CENTRAL BOARD OF SECONDARY EDUCATION

MAYO INTERNATIONAL SCHOOL

IP Extension Delhi - 110092

Computer Science practical file

SUBMITTED BY: ALOK YADAV

HOD(COMPUTER):Mrs. Sapna rai

CLASS : 12 TH B

ROLL NO:

ACKNOWLEDGEMENT

I would like to thank my **COMPUTER SCIENCE** *teacher who gave me a golden opportunity to work on this project. I'd also like to express my gratitude to my school principal* Mrs. Hembala mam *wholeheartedly.*

I must also thank my parents and friends for the immense support and help during this project. Without their help, completing this project would have been very difficult.

CERTIFICATE

This is to certify that this project file is a bona- fide work done by **ALOK Yadav** of class **XII-B** in session **2021-22** in partial fulfillment of CBSE′ s AISSCE Examination 2021 and has been carried out under my direct supervision and guidance. This report or a similar report on the topic **has not been submitted for any other examination and does not form a part of any other course undergone by the candidate**.

Signature of Principal Name : **Mrs. Hem Bala**	Signature of Teacher Name : **Sapna Ma'am**

CONTENTS

Part A Data Structure

1. **Write a menu-driven python program to implement stack operation.**
   Code:

```
def check_stack_isEmpty(stk):
    if stk==[]:
        return True
    else:
        return False
# An empty list to store stack elements, initially empty
s=[]
top = None # This is top pointer for push and pop
def main_menu():
    while True:
        print("Stack Implementation")
        print("1 - Push")
        print("2 - Pop")
        print("3 - Peek")
        print("4 - Display")
        print("5 - Exit")
        ch = int(input("Enter the your choice:"))
        if ch==1:
            el = int(input("Enter the value to push an element:"))
            push(s,el)
        elif ch==2:
            e=pop_stack(s)
            if e=="UnderFlow":
                print("Stack is underflow!")
            else:
                print("Element popped:",e)
        elif ch==3:
            e=pop_stack(s)
            if e=="UnderFlow":
                print("Stack is underflow!")
            else:
                print("The element on top is:",e)
        elif ch==4:
            display(s)
        elif ch==5:
            break
        else:
            print("Sorry, You have entered invalid option")
def push(stk,e):
    stk.append(e)
    top = len(stk)-1

def display(stk):
    if check_stack_isEmpty(stk):
        print("Stack is Empty")
    else:
        top = len(stk)-1
        print(stk[top],"-Top")
        for i in range(top-1,-1,-1):
            print(stk[i])

def pop_stack(stk):
    if check_stack_isEmpty(stk):
        return "UnderFlow"
    else:
        e = stk.pop()
        if len(stk)==0:
            top = None
        else:
            top = len(stk)-1
        return e
def peek(stk):
    if check_stack_isEmpty(stk):
        return "UnderFlow"
    else:
        top = len(stk)-1
        return stk[top]
```

Output:

```
>>> main_menu()
Stack Implementation
1 - Push
2 - Pop
3 - Peek
4 - Display
5 - Exit
Enter the your choice:1
Enter the value to push an element:24
Stack Implementation
1 - Push
2 - Pop
3 - Peek
4 - Display
5 - Exit
Enter the your choice:1
Enter the value to push an element:56
Stack Implementation
1 - Push
2 - Pop
3 - Peek
4 - Display
5 - Exit
Enter the your choice:166
Sorry, You have entered invalid option
Stack Implementation
1 - Push
2 - Pop
3 - Peek
4 - Display
5 - Exit
Enter the your choice:4
56 -Top
24
Stack Implementation
1 - Push
2 - Pop
3 - Peek
4 - Display
5 - Exit
Enter the your choice:2
Element popped: 56
Stack Implementation
1 - Push
2 - Pop
3 - Peek
4 - Display
5 - Exit
Enter the your choice:4
24 -Top
Stack Implementation
1 - Push
2 - Pop
3 - Peek
4 - Display
5 - Exit
Enter the your choice:5
```

2. Write a program to implement a stack for the employee details (empno, name).
Code:

```
stk=[]
top=-1
def line():
    print('~'*100)
def isEmpty():
    global stk

    if stk==[]:
        print("Stack is empty!!!")
    else:
        None

def push():
    global stk
    global top
    empno=int(input("Enter the employee number to push:"))
    ename=input("Enter the employee name to push:")
    stk.append([empno,ename])
    top=len(stk)-1

def display():
    global stk
    global top
    if top==-1:
        isEmpty()
    else:
        top=len(stk)-1
        print(stk[top],"<-top")
        for i in range(top-1,-1,-1):
            print(stk[i])

def pop_ele():
    global stk
    global top
    if top==-1:
        isEmpty()
    else:
        stk.pop()
        top=top-1


def main():
    while True:
        line()
        print("1. Push")
        print("2. Pop")
        print("3. Display")
        print("4. Exit")
        ch=int(input("Enter your choice:"))
        if ch==1:
            push()
            print("Element Pushed")
        elif ch==2:
            pop_ele()
        elif ch==3:
            display()
        elif ch==4:
            break
        else:
            print("Invalid Choice")
```

Output:

```
>>> main()
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
1. Push
2. Pop
3. Display
4. Exit
Enter your choice:1
Enter the employee number to push:23
Enter the employee name to push:ALOK
Element Pushed
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
1. Push
2. Pop
3. Display
4. Exit
Enter your choice:1
Enter the employee number to push:56
Enter the employee name to push:OM
Element Pushed
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
1. Push
2. Pop
3. Display
4. Exit
Enter your choice:3
[56, 'OM'] <-top
[23, 'ALOK']
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
1. Push
2. Pop
3. Display
4. Exit
Enter your choice:2
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
1. Push
2. Pop
3. Display
4. Exit
Enter your choice:4
```

3. **Write a python program to check whether a string is a palindrome or not using stack.**

Code:

```
stack = []
top = -1
# push function
def push(ele):
    global top
    top += 1
    stack[top] = ele

# pop function
def pop():
    global top
    ele = stack[top]
    top -= 1
    return ele

def isPalindrome(string):
    global stack
    length = len(string)

    stack = ['0'] * (length + 1)

    mid = length // 2
    i = 0
    while i < mid:
        push(string[i])
        i += 1

    if length % 2 != 0:
        i += 1

    while i < length:
        ele = pop()

        if ele != string[i]:
            return False
        i += 1
    return True
string = input("Enter string to check:")
if isPalindrome(string):
    print("Yes, the string is a palindrome")
else:
    print("No, the string is not a palindrome")
```

Output:

```
Enter string to check:olalo
Yes, the string is a palindrome

Enter string to check:alok
No, the string is not a palindrome
```

MYsql queries

1. Consider the following MOVIE table and write the SQL queries based on it.

Movie_ID	MovieName	Type	Releaeyear	IMDB RATING
M001	Spider man :nhw	Action	2021	8.7
M002	Extraction	Action	2020	6.7
M003	Looop Lapeta	Thriller	2022	5.1
M004	Jai bhim	Drama	2021	9.3
M005	Sardar udham	Biography	2021	8.7
M006	Gehraiyaan	Romance	2022	6.4

a) Display all information from movie.

b) Display the type of movies.

c) Display movieid, moviename, IMDB- RATING by released year.

d) Display movieid, moviename and type of all movies with imdb rating of 6.0 or above

e) Display the movie of type action and romance.

f) Display the list of movies which are released in 2021

Answers:

A)

```
mysql> select * from movie ;
+----------+------------------+-----------+--------------+-------------+
| movie_id | moviename        | type      | release_year | imdb_rating |
+----------+------------------+-----------+--------------+-------------+
| M001     | Spider man :NHW  | Action    |         2021 |         8.7 |
| M002     | Extraction       | Action    |         2020 |         6.7 |
| M003     | Looop lapeta     | Thriller  |         2022 |         5.1 |
| M004     | Jai Bhim         | Drama     |         2021 |         9.3 |
| M005     | Sardar Udham     | Biography |         2021 |         8.7 |
| M006     | Gehraiyaan       | Romance   |         2022 |         6.4 |
+----------+------------------+-----------+--------------+-------------+
6 rows in set (0.12 sec)
```

B)

```
mysql> select distinct type from movie
    -> ;
+-------------+
| type        |
+-------------+
| Action      |
| Thriller    |
| Drama       |
| Biography   |
| Romance     |
+-------------+
5 rows in set (0.21 sec)
```

C)

```
mysql> select movie_id,moviename,imdb_rating from movie;
+----------+--------------------+--------------+
| movie_id | moviename          | imdb_rating  |
+----------+--------------------+--------------+
| M001     | Spider man :NHW    |          8.7 |
| M002     | Extraction         |          6.7 |
| M003     | Looop lapeta       |          5.1 |
| M004     | Jai Bhim           |          9.3 |
| M005     | Sardar Udham       |          8.7 |
| M006     | Gehraiyaan         |          6.4 |
+----------+--------------------+--------------+
6 rows in set (0.00 sec)
```

D)

```
mysql> select movie_id ,moviename,type from movie where imdb_rating>=6.0;
+----------+------------------+-----------+
| movie_id | moviename        | type      |
+----------+------------------+-----------+
| M001     | Spider man :NHW  | Action    |
| M002     | Extraction       | Action    |
| M004     | Jai Bhim         | Drama     |
| M005     | Sardar Udham     | Biography |
| M006     | Gehraiyaan       | Romance   |
+----------+------------------+-----------+
5 rows in set (0.23 sec)
```

E)

```
mysql> select moviename from movie where type="Action" or type ="Romance";
+------------------+
| moviename        |
+------------------+
| Spider man :NHW  |
| Extraction       |
| Gehraiyaan       |
+------------------+
3 rows in set (0.00 sec)
```

F)

```
mysql> select moviename from movie where release_year=2021;
+------------------+
| moviename        |
+------------------+
| Spider man :NHW  |
| Jai Bhim         |
| Sardar Udham     |
+------------------+
3 rows in set (0.00 sec)
```

2. Write following queries:

a) Write a query to display the square root of 4

b) Write a query to display the number 42423.342344234 rounding off to the next six decimal places

c) Write a query to display "r s" from the word "Computer science".

d) Write a query to display today's date into DD.MM.YYYY format.

e) Write a query to display 'DIA' from the word "MEDIA".

f) Write a query to display moviename - type from the table movie.

Answers:

a)

```
mysql> select pow(4,2);
+----------+
| pow(4,2) |
+----------+
|       16 |
+----------+
1 row in set (0.12 sec)
```

b)

```
mysql> select round(42423.342344234,6);
+--------------------------+
| round(42423.342344234,6) |
+--------------------------+
|             42423.342344 |
+--------------------------+
1 row in set (0.10 sec)
```

c).

```
mysql> select mid("Computer",4,3);
+-----------------------+
| mid("Computer",4,3) |
+-----------------------+
| put                   |
+-----------------------+
1 row in set (0.00 sec)
```

d).

```
mysql> select concat(day(now()),concat('.',month(now()),concat('.',year(now())))) "Date";
+----------+
| Date     |
+----------+
| 9.1.2022 |
+----------+
1 row in set (0.00 sec)
```

e).

```
mysql> select right("Media",3);
+---------------------+
| right("Media",3) |
+---------------------+
| dia                 |
+---------------------+
1 row in set (0.00 sec)
```

f).

```
mysql> select concat(moviename,concat(' - ',type)) from movie;
+---------------------------------------+
| concat(moviename,concat(' - ',type)) |
+---------------------------------------+
| The Kashmir Files - Action            |
| Attack - Action                       |
| Loop Lapeta - Thriller                |
| Bdhai Do - Drama                      |
| Shabaash Mothu - Biography            |
| Gehriyaan - Romance                   |
+---------------------------------------+
6 rows in set (0.00 sec)
```

3. Suppose your school management has decided to conduct cricket matches between students of Class XI and Class XII. Students of each class are asked to join any one of the four teams – Team AOT, Team JJK, Team HXH and Team DBZ. During summer vacations, various matches will be conducted between these teams. Help your sports teacher to do the following:

a) Create a database "game".

b) Create a table "TEAM" with following considerations:

   a. It should have a column TeamID for storing an integer value between 1 to 9, which refers to unique identification of a team.

   b. Each TeamID should have its associated name (TeamName), which should be a string of length not less than 10 characters.

   c. Using table level constraint, make TeamID as the primary key

c) Show the structure of the table TEAM using a SQL statement.

d) As per the preferences of the students four teams were formed as given below. Insert these four rows in TEAM table:

   a. Row 1: (1, AOT)
   b. Row 2: (2, JJK)
   c. Row 3: (3, HXH)
   d. Row 3: (4, DBZ)

e) Show the contents of the table TEAM using a DML statement.

f) Now create another table MATCH_DETAILS and insert data as shown below. Choose appropriate data types and constraints for each attribute.

Answers:

a) `create database sports;`

```
mysql> create database game;
Query OK, 1 row affected (0.50 sec)
```

b)

```
mysql> create table team (teamid int,team_name varchar(20),primary key (teamid));
Query OK, 0 rows affected (2.84 sec)
```

c)

```
mysql> desc team;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| teamid    | int         | NO   | PRI | NULL    |       |
| team_name | varchar(20) | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
2 rows in set (0.13 sec)
```

- **Inserting data:**

```
mqsql> insert into team -> values(1,'Tehlka');
```

```
mysql> insert into team values(1,"AOT");
Query OK, 1 row affected (0.27 sec)

mysql> insert into team values(2,"JJK");
Query OK, 1 row affected (0.14 sec)

mysql> insert into team values(3,"HXH");
Query OK, 1 row affected (0.06 sec)

mysql> insert into team values(4,"DBZ");
Query OK, 1 row affected (0.06 sec)
```

- **Show the content of table - team:**

```
select * from team;
```

```
mysql> select * from team;
+--------+-----------+
| teamid | team_name |
+--------+-----------+
|      1 | AOT       |
|      2 | JJK       |
|      3 | HXH       |
|      4 | DBZ       |
+--------+-----------+
4 rows in set (0.00 sec)
```

0

- **Creating another table:**

```
mysql> create table match_details
    -> (matchid varchar(3),
    -> matchdate date,
    -> fstteamid int(1) references team(teamid),
    -> scndteamid int(1) references team(teamid),
    -> fstteamscore int(3),
    -> scndteamscore int(3),primary key(matchid));
Query OK, 0 rows affected, 4 warnings (2.06 sec)
```

```
mysql> select * from match_details;
+---------+------------+----------+-----------+-------------+--------------+
| matchid | matchdate  | fstteamid | scndteamid | fstteamscore | scndteamscore |
+---------+------------+----------+-----------+-------------+--------------+
| 1M      | 2022-01-20 |        1 |         2 |         186 |          173 |
| 2M      | 2022-01-21 |        3 |         4 |         149 |          150 |
| 3M      | 2022-01-22 |        4 |         1 |         146 |          151 |
| 4M      | 2022-01-23 |        2 |         3 |         140 |          113 |
| 5M      | 2022-01-24 |        2 |         4 |         133 |          134 |
| 6M      | 2022-01-25 |        1 |         4 |         155 |          154 |
+---------+------------+----------+-----------+-------------+--------------+
6 rows in set (0.00 sec)
```

4 . w rite following queries:

a) Display all themacth details
b) Display matchid,team_name teamscore of frstteam >140.
c) Display matchid, team_names along their matchid and team id
d) Display unique team names
e) Display matchid and matchdate played by Aot and JJK .

Answers:

a) `SELECT * FEOM MATCH_DETAILS;`

```
mysql> select * from match_details;
+---------+------------+----------+-----------+-------------+--------------+
| matchid | matchdate  | fstteamid | scndteamid | fstteamscore | scndteamscore |
+---------+------------+----------+-----------+-------------+--------------+
| 1M      | 2022-01-20 |        1 |         2 |         186 |          173 |
| 2M      | 2022-01-21 |        3 |         4 |         149 |          150 |
| 3M      | 2022-01-22 |        4 |         1 |         146 |          151 |
| 4M      | 2022-01-23 |        2 |         3 |         140 |          113 |
| 5M      | 2022-01-24 |        2 |         4 |         133 |          134 |
| 6M      | 2022-01-25 |        1 |         4 |         155 |          154 |
+---------+------------+----------+-----------+-------------+--------------+
6 rows in set (0.00 sec)
```

b)

```
mysql> select match_details.matchid, match_details.fstteamid, team.team_name,match_details.fstteamscore from match_details, team where match_details.fstteamid = team.te
amid and match_details.fstteamscore>140;
+---------+-----------+-----------+-------------+
| matchid | fstteamid | team_name | fstteamscore |
+---------+-----------+-----------+-------------+
| 6M      |         1 | AOT       |         155 |
| 1M      |         1 | AOT       |         186 |
| 2M      |         3 | HXH       |         149 |
| 3M      |         4 | DBZ       |         146 |
+---------+-----------+-----------+-------------+
```

c) select matchid, team_name, fstteamid, S cndteamid, matchdate from match_details, team where match_details.fstteamid = team.teamid;

```
mysql> select matchid, team_name, fstteamid, scndteamid, matchdate from match_details, team where match_details.fstteamid = team.teamid;
+---------+-----------+-----------+------------+------------+
| matchid | team_name | fstteamid | scndteamid | matchdate  |
+---------+-----------+-----------+------------+------------+
| 1M      | AOT       |         1 |          2 | 2022-01-20 |
| 2M      | HXH       |         3 |          4 | 2022-01-21 |
| 3M      | DBZ       |         4 |          1 | 2022-01-22 |
| 4M      | JJK       |         2 |          3 | 2022-01-23 |
| 5M      | JJK       |         2 |          4 | 2022-01-24 |
| 6M      | AOT       |         1 |          4 | 2022-01-25 |
+---------+-----------+-----------+------------+------------+
6 rows in set (0.00 sec)
```

d) select distinct(team_name) from match_details, team where match_details.fstteamid = team.teamid;

```
mysql> select distinct(team_name) from match_details, team where match_details.fstteamid = team.teamid;
+-----------+
| team_name |
+-----------+
| AOT       |
| HXH       |
| DBZ       |
| JJK       |
+-----------+
4 rows in set (0.10 sec)
```

e) select matchid,matchdate from match_details, team where match_details.fstteamid = team.teamid and team.teamname in ('AOT','JJK')

```
mysql> select matchid,matchdate from match_details, team where match_details.fstteamid = team.teamid and team.team_name in ('AOT','JJK');
+---------+------------+
| matchid | matchdate  |
+---------+------------+
| 1M      | 2022-01-20 |
| 4M      | 2022-01-23 |
| 5M      | 2022-01-24 |
| 6M      | 2022-01-25 |
+---------+------------+
4 rows in set (0.00 sec)
```

5. Consider the following table and write the queries:

N0	Name	Age	Department	Charges	Dateofadm
S005	Sandeep	65	surgery	300	2022/04/22
S003	Ravina	24	OPD	200	2022/03/18
S002	Karan	45	OPD	200	2022/02/25
S006	Tarun	25	surgery	300	2022/06/11
S001	Zubin	36	ENT	300	2022/05/10
S004	Kateki	66	ENT	250	2022/01/12
S009	Ankita	29	ENT	800	2022/07/17

a) Display all the Doctors in the ascending order of age .

b) Display maximum charge of department for each doctor individually as per age from hospital.

c) Display all the doctor in descending orders of department.

d) Display average charges of department for each doctor individually as per NO from hospital which average charge is more than 160.

e) Display the sum of charge for each department.

Answers:

a) `select * from hospital order by age;`

```
mysql> select * from hospital;
+------+---------+------+------------+---------+------------+
| NO   | name    | age  | department | charges | dateofadm  |
+------+---------+------+------------+---------+------------+
| S005 | Sandeep |   65 | surgery    |     300 | 2022-04-22 |
| S003 | Ravina  |   24 | OPD        |     200 | 2022-03-18 |
| S002 | Karan   |   45 | OPD        |     200 | 2022-02-25 |
| S006 | Tarun   |   25 | surgery    |     300 | 2022-06-11 |
| S001 | Zubin   |   36 | ENT        |     300 | 2022-05-10 |
| S004 | kateki  |   66 | ENT        |     250 | 2022-01-12 |
| S009 | Ankita  |   29 | ENT        |      80 | 2022-07-17 |
```

b) select age,max(charges) from hospital group by age;

```
mysql> select age,max(charges) from hospital group by age;
+------+--------------+
| age  | max(charges) |
+------+--------------+
|   65 |          300 |
|   24 |          200 |
|   45 |          200 |
|   25 |          300 |
|   36 |          300 |
```

select * from hospital order by department desc;

```
mysql> select * from hospital order by department desc;
+------+---------+------+------------+---------+------------+
| NO   | name    | age  | department | charges | dateofadm  |
+------+---------+------+------------+---------+------------+
| S005 | Sandeep |   65 | surgery    |     300 | 2022-04-22 |
| S006 | Tarun   |   25 | surgery    |     300 | 2022-06-11 |
| S003 | Ravina  |   24 | OPD        |     200 | 2022-03-18 |
| S002 | Karan   |   45 | OPD        |     200 | 2022-02-25 |
| S001 | Zubin   |   36 | ENT        |     300 | 2022-05-10 |
| S004 | kateki  |   66 | ENT        |     250 | 2022-01-12 |
| S009 | Ankita  |   29 | ENT        |      80 | 2022-07-17 |
+------+---------+------+------------+---------+------------+
7 rows in set (0.00 sec)
```

d) select NO,avg(charges) from hospital group by NO having avg(charges)>160;

```
mysql> select NO,avg(charges) from hospital group by NO having avg(charges)>160;
+------+--------------+
| NO   | avg(charges) |
+------+--------------+
| S005 |     300.0000 |
| S003 |     200.0000 |
| S002 |     200.0000 |
| S006 |     300.0000 |
| S001 |     300.0000 |
| S004 |     250.0000 |
+------+--------------+
6 rows in set (0.09 sec)
```

E) select department,sum(charges) from hospital group by department;

```
mysql> select department,sum(charges) from hospital group by department;
+------------+--------------+
| department | sum(charges) |
+------------+--------------+
| surgery    |          600 |
| OPD        |          400 |
| ENT        |          630 |
+------------+--------------+
3 rows in set (0.03 sec)
```

Part C Python Database connectivity

Code:

```
import os
import platform

import mysql.connector

mydb=mysql.connector.connect(host="localhost",\
                             user="root",\
                             passwd="A@l*k",\
                             database="candy",charset="utf8")
print(mydb)
mycursor=mydb.cursor()

def removeStu():
    roll=int(input("Enter the roll number of the student to be deleted : "))
    rl=(roll,)
    sql="Delete from student where roll_number=%s"
    mycursor.execute(sql,rl)
    print('Record deleted!!!')
    mydb.commit()

def stuview():
    mycursor.execute("select * from student")
    myrus=mycursor.fetchall()
    for x in myrus:
        print(x)

def MenuSet(): #Function For The Student Management System
    print("Enter 1 : To Delete Student")
    print("Enter 2 : To View Students")
    userInput = int(input("Please Select An Above Option: ")) #Will Take Input From User
    if(userInput == 1):
        removeStu()
    if(userInput == 2):
        stuview()
MenuSet()
def runAgain():
    runAgn = input("\nwant To Run Again Y/n: ")
    while(runAgn.lower() == 'y'):
        if(platform.system() == "Windows"):
            print(os.system('cls'))


        else:
            print(os.system('clear'))
        MenuSet()
        runAgn = input("\nwant To Run Again y/n: ")
runAgain()
```

Output

```
Enter 1 : To Delete Student
Enter 2 : To View Students

Please Select An Above Option: 1

Enter the roll number of the student to be deleted : 16

Enter the class of the student to be deleted:11
Record deleted!!!


want To Run Again Y/n: Y
0
Enter 1 : To Delete Student
Enter 2 : To View Students

Please Select An Above Option: 2
(1, 'Aryan', 16, 12)
(2, 'Om tripathi', 18, 12)
(3, 'surya bhadoria', 15, 10)
(4, 'Simmi kaur', 16, 11)


want To Run Again y/n: n
```

Code

```
import mysql.connector as mycon
cn = mycon.connect(host='localhost',user='root',password="A@1*k",database="candy",charset="utf8")
cur = cn.cursor()
print('Welcome to student Details Updation screen... ')

print("******************EDIT STUDENT DETAILS *************************")
ro = int(input("Enter Student's roll number to edit :"))
query="select * from student where roll_number="+str(ro)
cur.execute(query)
results = cur.fetchall()
if cur.rowcount<=0:
    print("\## SORRY! NO MATCHING DETAILS AVAILABLE ##")
else:
    print("*************************************************")
    print('%5s'%"ROLL NO",'%15s'%'NAME','%12s'%'AGE','%10s'%'CLASS')
    print("*************************************************")
    for row in results:
        print('%5s' % row[0],'%15s'%row[1],'%12s'%row[2],'%10s'%row[3])
print("-"*50)
ans = input("Are you sure to update ? (y/n)")
if ans=="y" or ans=="Y":
    d = input("Enter new name to update (enter old value if not to update) :")
    s = int(input("Enter new age to update (enter old value if not to update) :"))

    query="update student set name='"+d+"',age="+str(s) + " where roll_number="+str(ro)

    cur.execute(query)
    cn.commit()
    print("\n## RECORD UPDATED  ##")
```

OUTPUT

```
Welcome to student Details Updation screen...
******************EDIT STUDENT DETAILS *************************

Enter Student's roll number to edit :01
*************************************************
ROLL NO            NAME          AGE      CLASS
*************************************************
    1      Alok yadav           17         12
-------------------------------------------------

Are you sure to update ? (y/n)y

Enter new name to update (enter old value if not to update) :Aryan

Enter new age to update (enter old value if not to update) :16

## RECORD UPDATED  ##

In [3]:
```