

15

Python

programs

Write a program to convert decimal to binary, octal and hexadecimal

Program:

```
1 # Change this line for a different result
2
3 dec = 198
4
5 print("The decimal value of",dec,"is:")
6
7 print(bin(dec),"in binary.")
8
9 print(oct(dec),"in octal.")
10
11 print(hex(dec),"in hexadecimal.")
12
13
14
15
16
17
18
```

Output:

```
The decimal value of 198 is:
0b11000110 in binary.
0o306 in octal.
0xc6 in hexadecimal.
>
```

Write a program to check if a number is positive, negative or 0.

Program:

```
1 num = float(input("Enter a number: "))
2
3 if num > 0:
4     print("Positive number")
5 elif num == 0:
6     print("Zero")
7 else:
8     print("Negative number")
9
10
11
12
13
14
15
```

Output:

```
Enter a number: -222.8
Negative number
```

Write a program to find ASCII value of character

Program:

```
1 # Change this value for a different result
2
3 c = 'p'
4
5 # Uncomment to take character from user
6 #c = input("Enter a character: ")
7
8 print("The ASCII value of '" + c + "' is",ord(c))
```

Output:

```
The ASCII value of 'p' is 112
>
```

Write a program to check whether a string is Palindrome or not.

Program:

```
1 # change this value for a different output
2 my_str = 'aIbohPhoBiA'
3
4 # make it suitable for caseless comparison
5 my_str = my_str.casefold()
6
7 # reverse the string
8 rev_str = reversed(my_str)
9
10 # check if the string is equal to its reverse
11 if list(my_str) == list(rev_str):
12     print("It is palindrome")
13 else:
14     print("It is not palindrome")
```

Output:

It is palindrome

Write a program to count the number of each vowel

Program:

```
1 # string of vowels
2 vowels = 'aeiou'
3
4 # change this value for a different result
5 ip_str = 'Hello, have you tried our tutorial section yet?'
6
7 # uncomment to take input from the user
8 #ip_str = input("Enter a string: ")
9
10 # make it suitable for caseless comparisons
11 ip_str = ip_str.casefold()
12
13 # make a dictionary with each vowel a key and value 0
14 count = {}.fromkeys(vowels, 0)
15
16 # count the vowels
17 for char in ip_str:
18     if char in count:
19         count[char] += 1
20
21 print(count)
```

Output:

```
{'a': 2, 'e': 5, 'i': 3, 'o': 5, 'u': 3}
```

```
>
```

Write a program to print the Fibonacci Sequence

Program:

```
1 # change this value for a different result
2 nterms = 10
3
4 # uncomment to take input from the user
5 #nterms = int(input("How many terms? "))
6
7 # first two terms
8 n1 = 0
9 n2 = 1
10 count = 0
11
12 # check if the number of terms is valid
13 if nterms <= 0:
14     print("Please enter a positive integer")
15
16 elif nterms == 1:
17     print("Fibonacci sequence upto",nterms,":")
18     print(n1)
19
20 else:
21     print("Fibonacci sequence upto",nterms,":")
22
23     while count < nterms:
24         print(n1,end=' , ')
25         nth = n1 + n2
26
27         # update values
28         n1 = n2
29         n2 = nth
30         count += 1
31
32 print()
33
```

34 Output:

Fibonacci sequence upto 10 :

0 , 1 , 1 , 2 , 3 , 5 , 8 , 13 , 21 , 34 ,

>

Write a program to find Armstrong numbers in an interval.

Program:

```
1 lower = 100
2 upper = 2000
3
4 # To take input from the user
5 # lower = int(input("Enter lower range: "))
6 # upper = int(input("Enter upper range: "))
7
8 for num in range(lower, upper + 1):
9
10     # order of number
11     order = len(str(num))
12
13     # initialize sum
14     sum = 0
15
16     # find the sum of the cube of each digit
17     temp = num
18
19     while temp > 0:
20         digit = temp % 10
21         sum += digit ** order
22         temp //= 10
23
24     if num == sum:
25         print(num)
26
27
28
```

Output:

```
153
370
371
407
1634
```


Write a program to find the Largest among three numbers

Program:

```
1 # Python program to find the largest number among the three input
  numbers
2
3 num1 = 10
4 num2 = 14
5 num3 = 12
6
7 # uncomment following lines to take three numbers from user
8 #num1 = float(input("Enter first number: "))
9 #num2 = float(input("Enter second number: "))
10 #num3 = float(input("Enter third number: "))
11
12 if (num1 >= num2) and (num1 >= num3):
13     largest = num1
14
15 elif (num2 >= num1) and (num2 >= num3):
16     largest = num2
17
18 else:
19     largest = num3
20
21 print("The largest number
  between", num1, ", ", num2, "and", num3, "is", largest)
22
23
24
25
```

Output:

The largest number between 10 , 14 and 12 is 14

>

Write a program to check if a year is leap or not.

Program:

```
1 # Python program to check if the input year is a leap year or not
2
3 year = 2024
4
5 # To get year (integer input) from the user
6 # year = int(input("Enter a year: "))
7
8 if (year % 4) == 0:
9     if (year % 100) == 0:
10
11         if (year % 400) == 0:
12             # if block of 400
13             print("{0} is a leap year".format(year))
14         else:
15             # else block of 400
16             print("{0} is not a leap year".format(year))
17
18     else:
19         # else block of 100
20         print("{0} is a leap year".format(year))
21
22 else:
23     # else block of 4
24     print("{0} is not a leap year".format(year))
25
26
27
28
```

Output:

2024 is a leap year

Write a program to display calendar

Program:

```
1 # import module
2 import calendar
3
4 yy = 2024
5 mm = 2
6
7 # To ask month and year from the user
8 # yy = int(input("Enter year: "))
9 # mm = int(input("Enter month: "))
10
11 # display the calendar
12 print(calendar.month(yy, mm))
13
14
15
16
17
```

Output:

```
February 2024
Mo Tu We Th Fr Sa Su
      1  2  3  4
 5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29
```

Write a program to find square root.

Program:

```
1 # Change this value and test yourself
2
3 num = 1225
4
5 # ** is used for exponentiation
6 num_sqrt = num ** 0.5
7 print('The square root of', num, 'is:', num_sqrt )
8
9
10
11
12
13
```

Output:

```
The square root of 1225 is: 35.0
>
```

Write a program to find the factorial of a number..

Program :

```
1 # change the value for a different result
2
3 num = 7
4
5 # uncomment to take input from the user
6 #num = int(input("Enter a number: "))
7
8 factorial = 1
9
10 # check if the number is negative, positive or zero
11
12 if num < 0:
13     print("Sorry, factorial does not exist for negative numbers")
14
15 elif num == 0:
16     print("The factorial of 0 is 1")
17
18 else:
19     for i in range(1,num + 1):
20         factorial = factorial*i
21
22     print("The factorial of",num,"is",factorial)
23
24
25
```

Output:

The factorial of 7 is 5040

Write a program to calculate the area of a triangle.

Program:

```
1 # sides of a triangle
2
3 a = 3
4 b = 4
5 c = 5
6
7 # calculate the semi-perimeter
8 s = (a + b + c) / 2
9
10 # calculate the area
11 area = (s*(s-a)*(s-b)*(s-c)) ** 0.5
12
13 print('The area of the triangle is', area)
14
15
16
17
18
```

Output:

```
The area of the triangle is 6.0
```

```
>
```

Write a program to check if a number is odd or even .

Program:

```
1 # Python program to check if the input number is odd or even.
2
3 num = int(input("Enter a number: "))
4
5 # A number is even if division by 2 give a remainder of 0.
6 # If remainder is 1, it is odd number.
7
8 if (num % 2) == 0:
9     print("{0} is Even".format(num))
10
11 else:
12     print("{0} is Odd".format(num))
13
14
15
16
17
18
19
20
```

Output:

```
Enter a number: 18
18 is Even
```

>

Write a program to display multiplication table .

Program:

```
1 # Using while loop
2
3 num = 25
4
5 # To take input from the user
6 # num = int(input("Display multiplication table of? "))
7
8 count = 1
9
10 while count < 11:
11     print(num, ' x ', count, ' = ', num * count)
12     count = count + 1
13
14
15
16
17
18
19
```

Output:

```
25 x 1 = 25
25 x 2 = 50
25 x 3 = 75
25 x 4 = 100
25 x 5 = 125
25 x 6 = 150
25 x 7 = 175
25 x 8 = 200
25 x 9 = 225
25 x 10 = 250
>
```


5 SQL

queries

SQL COMMANDS EXERCISE - 1

(Basic Queries - I)

AIM:

To write Queries for the following Questions based on the given table:

EmpID	Name	Gender	Age	Dept	DOJ	Salary	City
1	Praveen	M	25	Sales	1989-06-08	20000	Chennai
2	Arun	M	29	Marketing	1989-09-26	22000	Chennai
3	Usha	F	27	Finance	1994-08-09	25000	Bangalore
4	Bala	M	31	Sales	1990-03-23	27000	NULL
5	Rani	F	28	Marketing	1990-04-23	27000	Mumbai
6	Nisha	F	26	NULL	1991-02-24	18000	Bangalore
7	Manoj	M	32	Finance	1982-05-06	30000	Goa

(a) Write a Query to Create a new database in the name of "EMPS".

CREATE DATABASE EMPS;

(b) Write a Query to Open the database EMPS.

USE EMPS;

(c) Write a Query to create the above table called: Info

CREATE TABLE INFO (EmpID int primary key, Name varchar(15), Gender varchar(3), Age int, Dept varchar(15), DOJ date, Salary int, City varchar(10));

(d) Write a Query to list all the existing database names.

SHOW DATABASES;

```
Database
-----
employees
hospital
hospitals
information_schema
ip_practicals
mysql
performance_schema
school
students
sys
```

(e) Write a Query to List all the tables that exists in the current database.

SHOW TABLES;

Output:

```
-----+-----
| Tables_in_ip_practicals |
+-----+-----
| info                     |
+-----+-----
```

(f) Write a Query to insert all the rows of above table into Info table.

```
INSERT INTO INFO VALUES (1,'Praveen','M', 25,'Sales','1989-06-08','20000','Chennai');
INSERT INTO INFO VALUES(2,'Arun','M',29,'Marketing','1989-09-26',22000,'Chennai');
INSERT INTO INFO VALUES(3,'Usha','F',27,'Finance','1994-08-09',25000,'Bangalore');
INSERT INTO INFO VALUES(4,'Bala','M',31,'Sales','1990-03-23',27000,NULL);
INSERT INTO INFO VALUES(5,'Rani','F',28,'Marketing','1990-04-23',27000,'Mumbai');
INSERT INTO INFO VALUES (6,'Nisha','F', 26, NULL,'1991-02-24', 18000,'Bangalore');
INSERT INTO INFO VALUES (7,'Manoj','M', 32,'Finance','1982-05-06', 30000,'Goa');
```

(g) Write a Query to display all the details of the Employees from the above table 'INFO'.

SELECT * FROM INFO;

Output:

EmpID	Name	Gender	Age	Dept	DOJ	Salary	City
1	Praveen	M	25	Sales	1989-06-08	20000	Chennai
2	Arun	M	29	Marketing	1989-09-26	22000	Chennai
3	Usha	F	27	Finance	1994-08-09	25000	Bangalore
4	Bala	M	31	Sales	1990-03-23	27000	NULL
5	Rani	F	28	Marketing	1990-04-23	27000	Mumbai
6	Nisha	F	26	NULL	1991-02-24	18000	Bangalore
7	Manoj	M	32	Finance	1982-05-06	30000	Goa

SQL COMMANDS EXERCISE - 2

(Basic Queries - II)

AIM:

To write Queries for the following Questions based on the given table:

EmpID	Name	Gender	Age	Dept	DOJ	Salary	City
1	Praveen	M	25	Sales	1989-06-08	20000	Chennai
2	Arun	M	29	Marketing	1989-09-26	22000	Chennai
3	Usha	F	27	Finance	1994-08-09	25000	Bangalore
4	Bala	M	31	Sales	1990-03-23	27000	NULL
5	Rani	F	28	Marketing	1990-04-23	27000	Mumbai
6	Nisha	F	26	NULL	1991-02-24	18000	Bangalore
7	Manoj	M	32	Finance	1982-05-06	30000	Goa

(a) Write a Query to Display Employees' name and City from the above table.

SELECT NAME, CITY FROM INFO;

Output:

NAME	CITY
Praveen	Chennai
Arun	Chennai
Usha	Bangalore
Bala	NULL
Rani	Mumbai
Nisha	Bangalore
Manoj	Goa

(b) Write a Query to Display all details of Employees who are living in Chennai.

SELECT * FROM INFO WHERE CITY='CHENNAI';

Output:

EmpID	Name	Gender	Age	Dept	DOJ	Salary	City
1	Praveen	M	25	Sales	1989-06-08	20000	Chennai
2	Arun	M	29	Marketing	1989-09-26	22000	Chennai

(c) Write a Query to get the name and salary of the employee whose salary is above 15000 and gender is not male.

```
SELECT NAME,SALARY FROM INFO WHERE SALARY >15000 AND GENDER<>'M';
```

Output:

NAME	SALARY
Usha	25000
Rani	27000
Nisha	18000

(d) Write a query to update increase 10% Salary of an employee whose City is 'CHENNAI' and Gender is 'MALE'.

```
UPDATE INFO SET SALARY=SALARY+ (SALARY*0.10) WHERE CITY='CHENNAI' AND GENDER='MALE';
```

Output (After Updating):

```
mysql> SELECT * FROM INFO;
```

EmpID	Name	Gender	Age	Dept	DOJ	Salary	City
1	Praveen	M	25	Sales	1989-06-08	22000	Chennai
2	Arun	M	29	Marketing	1989-09-26	24200	Chennai
3	Usha	F	27	Finance	1994-08-09	25000	Bangalore
4	Bala	M	31	Sales	1990-03-23	27000	NULL
5	Rani	F	28	Marketing	1990-04-23	27000	Mumbai
6	Nisha	F	26	NULL	1991-02-24	18000	Bangalore
7	Manoj	M	32	Finance	1982-05-06	30000	Goa

SQL COMMANDS EXERCISE - 3

(Aggregate Functions, Order By Group By, Having Clause)

AIM:

To write Queries for the following Questions based on the given table:

EmpID	Name	Gender	Age	Dept	DOJ	Salary	City
1	Praveen	M	25	Sales	1989-06-08	20000	Chennai
2	Arun	M	29	Marketing	1989-09-26	22000	Chennai
3	Usha	F	27	Finance	1994-08-09	25000	Bangalore
4	Bala	M	31	Sales	1990-03-23	27000	NULL
5	Rani	F	28	Marketing	1990-04-23	27000	Mumbai
6	Nisha	F	26	NULL	1991-02-24	18000	Bangalore
7	Manoj	M	32	Finance	1982-05-06	30000	Goa

(a) Write a Query to list names of Employees in Descending order.

SELECT NAME FROM INFO ORDER BY NAME DESC;

Output:

```
+-----+
| NAME   |
+-----+
| Usha   |
| Rani   |
| Praveen |
| Manoj  |
| Bala   |
| Arun   |
+-----+
```

(b) Write a Query to find a total salary of all employees.

SELECT SUM(SALARY) FROM INFO;

Output:

```
+-----+
| SUM(SALARY) |
+-----+
| 159820      |
+-----+
```

(c) Write a Query to display maximum salary and minimum salary of employees.

SELECT MAX(SALARY), MIN(SALARY) FROM INFO;

Output:

```
+-----+
| MAX(SALARY) | MIN(SALARY) |
+-----+
| 30000       | 18000       |
+-----+
```

(d) Write a Query to count the number of employees earning more than 25000.

SELECT COUNT(SALARY) FROM INFO WHERE SALARY>25000;

Output:

COUNT(SALARY)
4

(e) Write a query to display sum of salary of the employees grouped by department wise.

SELECT DEPT, SUM(SALARY) FROM INFO GROUP BY DEPT;

Output:

DEPT	SUM(SALARY)
Sales	51200
Marketing	53620
Finance	55000

(f) Write a query to display the department names where number of employees are greater than or equal to 2.

SELECT DEPT FROM INFO GROUP BY DEPT HAVING COUNT(*)>=2;

Output:

DEPT
Sales
Marketing
Finance

SOL COMMANDS EXERCISE - 4 (Date Functions)

AIM:

To write Queries for the following Questions based on the given table:

Rollno	Name	Gender	AGE	DEPT	DOA	Percentage
1	Arun	M	16	CS	1997-01-10	97.8
2	ANKIT	M	17	IP	1998-06-24	95.2
3	ANU	F	15	HINDI	1996-07-12	81
4	BALA	F	17	IP	1999-07-01	75.5
5	CHARAN	M	16	HINDI	1997-06-27	92.13
6	DEEPA	F	16	HISTORY	1997-06-27	60.54
7	DINESH	M	15	CS	1996-08-25	82

- (a) Write a Query to display student name and month of date of admission of all students.

```
SELECT NAME, MONTH(DOA) FROM STU;
```

Output:

NAME	MONTH(DOA)
ARUN	1
ANKIT	6
ANU	7
BALA	7
CHARAN	6
DEEPA	6
DINESH	8

- (b) Write a Query to display Student name and day name of the students' DOA of the table STU.

```
SELECT NAME, DAYNAME(DOA) FROM STU;
```

Output:

NAME	DAYNAME(DOA)
ARUN	Friday
ANKIT	Wednesday
ANU	Friday
BALA	Thursday
CHARAN	Friday
DEEPA	Friday
DINESH	Sunday

(c) Write a query to display the joining year of IP students.

SELECT YEAR(DOA) FROM STU WHERE DEPT='IP'

Output:

YEAR(DOA)
1998
1999

(d) Write a Query to Display the month for the date_of_birth of all students.

SELECT NAME, MONTHNAME(DOA)FROM STU;

Output:

NAME	MONTHNAME(DOA)
ARUN	January
ANKIT	June
ANU	July
BALA	July
CHARAN	June
DEEPA	June
DINESH	August

SOL COMMANDS EXERCISE - 5

(Text Functions)

AIM:

To write Queries for the following Questions based on the given table -"STU":

Rollno	Name	Gender	AGE	DEPT	DOA	Percentage
1	Arun	M	16	CS	1997-01-10	97.8
2	ANKIT	M	17	IP	1998-06-24	95.2
3	ANU	F	15	HINDI	1996-07-12	81
4	BALA	F	17	IP	1999-07-01	75.5
5	CHARAN	M	16	HINDI	1997-06-27	92.13
6	DEEPA	F	16	HISTORY	1997-06-27	60.54
7	DINESH	M	15	CS	1996-08-25	82

(a) Write a Query to display Department name in lower case letters.

```
SELECT LCASE(DEPT) FROM STU;
```

Output:

LCASE(DEPT)
cs
ip
hindi
ip
hindi
history
cs

(b) Write a Query to display department name and its respective number of characters in Dept column.

```
SELECT DEPT,LENGTH(DEPT) FROM STU;
```

Output:

DEPT	LENGTH(DEPT)
COMPUTER	8
IP	2
HINDI	5
IP	2
HINDI	5
HISTORY	7
COMPUTER	8

(c) Write a Query to display first 2 characters of the column Name.

```
SELECT LEFT(NAME,2) FROM STU;
```

Output:

LEFT(NAME,2)
AR
AN
AN
BA
CH
DE
DI

(d) Write a Query to display first 2 characters of the column Name.

```
SELECT RIGHT(NAME,2) FROM STU;
```

Output:

RIGHT(NAME,2)
UN
IT
NU
LA
AN
PA
SH

(e) Write a query to display the names of all students and extract five characters from the third position of the 'Name' field.

```
SELECT SUBSTR(NAME,3,5) FROM STU;
```

Output:

SUBSTR(NAME,3,5)
UN
KIT
U
LA
ARAN
EPA
NESH

4 Python ~

SQL

Connectivity

Queries

Python interface with MySQL

Write a function to search a record stored in a table using python and MySQL interface.

```
def search_roll():
    import mysql.connector
    db =
mysql.connector.connect(host="localhost",user="root",passwd="admin",databa
se="test")
    try:
        z=0
        roll=int(input("Enter roll no to search "))
        c = db.cursor()
        sql='select * from student;'
        c.execute(sql)
        countrow=c.execute(sql)
        print("number of rows : ",countrow)
        data=c.fetchall()
        for eachrow in data:
            r=eachrow[0]
            n=eachrow[1]
            p=eachrow[2]
            if(r==roll):
                z=1
                print(r,n,p)
            if(z==0):
                print("Record is not present")
    except:
        db.rollback()
        db.close()
# function calling
search_roll()
```

Output-

```
Enter roll no to search 101
number of rows : 2
101 amit 97
>>>
```

Write a function to insert a record in table using python and MySQL interface.

```
def insert_data():
#take input for the details and then save the record in the database
#to insert data into the existing table in an existing database
    import mysql.connector
    db =
mysql.connector.connect(host="localhost",user="root",password="admin")
    c = db.cursor()
    r=int(input("Enter roll no "))
    n=input("Enter name ")
    p=int(input("Enter per "))
    try:
        c.execute("insert into student (roll,name,per) values
(%s,%s,%s)", (r,n,p))
        db.commit()
        print("Record saved")
    except:
        db.rollback()
        db.close()
# function calling
insert_data()
```

Output:

```
Enter roll no 101
Enter name amit
Enter per 97
Record saved
>>>
```

Write a function to display all the records stored in a table using python and MySQL interface.

```
def display_all():
    import mysql.connector
    db =
mysql.connector.connect(host='localhost',user='root',passwd='admin',database='test4')
    try:
        c = db.cursor()
        sql='select * from student;'
        c.execute(sql)
        countrow=c.execute(sql)
        print("number of rows : ",countrow)
        data=c.fetchall()
        print("=====")
        print("Roll No Name Per ")
        print("=====")
        for eachrow in data:
            r=eachrow[0]
            n=eachrow[1]
            p=eachrow[2]
            print(r, ' ',n, ' ',p)
            print("=====")
    except:
        db.rollback()
        db.close()
# function calling
display_all()
```

Output:

```
number of rows : 2
=====
Roll No Name Per
=====
102 aaa 99
101 amit 97
=====
>>>
```

1. Code to create a new database from python, use desc command and insert new records.

```
1
2
3 import mysql.connector
4 cn=mysql.connector.connect(user='root',password='root',host='localhost',charset='utf8')
5 con=cn.cursor()
6 con.execute("create database practice_python1;")
7 con.execute("use practice_python1;")
8 con.execute("create table t1(cust_id char(4) primary key,cust_name varchar(15), address varchar(15));")
9 con.execute("desc t1;")
10 for t in con:
11     print(t)
12 con.execute("select * from t1;")
13 for t1 in con:
14     print(t1)
15 con.execute("insert into t1 values('A001','Rishi','Haldwani');")
16 con.execute("insert into t1 values('A002','Somya','Haldwani');")
17 con.execute("insert into t1 values('A003','Roohi','Haldwani');")
18 con.execute("insert into t1 values('A004','Varun','Haldwani');")
19 cn.commit()
20 for a in con:
21     print(a)
22 con.execute("select * from t1;")
23 for t2 in con:
24     print(t2)
25 cn.commit()
```

2. See the bottom right pane for its output

```
1
2
3 import mysql.connector
4 cn=mysql.connector.connect(user='root',password='r
5 con=cn.cursor()
6 con.execute("create database practice_python1;")
7 con.execute("use practice_python1;")
8 con.execute("create table t1(cust_id char(4) prima
9 con.execute("desc t1;")
10 for t in con:
11     print(t)
12 con.execute("select * from t1;")
13 for t1 in con:
14     print(t1)
15 con.execute("insert into t1 values('A001','Rishi'
16 con.execute("insert into t1 values('A002','Somya'
17 con.execute("insert into t1 values('A003','Roohi'
18 con.execute("insert into t1 values('A004','Varun'
19 cn.commit()
20 for a in con:
21     print(a)
22 con.execute("select * from t1;")
23 for t2 in con:
24     print(t2)
25 cn.commit()
```

Usage
Here you can get help of any object by pressing Ctrl+F in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior Preferences > Help.

```
In [2]: runfile('C:/Users/Lenovo/new_db.py', wdir='C:/Users/Lenovo')
{'cust_id': 'char(4)', 'NO', 'PRI', None, ''}
{'cust_name': 'varchar(15)', 'YES', '', None, ''}
{'address': 'varchar(15)', 'YES', '', None, ''}
{'A001', 'Rishi', 'Haldwani'}
{'A002', 'Somya', 'Haldwani'}
{'A003', 'Roohi', 'Haldwani'}
{'A004', 'Varun', 'Haldwani'}

In [3]: |
```

3. The Mysql view.

```
mysql> use practice_python1;
Database changed
mysql> select * from t1;
+-----+-----+-----+
| cust_id | cust_name | address |
+-----+-----+-----+
| A001    | Rishi    | Haldwani |
| A002    | Somya    | Haldwani |
| A003    | Roohi    | Haldwani |
| A004    | Varun    | Haldwani |
+-----+-----+-----+
4 rows in set (0.00 sec)
```

So in these codes (1-3) a new database is created, its table structure is seen and a new table is

4. Another code to just view the data from mysql in Python.

```
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Lenovo\select.py
new_debug x Unintended.py x select.py
1 # -*- coding: utf-8 -*-
2 """
3 Created on Fri Aug 27 23:27:21 2021
4
5 @author: Lenovo
6 """
7
8 import mysql.connector
9 #import pandas as pd
10 cn=mysql.connector.connect(user='root',password='root',host='localhost',charset='utf8')
11 con=cn.cursor()
12 con.execute("use xii;")
13 con.execute("select * from a;")
14 for a in con:
15     print(a)
16 cn.commit()
```

5. Output of the above code.

```
Console I/A x
In [4]: runfile('C:/Users/Lenovo/select.py', wdir='C:/Users/Lenovo')
(101, 'Siddharth', 19)
(102, 'Kushagra', 18)
(103, 'Usman', 18)
In [5]: |
```

6. The Mysql view of the table and its records.

```
mysql> use xii;
Database changed
mysql> select * from a;
+-----+-----+-----+
| rollno | name      | age  |
+-----+-----+-----+
| 101    | Siddharth | 19   |
| 102    | Kushagra  | 18   |
| 103    | Usman     | 18   |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

So in these codes (4-6) an already made database is opened and simple select statement is used to view its records.