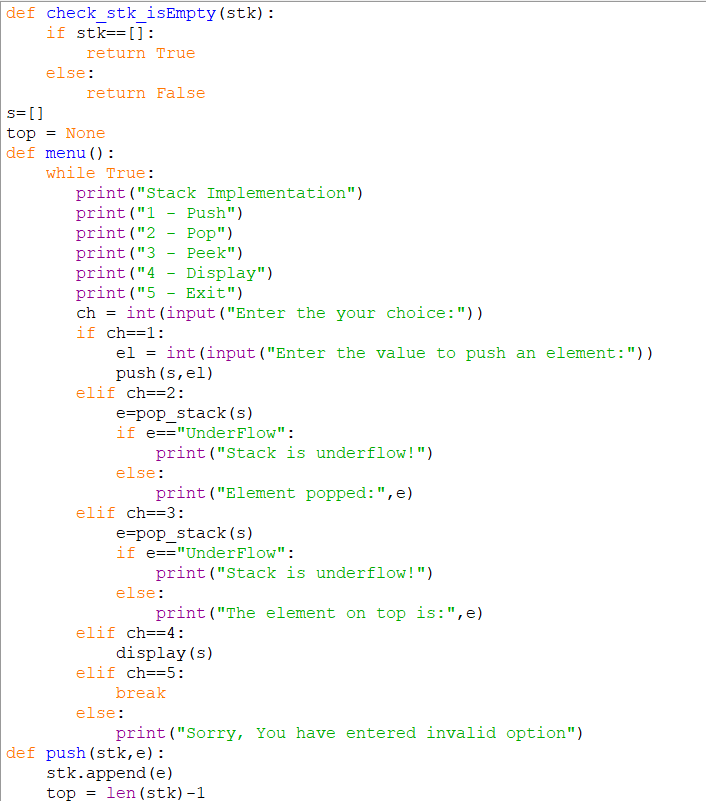
**Part A – Data Structure**

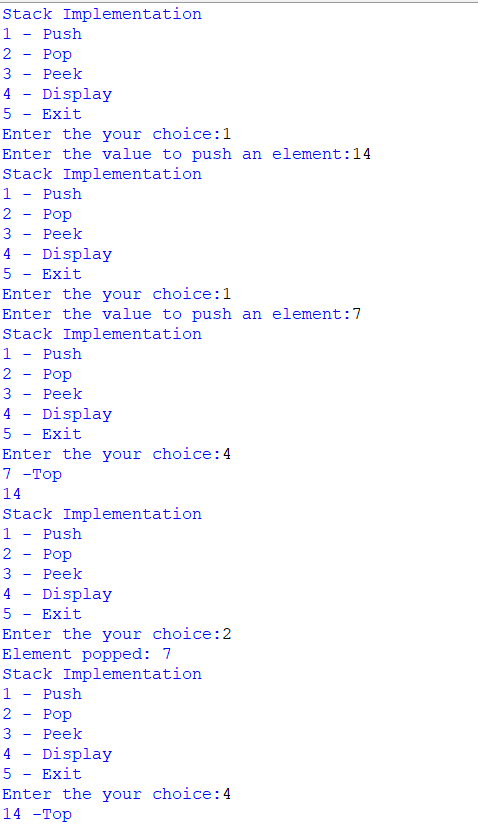
1. Write a menu-driven python program to implement stack operation.

Code:





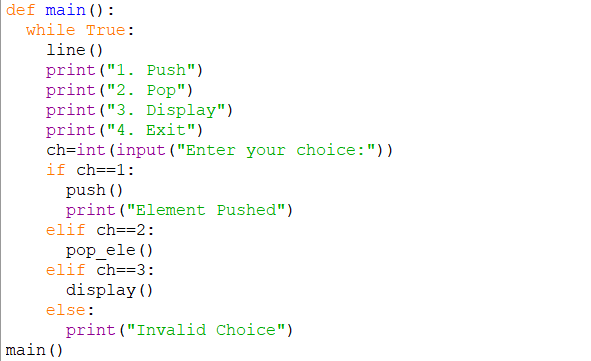
Output:



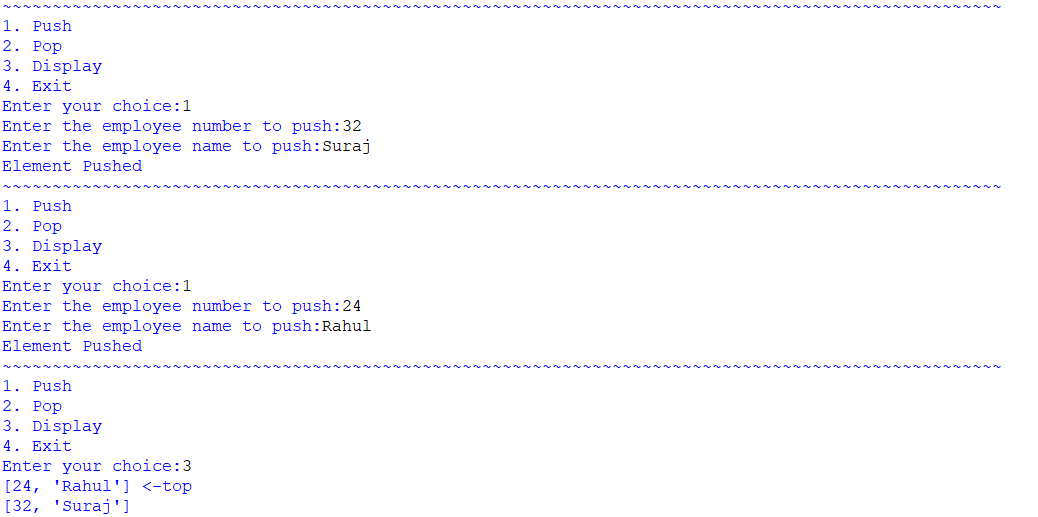
1. Write a program to implement a stack for the employee details (empno, name).

Code:



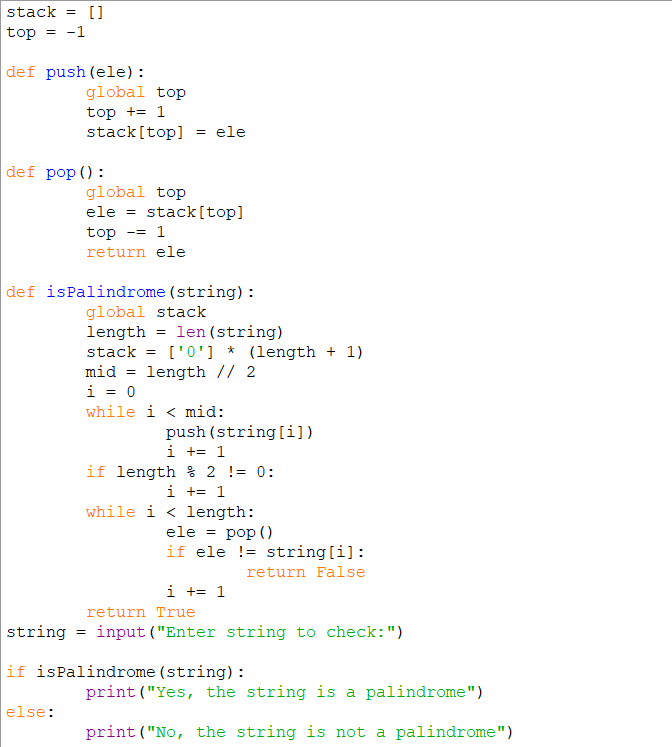


Output:



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

**Code:**



**Output:**



**Part B: SQL queries using one/two tables**

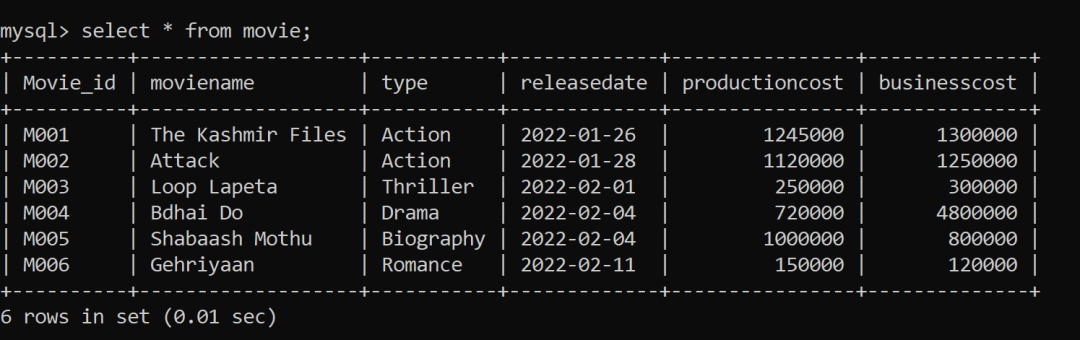
1. Consider the following MOVIE table and write the SQL queries based on it.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Movie\_ID | MovieName | Type | ReleaseDate | ProductionCost | BusinessCost |
| M001 | The Kashmir Files | Action | 2022/01/26 | 1245000 | 1300000 |
| M002 | Attack | Action | 2022/01/28 | 1120000 | 1250000 |
| M003 | Looop Lapeta | Thriller | 2022/02/01 | 250000 | 300000 |
| M004 | Badhai Do | Drama | 2022/02/04 | 720000 | 68000 |
| M005 | Shabaash Mithu | Biography | 2022/02/04 | 1000000 | 800000 |
| M006 | Gehraiyaan | Romance | 2022/02/11 | 150000 | 120000 |

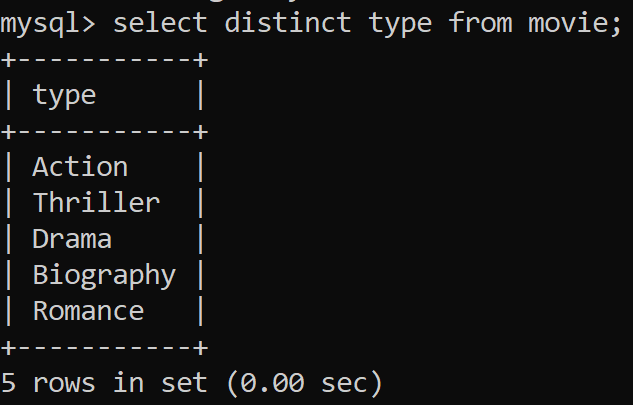
1. Display all information from movie.
2. Display the type of movies.
3. Display movieid, moviename, total\_eraning by showing the business done by the movies. Claculate the business done by movie using the sum of productioncost and businesscost.
4. Display movieid, moviename and productioncost for all movies with productioncost greater thatn 150000 and less than 1000000.
5. Display the movie of type action and romance.
6. Display the list of movies which are going to release in February, 2022.

**Code/Output:**

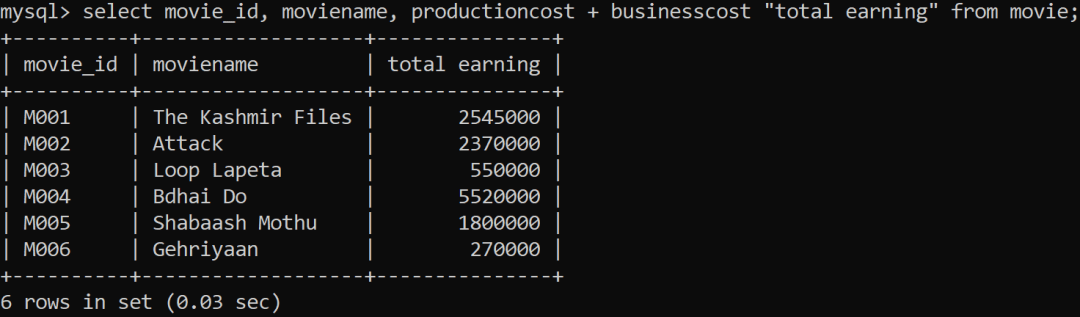
1.



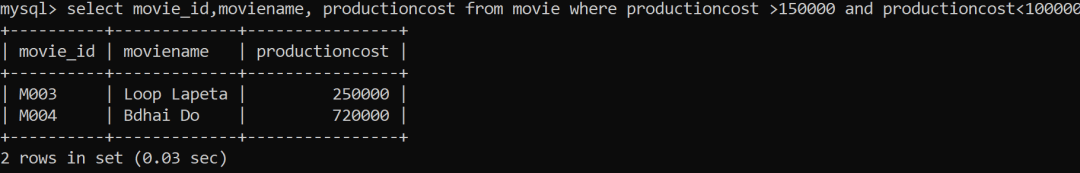
2.



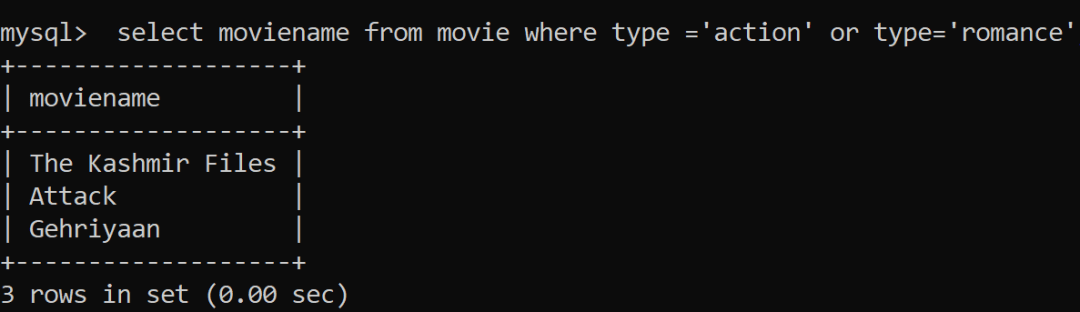
3.



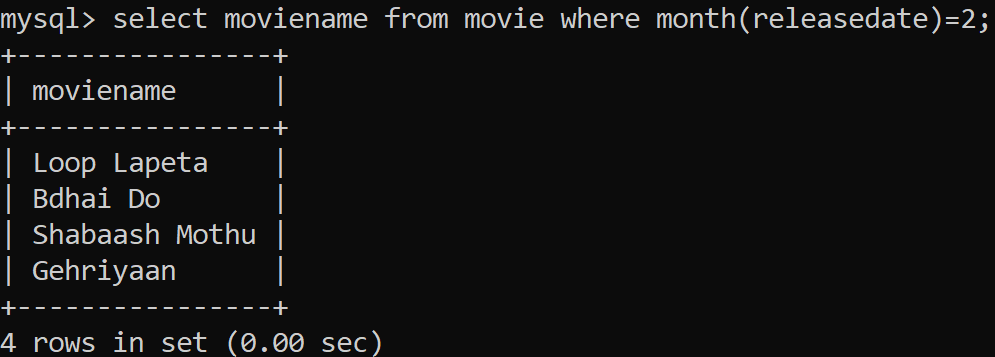
4.



5.



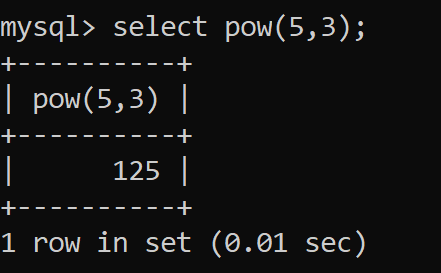
6.



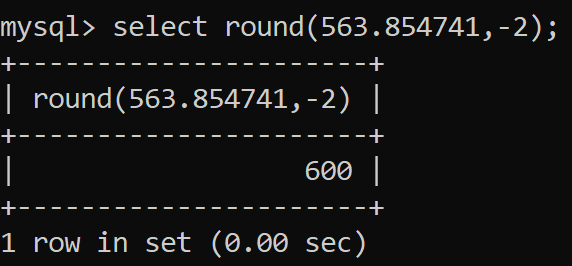
**2. (Based on Functions)**

1. Write a query to display cube of 5.
2. Write a query to display the number 563.854741 rounding off to the next hnudred.
3. Write a query to display “put” from the word “Computer”.
4. Write a query to display today’s date into DD.MM.YYYY format.
5. Write a query to display ‘DIA’ from the word “MEDIA”.
6. Write a query to display moviename – type from the table movie.
7. Write a query to display first four digits of productioncost.
8. Write a query to display last four digits of businesscost.
9. Write a query to display weekday of release dates.
10. Write a query to display dayname on which movies are going to be released.

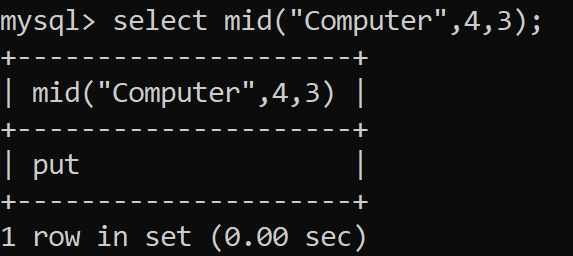
1.



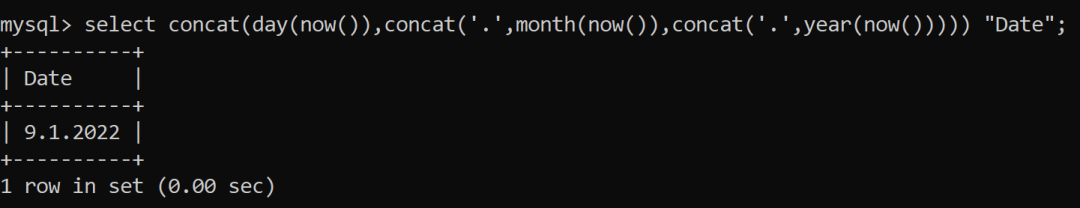
2.



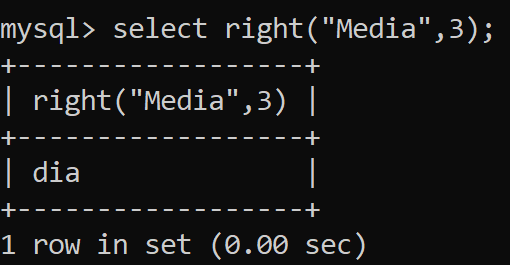
3.



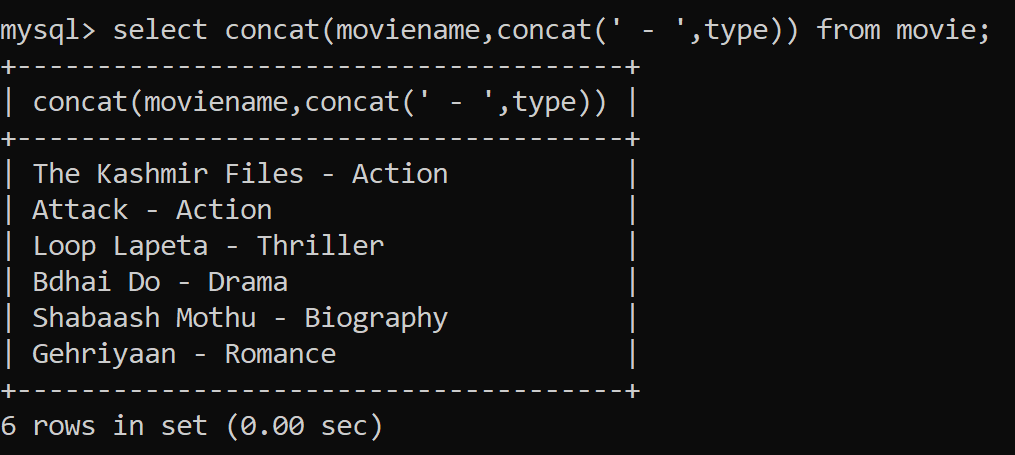
4.



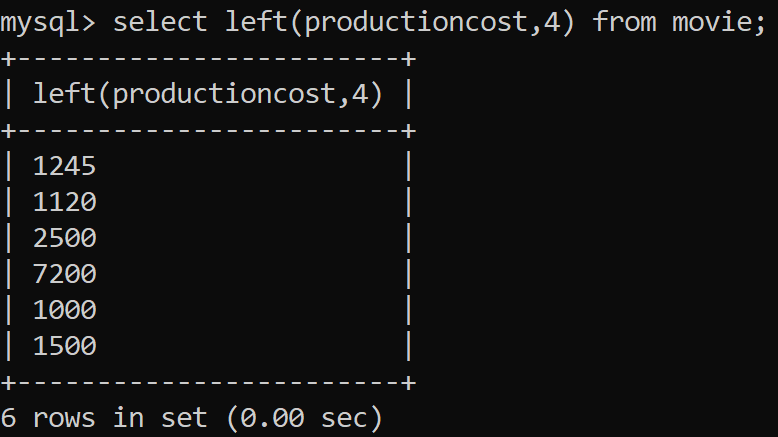
5.



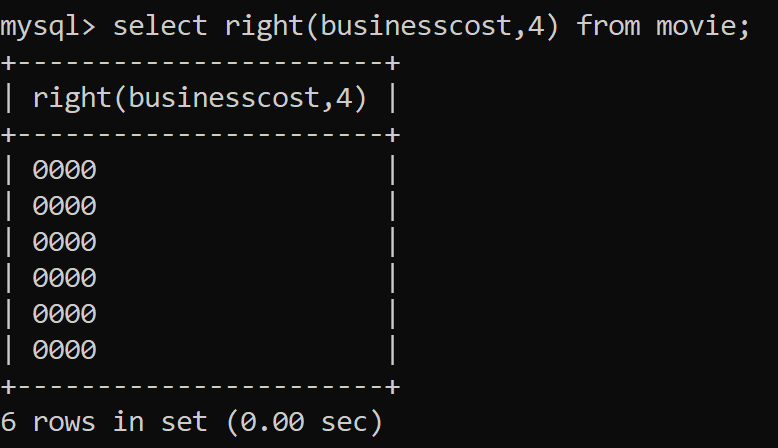
6.



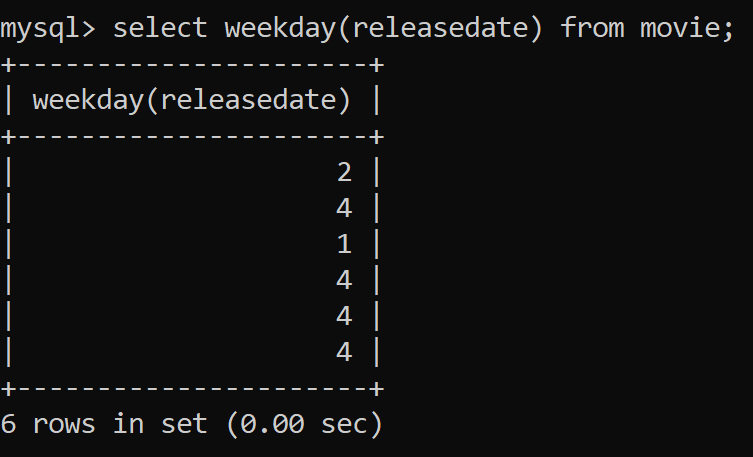
7.



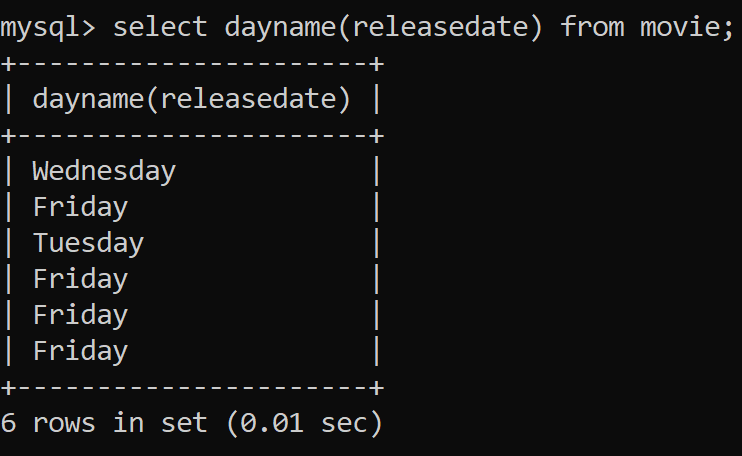
8.



9.



10.



**3. (DDL Commands)**

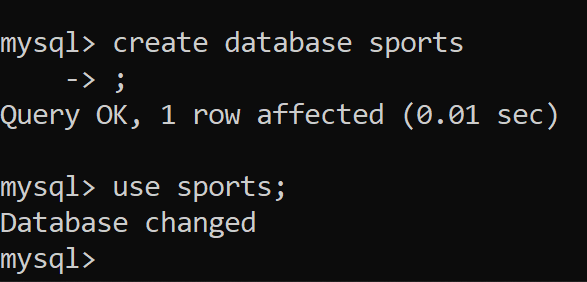
Suppose your school management has decided to conduct cricket matches between students of Class XI and Class XII. Students of each class are asked to join any one of the four teams – Team Titan, Team Rockers, Team Magnet and Team Hurricane. During summer vacations, various matches will be conducted between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

1. Create a database “Sports”.
2. Create a table “TEAM” with following considerations:
   * It should have a column TeamID for storing an integer value between 1 to 9, which refers to unique identification of a team.
   * Each TeamID should have its associated name (TeamName), which should be a string of length not less than 10 characters.
   * Using table level constraint, make TeamID as the primary key.
   * Show the structure of the table TEAM using a SQL statement.
   * As per the preferences of the students four teams were formed as given below. Insert these four rows in TEAM table:
     + Row 1: (1, Tehlka)
     + Row 2: (2, Toofan)
     + Row 3: (3, Aandhi)
     + Row 3: (4, Shailab)
   * Show the contents of the table TEAM using a DML statement.
3. Now create another table MATCH\_DETAILS and insert data as shown below. Choose appropriate data types and constraints for each attribute.

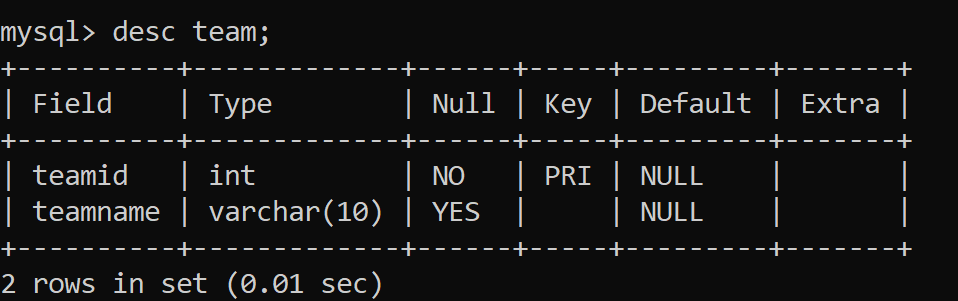
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| MatchID | MatchDate | FirstTeamID | SecondTeamID | FirstTeamScore | SecondTeamScore |
| M1 | 2021/12/20 | 1 | 2 | 107 | 93 |
| M2 | 2021/12/21 | 3 | 4 | 156 | 158 |
| M3 | 2021/12/22 | 1 | 3 | 86 | 81 |
| M4 | 2021/12/23 | 2 | 4 | 65 | 67 |
| M5 | 2021/12/24 | 1 | 4 | 52 | 88 |
| M6 | 2021/12/25 | 2 | 3 | 97 | 68 |

**Code/Output:**

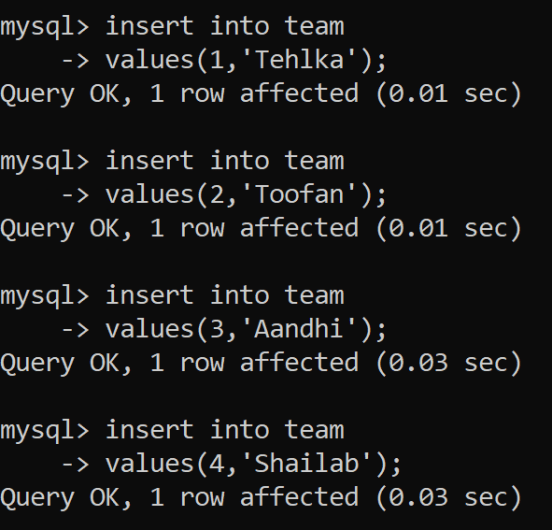
[1] create database sports



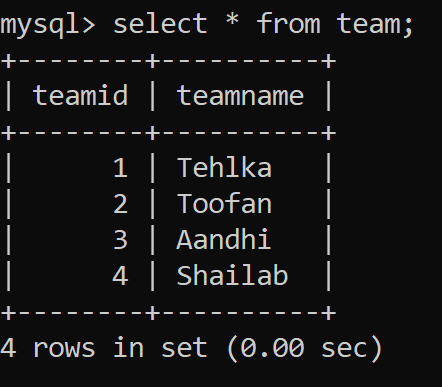
[2] Creating table with the given specification



Inserting data:



Show the content of table – team:



create table match\_details

-> (matchid varchar(2) primary key,

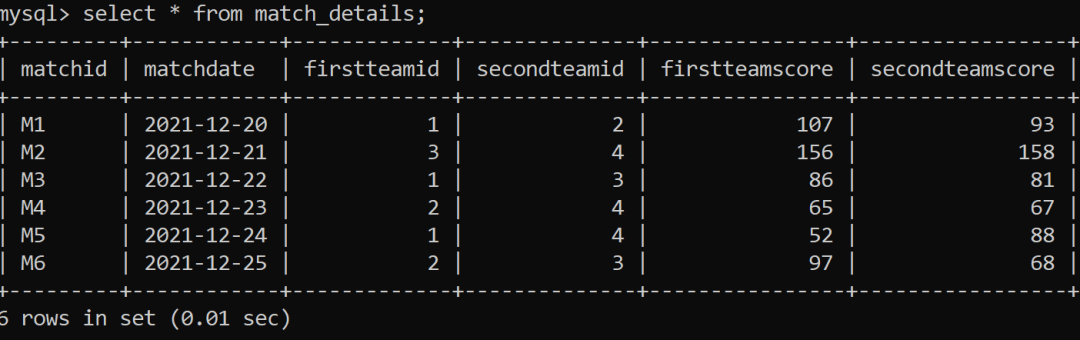
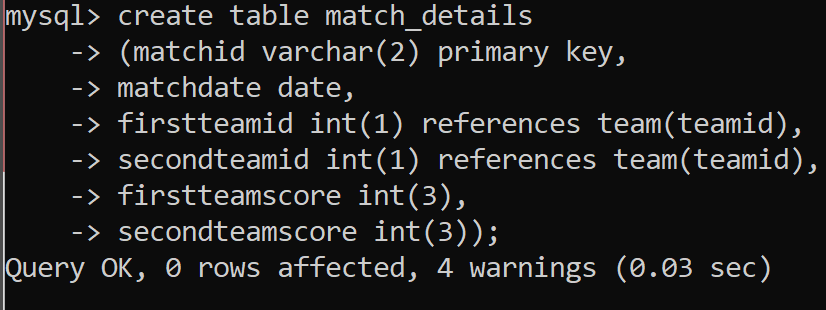
-> matchdate date,

-> firstteamid int(1) references team(teamid),

-> secondteamid int(1) references team(teamid),

-> firstteamscore int(3),

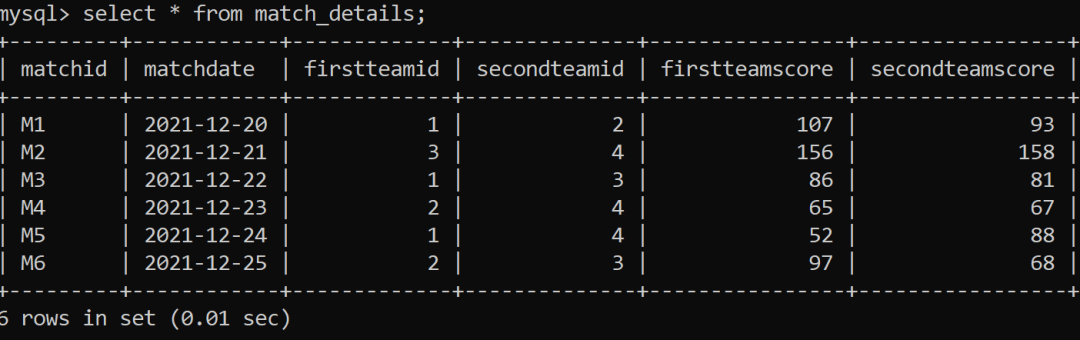
-> secondteamscore int(3));

****

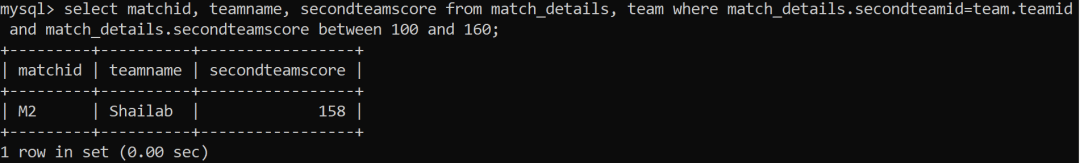
**4. (Based on Two Tables)**

1. Display the matchid, teamid, teamscore whoscored more than 70 in first ining along with team name.
2. Display matchid, teamname and secondteamscore between 100 to 160.
3. Display matchid, teamnames along with matchdates.
4. Display unique team names
5. Display matchid and matchdate played by Anadhi and Shailab.

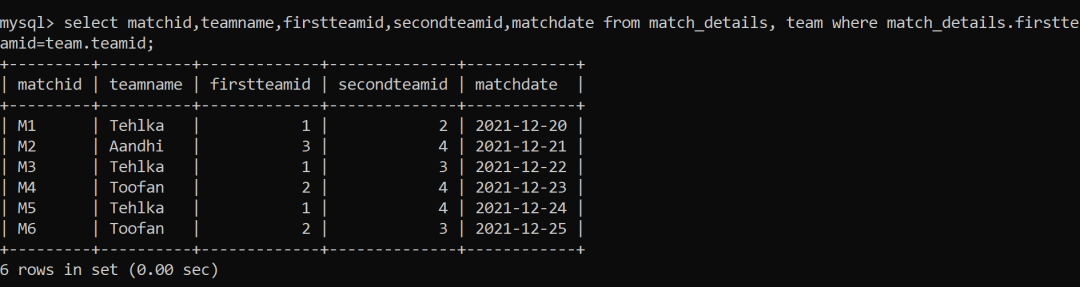
[1]



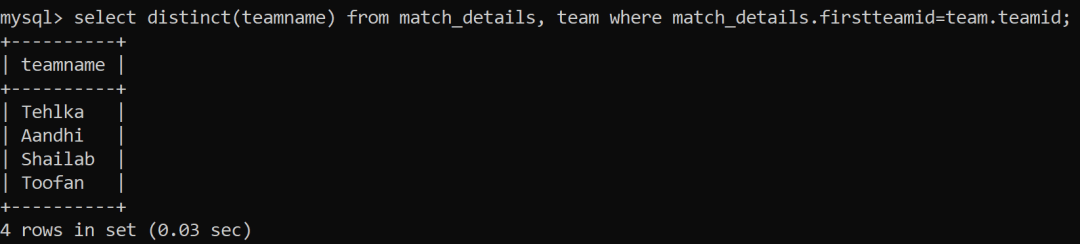
[2]



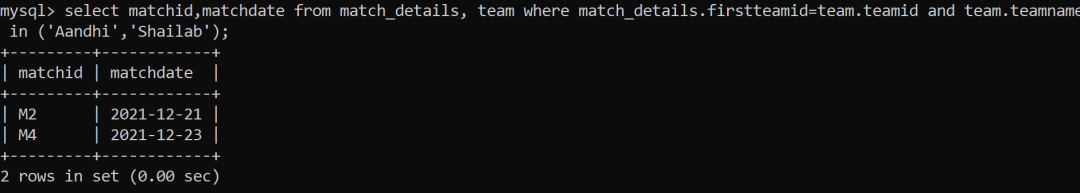
[3]



[4]



[5]



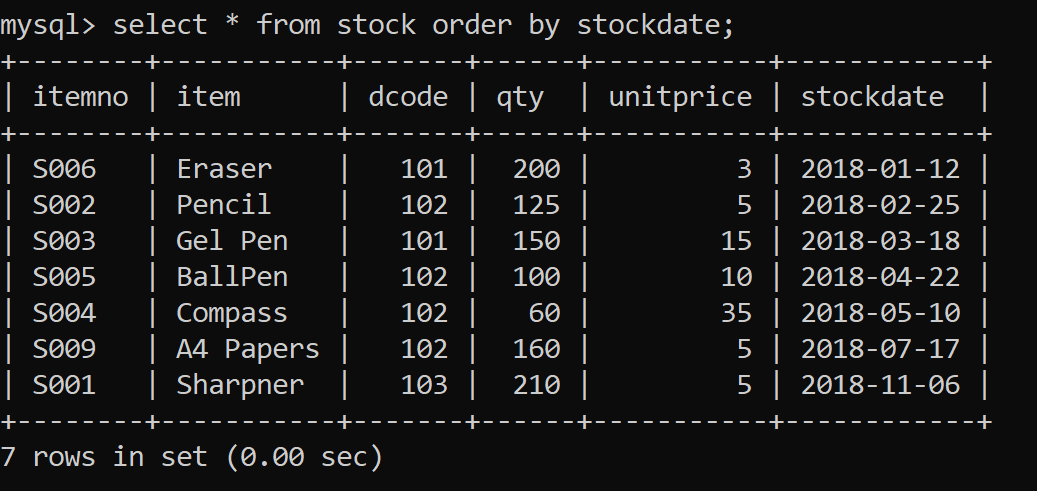
**5. (Group by , Order By)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| itemno | item | dcode | qty | unitprice | stockdate |
| S005 | Ballpen | 102 | 100 | 10 | 2018/04/22 |
| S003 | Gel Pen | 101 | 150 | 15 | 2018/03/18 |
| S002 | Pencil | 102 | 125 | 5 | 2018/02/25 |
| S006 | Eraser | 101 | 200 | 3 | 2018/01/12 |
| S001 | Sharpner | 103 | 210 | 5 | 2018/06/11 |
| S004 | Compass | 102 | 60 | 35 | 2018/05/10 |
| S009 | A4 Papers | 102 | 160 | 5 | 2018/07/17 |

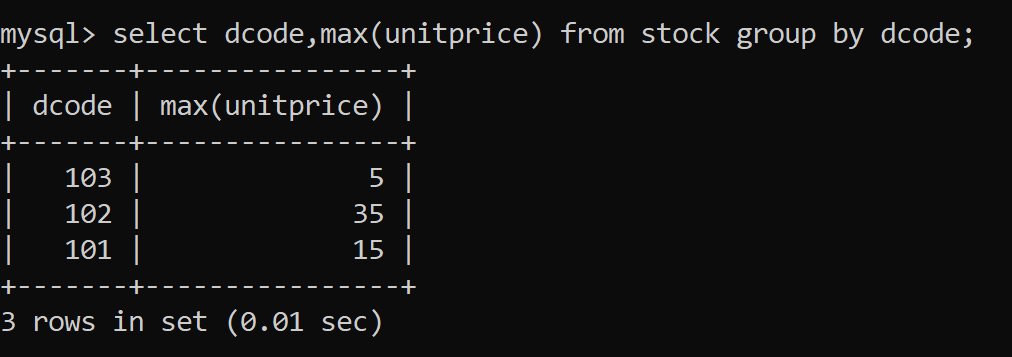
Consider the following table stock table to answer the queries:

1. Display all the items in the ascending order of stockdate.
2. Display maximum price of items for each dealer individually as per dcode from stock.
3. Display all the items in descending orders of itemnames.
4. Display average price of items for each dealer individually as per doce from stock which avergae price is more than 5.
5. Diisplay the sum of quantity for each dcode.

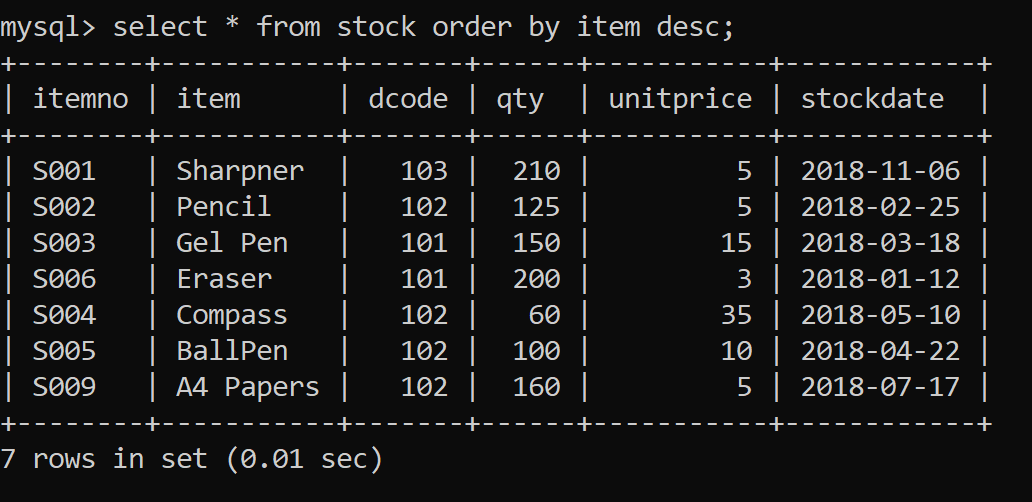
[1]



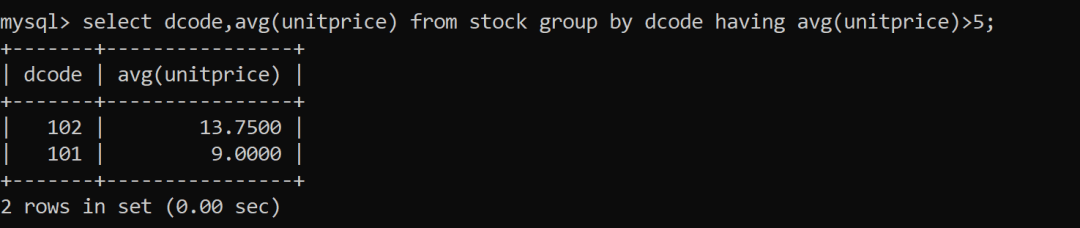
[2] select dcode,max(unitprice) from stock group by code;



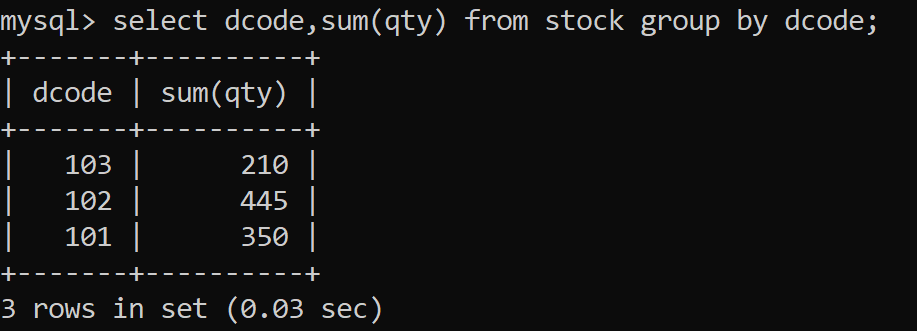
[3]



[4]



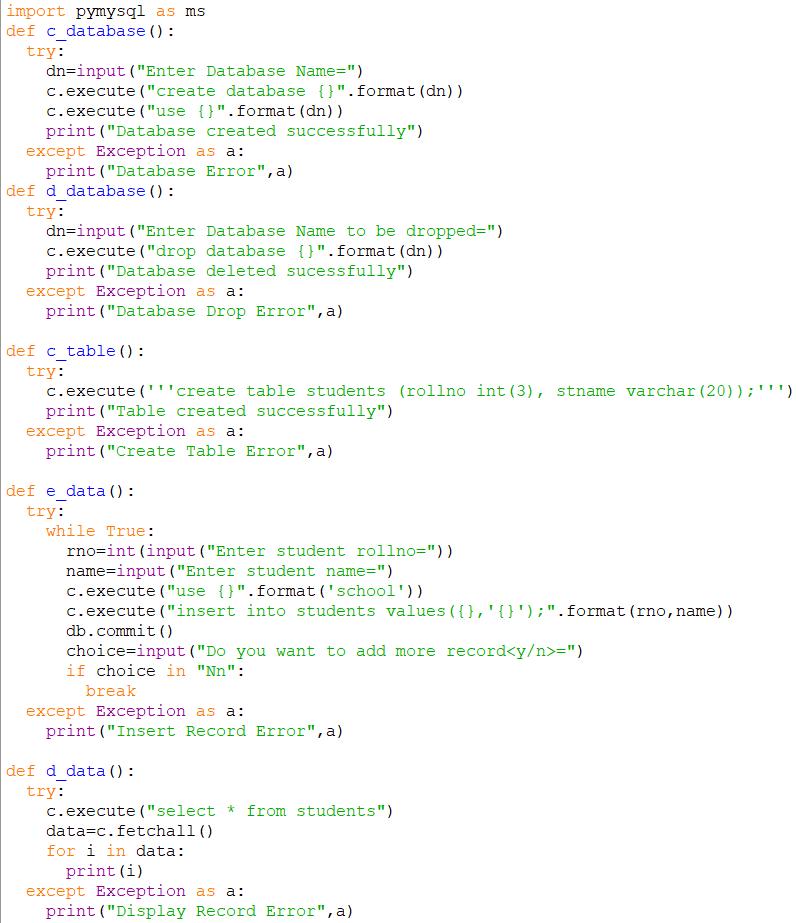
[5]

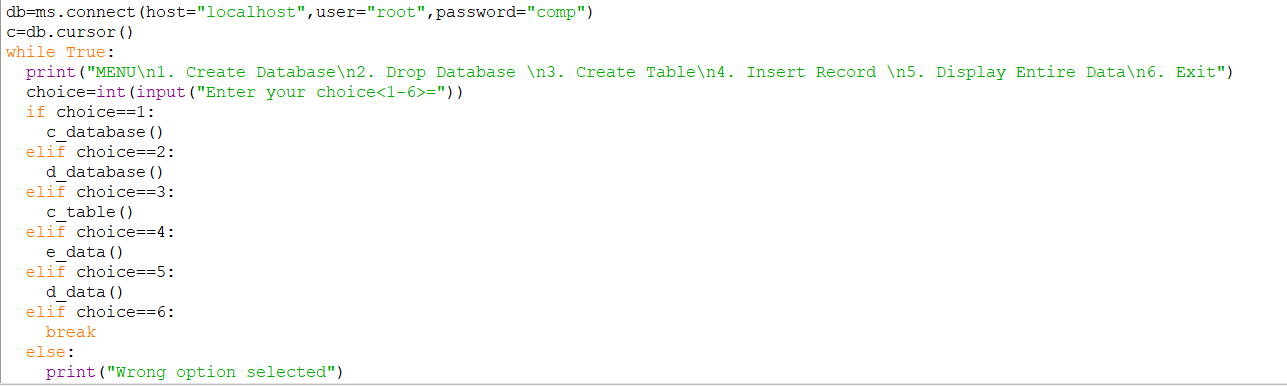


**Part C: SQL Database Connectivity 2 Programs**

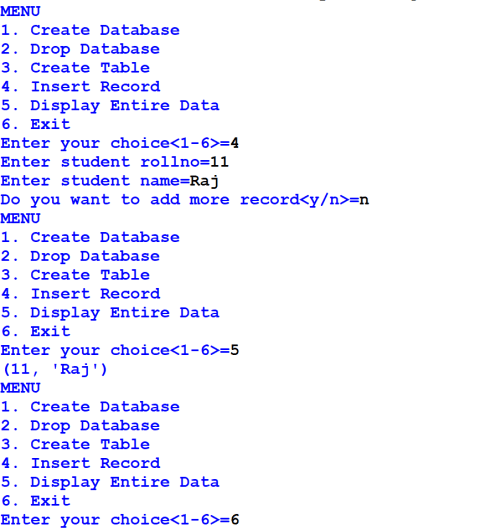
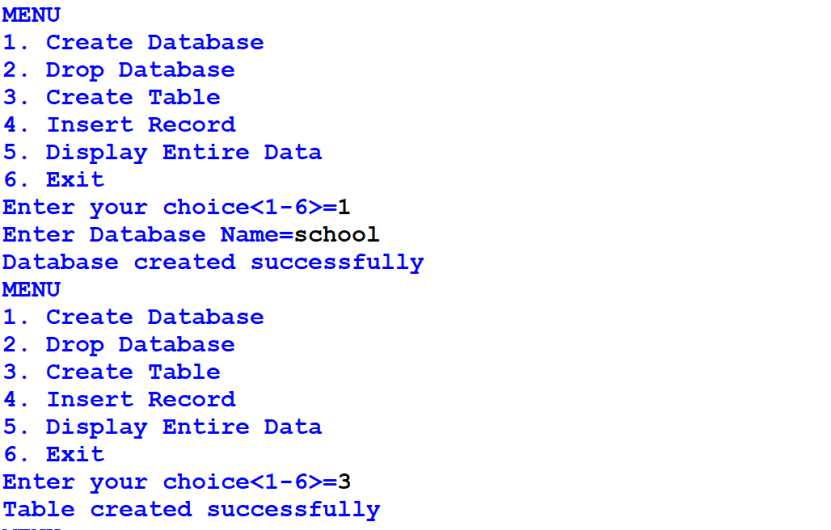
1. Write a MySQL connectivity program in Python to
   * Create a database school
   * Create a table students with the specifications – ROLLNO integer, STNAME character(10) in MySQL and perform the following operations:
     + Insert two records in it
     + Display the contents of the table
2. Perform all the operations with reference to table ‘students’ through MySQL-Python connectivity.

Code:





Output:



[2] using mysqlconnector



