALE	11	A	456987123	khusi2@gmail.com	ARTS	544
10016	M. KOMAL	FEMALE	12	C	334433445	koma14@gmail.com	ARTS	566
10017	ARADHIYA ROY	FEMALE	12	A	987876345	aryu@gmail.com	SCIENCE	567
54354	PRITI SHARMA	FEMALE	11	A	674576587	aFh223@GMAIL.COM	SCIENCE	567

ADMISSION_NUMBER	NAME	SEX	CLASS	SEC	PHONE_NUMBER	EMAIL_ID	STREAM	MARKS
10001	SAMEER BEHERA	MALE	12	A	546788776	sam123@gmail.com	SCIENCE	567
10002	HARDIK BHISITH	FEMALE	11	B	456565675	sheeta169@gmail.com	ARTS	543
10003	PRIYANKA MAHAPATRA	FEMALE	12	A	898859855	priya@gmail.com	SCIENCE	585
10004	PUJA BEHERA	FEMALE	12	A	354365765	sweta@gmail.com	SCIENCE	567
10005	ASHISH KUMAR SATAPATHY	MALE	12	A	354444445	ashis@gmail.com	SCIENCE	500
10006	ANSHUMAN KUMAR	MALE	12	A	896875758	ans@gmail.com	SCIENCE	585
10007	DIVYA RANI PARIDA	FEMALE	12	A	987323212	divya56@gmail.com	SCIENCE	576
10008	ANIKET MISHRA	MALE	12	B	911222333	ani32@gmail.com	SCIENCE	598
10009	ANISHA BEHERA	FEMALE	12	A	465387257	anisha68@gmail.com	ARTS	576
10010	ANKITA BEHERA	FEMALE	12	A	898877675	mahi789@gmail.com	SCIENCE	599
10011	SHREYA SAHU	FEMALE	11	C	492857396	priya76@gmail.com	ARTS	543
10013	MANISHA RANI	FEMALE	12	A	432888666	manisha@gmail.com	SCIENCE	555
10014	ANUPAMA	FEMALE	11	B	263547587	anu846@gmail.com	ARTS	522
10015	SALONI	FEMALE	11	A	456987123	khusi2@gmail.com	ARTS	544
10016	M. KOMAL	FEMALE	12	C	334433445	koma14@gmail.com	ARTS	566
10017	ARADHIYA ROY	FEMALE	12	A	987876345	aryu@gmail.com	SCIENCE	567

***#Deleted the Details of student name PRITI SHARMA from STUDENT\_ DETAILS TABLE.***



**OUTPUTS**

# #User Panel and Admin Panel:

```
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct 2 2023, 13:03:39) [MSC v.1935 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
RESTART: C:\Users\KV BERHAMPUR\OneDrive\STUDENT MANAGEMENT SYSTEM\STUDENT MANAGEMENT SYSTEM.py
-----
*****WELCOME TO STUDENT MANAGEMENT*****
1.admin
2.user
3.exit
login through: 1
*****WELCOME TO STUDENT MANAGEMENT *****
```

# #search Student Details:

```
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct 2 2023, 13:03:39) [MSC v.1935 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
RESTART: C:\Users\KV BERHAMPUR\OneDrive\STUDENT MANAGEMENT SYSTEM\STUDENT MANAGEMENT SYSTEM.py
-----
*****WELCOME TO STUDENT MANAGEMENT*****
1.admin
2.user
3.exit
login through: 1
*****WELCOME TO STUDENT MANAGEMENT *****
-----
1.search student details
2.update student details
3.view student details
4.add student details
5.delete student details
6.exit
your choice: 1
*****STUDENT DETAILS*****
Enter name to search student Name : ANKITA
Admission no: 10010
Name is: ANKITA BEHERA
Sex: FEMALE
Class 12
Sec: A
Phone number is 898877675
Email id: mahi789@gmail.com
Stream is: SCIENCE
-----
2023.11.29
```

# #Update student Details:

## ##Name:

```
*****WELCOME TO STUDENT MANAGEMENT *****
-----
1.search student details
2.update student details
3.view student details
4.add student details
5.delete student details
6.exit
your choice: 2
1. Update Name
2. Update Gender
3. Update class
4. Update Section
5. Update Phone no.
6. Update E-mail id
7. Update Stream
Enter your choice to update1
-----
You are inside updating name.
Enter ADMISSION_NUMBER whose data you want to update: 10011
Enter your updated name: SHREYA SAHU
```

## ##Gender:

```
"IDLE Shell 3.12.0"
File Edit Shell Debug Options Window Help
*****WELCOME TO STUDENT MANAGEMENT *****
-----
1.search student details
2.update student details
3.view student details
4.add student details
5.delete student details
6.exit
your choice: 2
1. Update Name
2. Update Gender
3. Update class
4. Update Section
5. Update Phone no.
6. Update E-mail id
7. Update Stream
Enter your choice to update2
-----
You are inside updating Gender.
Enter name whose data you want to update: SAMEER BEHERA
Enter your updated Gender: MALE
```

## ##Class:

```

| "IDLE Shell 3.12.0"
| File Edit Shell Debug Options Window Help
|-----|
| *****WELCOME TO STUDENT MANAGEMENT *****
|-----|
| 1.search student details
| 2.update student details
| 3.view student details
| 4.add student details
| 5.delete student details
| 6.exit
| your choice: 2
| 1. Update Name
| 2. Update Gender
| 3. Update class
| 4. Update Section
| 5. Update Phone no.
| 6. Update E-mail id
| 7. Update Stream
| Enter your choice to update3
|-----|
| You are inside updating class.
| Enter name whose data you want to update: SALONI
| Enter your updated class: 11

```

## ##Section:

```

| *****WELCOME TO STUDENT MANAGEMENT *****
|-----|
| 1.search student details
| 2.update student details
| 3.view student details
| 4.add student details
| 5.delete student details
| 6.exit
| your choice: 2
| 1. Update Name
| 2. Update Gender
| 3. Update class
| 4. Update Section
| 5. Update Phone no.
| 6. Update E-mail id
| 7. Update Stream
| Enter your choice to update4
|-----|
| You are inside updating section.
| Enter name whose data you want to update: SALONI
| Enter your updated section: A

```

## ##Stream:

```
*****WELCOME TO STUDENT MANAGEMENT *****
-----
1.search student details
2.update student details
3.view student details
4.add student details
5.delete student details
6.exit
your choice: 2
1. Update Name
2. Update Gender
3. Update class
4. Update Section
5. Update Phone no.
6. Update E-mail id
7. Update Stream
Enter your choice to update7
-----
You are inside updating stream.
Enter name whose data you want to update: SALONI
Enter your updated stream : ARTS
```

## ##View Mark of student :

```
*****WELCOME TO STUDENT MANAGEMENT *****
-----
1.search student details
2.update student details
3.view student details
4.add student details
5.delete student details
6.exit
your choice: 3
***** VIEW STUDENT DETAILS*****
Enter name to view student details : SALONI
MARKS: 544
-----
```



## #Add student Details:

```
"IDE Shd3.120"
File Edit Shell Debug Options Window Help
-----
*****WELCOME TO STUDENT MANAGEMENT *****
-----
1.search student details
2.update student details
3.view student details
4.add student details
5.delete student details
6.exit
your choice: 4
..... ADD STUDENT DETAILS.....
Enter student name : PRITI SHARMA
Enter student admission number: 54354
Enter gender of student: FEMALE
Enter class of student:11
Enter student section : A
Enter student phone number:674576587
Enter student stream: SCIENCE
Enter your mail id:afh223@GMAIL.COM
Enter your marks:567
student data added successfully
*****
```

## #Delete Student Details:

```
*****WELCOME TO STUDENT MANAGEMENT *****
-----
1.search student details
2.update student details
3.view student details
4.add student details
5.delete student details
6.exit
your choice: 5
***** DELETE STUDENT DETAILS*****
Enter name to DELETE student details : PRITI SHARMA
.....Data deleted successfully.....
```

## **REFERENCES**

1. News API  
<https://newsapi.org/>
2. Wikipedia  
<https://www.wikipedia.org/>
3. Python  
<https://www.python.org/>
4. MySQL  
<https://www.mysql.com/>
5. ANSI Escape Codes in Python
  - <https://pypi.org/project/ansi/>
  - <https://replit.com/talk/learn/ANSI-Escape-Codes-in-Python/22803>
  - <https://gist.github.com/rene-d/9e584a7dd2935d0f461904b9f2950007>
6. Class 11<sup>th</sup> & 12<sup>th</sup> Computer Science  
Arihant Books

**THANK YOU**