Basic Steps to communicate with database

1. Establish connection to database
2. Create cursor for sending sql statements/query
3. Send sql statements
4. Read the result from cursor
5. Close connection

```
import mysql.connector
```

How to establish connection to mysql database?

connect() : this function establish connection to mysql database and return MySQLConnection object. Creating connection is nothing but creating session. All SQL statements get executed within session or connection.

 # write a program to establish connection to mysql database

```
import mysql.connector
def main():
cn=mysql.connector.connect(database="db1",user="root",password="root")
    print("connection established")

main()
```

Output:

```
connection established
```

How to send SQL statements/Queries to database?

cursor(): it is a method of MySQLConnection object. This method returns MySQLCursor object. Cursor is an object which contains result of executed query

cursor provides the following methods.

1. execute(Query)
2. executemany(list)

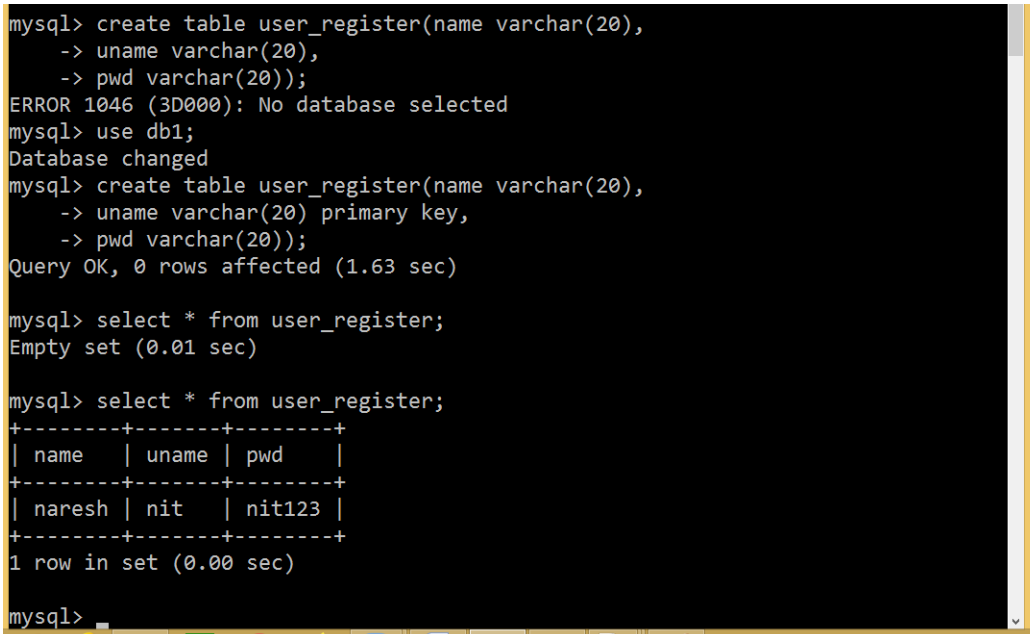
Example:

```
# write a program to register user (inserting user details into user_register table)
import mysql.connector
def main():
cn=mysql.connector.connect(database="db1",user="root",password="root")
```

```
    print("connection established")
    c=cn.cursor()
    print("cursor object is created")
    c.execute("insert into user_register values('naresh','nit','nit123')")
    print("command executed")
    cn.commit()
    cn.close()
main()
```

Output:

```
connection established
cursor object is created
command executed
```



```
mysql> create table user_register(name varchar(20),
    -> uname varchar(20),
    -> pwd varchar(20));
ERROR 1046 (3D000): No database selected
mysql> use db1;
Database changed
mysql> create table user_register(name varchar(20),
    -> uname varchar(20) primary key,
    -> pwd varchar(20));
Query OK, 0 rows affected (1.63 sec)

mysql> select * from user_register;
Empty set (0.01 sec)

mysql> select * from user_register;
+--------+-------+--------+
| name   | uname | pwd    |
+--------+-------+--------+
| naresh | nit   | nit123 |
+--------+-------+--------+
1 row in set (0.00 sec)

mysql>
```

 # write a program to register user (inserting user details into user_register table)

```
import mysql.connector
def main():
cn=mysql.connector.connect(database="db1",user="root",password="root")
    c=cn.cursor()
    while True:
        name=input("Name:")
        uname=input("UserName:")
        pwd=input("Password:")
```

```
        cmd=f"insert into user_register values('{name}','{uname}','{pwd}')"
        c.execute(cmd)
        ans=input("Register Another  User?(Yes/No)")
        if ans=="No":
            break
    cn.commit()
    cn.close()

main()
```

Output:

```
Name:ram
UserName:ram
Password:ram123
Register Another  User?(Yes/No)Yes
Name:kishore
UserName:k
Password:k123
Register Another  User?(Yes/No)No
```

Create the following table in mysql database

```
mysql> create table emp(empno int(5) primary key,
    -> ename varchar(15),
    -> job varchar(10),
    -> sal float(10,2));
Query OK, 0 rows affected, 2 warnings (1.04 sec)

mysql>
```

 # write a program to insert values into emp table

```
import mysql.connector as mysql
def main():
    cn=mysql.connect(database="db1",user="root",password="root")
    c=cn.cursor()
    empno=int(input("EmployeeNo: "))
    ename=input("EmployeeName: ")
    job=input("Job:")
    sal=float(input("Salary:"))
    cmd=f"insert into emp values({empno},'{ename}','{job}',{sal})"
    c.execute(cmd)
    rc=c.rowcount
    print(rc)
    cn.commit()
```

```
    cn.close()

main()
```

Output:

```
 EmployeeNo: 1
EmployeeName: naresh
Job:ceo
Salary:45000
1
```

SQL command with parameters or replacement fields

SQL command with parameters receive/replace values during runtime. This SQL statement is also called dynamic SQL statement.
This SQL statement is replacement fields and each field replaced with values. Each replacement field is defined with %s

 # write a program to insert values into emp table

```
import mysql.connector as mysql
def main():
    cn=mysql.connect(database="db1",user="root",password="root")
    c=cn.cursor()
    empno=int(input("EmployeeNo: "))
    ename=input("EmployeeName: ")
    job=input("Job:")
    sal=float(input("Salary:"))
    cmd="insert into emp values(%s,%s,%s,%s)"
    c.execute(cmd,params=[empno,ename,job,sal])
    rc=c.rowcount
    print(rc)
    cn.commit()
    cn.close()

main()
```

Output:

```
EmployeeNo: 2
EmployeeName: suresh
Job:SE
Salary:50000
```

1