In [1]:
```
# importing mysql.connector to use python+sql
import mysql.connector as sql
```

In [2]:
```
# establishing coonection betwwen python+sql
connection = sql.connect(host='localhost',user='root',password='......')
connection
```

Out[2]:
```
<mysql.connector.connection_cext.CMySQLConnection at 0x275546d76a0>
```

In [3]:
```
# creating cursor to hold result of query execution
cursor = connection.cursor()
```

In [4]:
```
# cursor.execute() method executes the command
cursor.execute("show databases")
for x in cursor:
    print(x)
```

```
('codespyder',)
('information_schema',)
('internship',)
('mysql',)
('performance_schema',)
('pysupportdb',)
('sakila',)
('sys',)
('world',)
```

In [5]:
```
# cursor.execute() method executes the command
cursor.execute("use codespyder")
```

In [6]:
```
# cursor.execute() method executes the command
cursor.execute("show tables")
```

In [7]:
```
# cursor.fetchall() method return all the rows(values)
cursor.fetchall()
```

Out[7]:
```
[('bonus',), ('title',), ('worker',)]
```

In [ ]:
```
# Creating table
# you have to provide every line in ""(quotes) while writing sql query
```

```
cursor.execute("CREATE TABLE Worker (WORKER_ID INT PRIMARY KEY,FIRST_NAME VARCHAR(20) NOT NULL,"
               "LAST_NAME VARCHAR(20) NOT NULL,SALARY INT,"
                "JOINING_DATE DATETIME,DEPARTMENT VARCHAR(10))")
# cursor.ecxecute("CREATE TABLE Worker (WORKER_ID INT PRIMARY KEY,FIRST_NAME VARCHAR(20) NOT NULL,LAST_NAME VARCHAR(20) NOT NULL,
```

In [ ]:
```
# Inserting values into table
# query for insert
query = "INSERT INTO Worker (WORKER_ID,FIRST_NAME,LAST_NAME,SALARY,JOINING_DATE,DEPARTMENT) VALUES (%s,%s,%s,%s,%s,%s)"
# values provided into list to insert
value = [(1,'Monika','Arora',100000,'2014-02-20 09:00:00','HR'),
         (2,'Niharika','Verma',80000,'2014-06-11 09:00:00','Admin'),
         (3,'Vishal','Singhal',300000,'2014-02-20 09:00:00','HR'),
         (4,'Amitabh','Singh',500000,'2014-02-20 09:00:00','Admin'),
         (5,'Vivek','Bhati',500000,'2014-06-11 09:00:00','Admin'),
         (6,'Vipul','Diwan',200000,'2014-06-11 09:00:00','Account'),
         (7,'Satish','Kumar',75000,'2014-01-20 09:00:00','Account'),
         (8,'Geetika','Chauhan',90000,'2014-04-11 09:00:00','Admin')]

# To enter multiple records into table
cursor.executemany(query,value)
```

In [ ]:
```
cursor.execute("CREATE TABLE Bonus(WORKER_REF_ID INT NOT NULL,BONUS_DATE DATETIME,BONUS_AMOUNT INT)")
```

In [ ]:
```
query = "INSERT INTO Bonus(WORKER_REF_ID,BONUS_DATE,BONUS_AMOUNT) VALUES (%s,%s,%s)"
value = [(1,'2016-02-20 00:00:00',5000),
         (2,'2016-06-11 00:00:00',3000),
         (3,'2016-02-20 00:00:00',4000),
         (1,'2016-02-20 00:00:00',4500),
         (2,'2016-06-11 00:00:00',3500)]

cursor.executemany(query,value)
```

In [ ]:
```
cursor.execute("CREATE TABLE Title (WORKER_REF_ID INT NOT NULL,WORKER_TITLE VARCHAR(20),AFFECTED_FROM DATETIME)")
```

In [ ]:
```
query = "INSERT INTO Title (WORKER_REF_ID,WORKER_TITLE,AFFECTED_FROM) VALUES (%s,%s,%s)"
value = [(1,'Manager','2016-02-20 00:00:00'),
         (2,'Executive','2016-06-11 00:00:00'),
         (8,'Executive','2016-06-11 00:00:00'),
         (5,'Manager','2016-06-11 00:00:00'),
         (4,'Asst. Manager','2016-06-11 00:00:00'),
         (7,'Executive','2016-06-11 00:00:00'),
         (6,'Lead','2016-06-11 00:00:00'),
```

```
        (3,'Lead','2016-06-11 00:00:00')]

cursor.executemany(query,value)
```

In [8]:
```
# execute the command
# AS is used to give alias name
cursor.execute("SELECT FIRST_NAME AS 'WORKER_NAME' FROM Worker")
```

In [9]:
```
# return all the values executed by command
cursor.fetchall()
```

Out[9]:
```
[('Monika',),
 ('Niharika',),
 ('Vishal',),
 ('Amitabh',),
 ('Vivek',),
 ('Vipul',),
 ('Satish',),
 ('Geetika',)]
```

In [10]:
```
# upper() used to typecast in uppercase
cursor.execute("SELECT upper(FIRST_NAME) FROM Worker")
cursor.fetchall()
```

Out[10]:
```
[('MONIKA',),
 ('NIHARIKA',),
 ('VISHAL',),
 ('AMITABH',),
 ('VIVEK',),
 ('VIPUL',),
 ('SATISH',),
 ('GEETIKA',)]
```

In [11]:
```
# distict is used find unique values
cursor.execute("SELECT DISTINCT DEPARTMENT FROM Worker")
cursor.fetchall()
```

Out[11]:
```
[('HR',), ('Admin',), ('Account',)]
```

In [12]:
```
# substring() is used to find substring
# substring(string_name,start position,length of substring)
cursor.execute("SELECT substring(FIRST_NAME,1,3) FROM Worker")
cursor.fetchall()
```

Out[12]:
```
[('Mon',),
 ('Nih',),
 ('Vis',),
 ('Ami',),
 ('Viv',),
 ('Vip',),
 ('Sat',),
 ('Gee',)]
```

In [13]:
```
# length() is used to find length of string
cursor.execute("SELECT DISTINCT DEPARTMENT, length(DEPARTMENT)  FROM Worker")
cursor.fetchall()
```

Out[13]:
```
[('HR', 2), ('Admin', 5), ('Account', 7)]
```

In [14]:
```
# concat() used to concat values of column
cursor.execute("SELECT concat(FIRST_NAME,' ',LAST_NAME) AS COMPLETE_NAME FROM Worker")
# *, gives whole table
# cursor.executabse("SELECT *, concat(FIRST_NAME,' ',LAST_NAME) AS COMPLETE_NAME FROM Worker")
cursor.fetchall()
```

Out[14]:
```
[('Monika Arora',),
 ('Niharika Verma',),
 ('Vishal Singhal',),
 ('Amitabh Singh',),
 ('Vivek Bhati',),
 ('Vipul Diwan',),
 ('Satish Kumar',),
 ('Geetika Chauhan',)]
```

In [15]:
```
# order by clause use for sorting data
# order by is by default in ascending order
cursor.execute("SELECT * FROM Worker ORDER BY FIRST_NAME")
cursor.fetchall()
```

Out[15]:
```
[(4,
  'Amitabh',
  'Singh',
  500000,
  datetime.datetime(2014, 2, 20, 9, 0),
  'Admin'),
 (8,
  'Geetika',
  'Chauhan',
  90000,
  datetime.datetime(2014, 4, 11, 9, 0),
  'Admin'),
 (1, 'Monika', 'Arora', 100000, datetime.datetime(2014, 2, 20, 9, 0), 'HR'),
 (2,
  'Niharika',
  'Verma',
  80000,
  datetime.datetime(2014, 6, 11, 9, 0),
  'Admin'),
 (7,
  'Satish',
  'Kumar',
  75000,
  datetime.datetime(2014, 1, 20, 9, 0),
  'Account'),
 (6,
  'Vipul',
  'Diwan',
  200000,
  datetime.datetime(2014, 6, 11, 9, 0),
  'Account'),
 (3, 'Vishal', 'Singhal', 300000, datetime.datetime(2014, 2, 20, 9, 0), 'HR'),
 (5, 'Vivek', 'Bhati', 500000, datetime.datetime(2014, 6, 11, 9, 0), 'Admin')]
```

In [16]:
```
# order by is use for sorting data
# DESC is use to sort data in descending order
cursor.execute("SELECT * FROM Worker ORDER BY FIRST_NAME ASC, DEPARTMENT DESC")
cursor.fetchall()
```

Out[16]:
```
[(4,
  'Amitabh',
  'Singh',
  500000,
  datetime.datetime(2014, 2, 20, 9, 0),
  'Admin'),
 (8,
  'Geetika',
  'Chauhan',
  90000,
  datetime.datetime(2014, 4, 11, 9, 0),
  'Admin'),
 (1, 'Monika', 'Arora', 100000, datetime.datetime(2014, 2, 20, 9, 0), 'HR'),
 (2,
  'Niharika',
  'Verma',
  80000,
  datetime.datetime(2014, 6, 11, 9, 0),
  'Admin'),
 (7,
  'Satish',
  'Kumar',
  75000,
  datetime.datetime(2014, 1, 20, 9, 0),
  'Account'),
 (6,
  'Vipul',
  'Diwan',
  200000,
  datetime.datetime(2014, 6, 11, 9, 0),
  'Account'),
 (3, 'Vishal', 'Singhal', 300000, datetime.datetime(2014, 2, 20, 9, 0), 'HR'),
 (5, 'Vivek', 'Bhati', 500000, datetime.datetime(2014, 6, 11, 9, 0), 'Admin')]
```

In [17]:
```
# WHERE clause is use to select data (filter data)
cursor.execute("SELECT * FROM Worker WHERE FIRST_NAME='Vipul' OR FIRST_NAME='Satish'")
cursor.fetchall()
```

Out[17]:
```
[(6,
  'Vipul',
  'Diwan',
  200000,
  datetime.datetime(2014, 6, 11, 9, 0),
  'Account'),
 (7,
  'Satish',
  'Kumar',
  75000,
  datetime.datetime(2014, 1, 20, 9, 0),
  'Account')]
```

In [18]:
```
cursor.execute("SELECT * FROM Worker WHERE DEPARTMENT='Admin'")
cursor.fetchall()
```

Out[18]:
```
[(2,
  'Niharika',
  'Verma',
  80000,
  datetime.datetime(2014, 6, 11, 9, 0),
  'Admin'),
 (4,
  'Amitabh',
  'Singh',
  500000,
  datetime.datetime(2014, 2, 20, 9, 0),
  'Admin'),
 (5, 'Vivek', 'Bhati', 500000, datetime.datetime(2014, 6, 11, 9, 0), 'Admin'),
 (8,
  'Geetika',
  'Chauhan',
  90000,
  datetime.datetime(2014, 4, 11, 9, 0),
  'Admin')]
```

In [19]:
```
# LIKE clause is use to compare 'a%' - gives string starting with a
cursor.execute("SELECT * FROM Worker WHERE FIRST_NAME LIKE 'a%'")
cursor.fetchall()
```

Out[19]:
```
[(4,
  'Amitabh',
  'Singh',
  500000,
  datetime.datetime(2014, 2, 20, 9, 0),
  'Admin')]
```

In [20]:
```
# LIKE clause is use to compare '%h' - gives string ending with h
cursor.execute("SELECT * FROM Worker WHERE FIRST_NAME LIKE '%h' AND length(FIRST_NAME)=6")
cursor.fetchall()
```

Out[20]:
```
[(7,
  'Satish',
  'Kumar',
  75000,
  datetime.datetime(2014, 1, 20, 9, 0),
  'Account')]
```

In [21]:
```
cursor.execute("SELECT * FROM Worker WHERE SALARY>=100000 AND SALARY<=500000")
cursor.fetchall()
```

Out[21]:
```
[(1, 'Monika', 'Arora', 100000, datetime.datetime(2014, 2, 20, 9, 0), 'HR'),
 (3, 'Vishal', 'Singhal', 300000, datetime.datetime(2014, 2, 20, 9, 0), 'HR'),
 (4,
  'Amitabh',
  'Singh',
  500000,
  datetime.datetime(2014, 2, 20, 9, 0),
  'Admin'),
 (5, 'Vivek', 'Bhati', 500000, datetime.datetime(2014, 6, 11, 9, 0), 'Admin'),
 (6,
  'Vipul',
  'Diwan',
  200000,
  datetime.datetime(2014, 6, 11, 9, 0),
  'Account')]
```

In [22]:
```
cursor.execute("SELECT FIRST_NAME,LAST_NAME FROM Worker WHERE SALARY<=100000 AND SALARY>=50000")
cursor.fetchall()
```

Out[22]:
```
[('Monika', 'Arora'),
 ('Niharika', 'Verma'),
 ('Satish', 'Kumar'),
 ('Geetika', 'Chauhan')]
```

In [23]:
```
# GROUP BY clause is use to grouped data
cursor.execute("SELECT DEPARTMENT,count(WORKER_ID) FROM Worker GROUP BY DEPARTMENT ORDER BY DEPARTMENT DESC")
cursor.fetchall()
```

Out[23]:
```
[('HR', 2), ('Admin', 4), ('Account', 2)]
```