In [1]:
```
pip install mysql-connector-python
```

```
Requirement already satisfied: mysql-connector-python in c:\users\kamis\anaconda3\lib\site-packages (9.0.0)
Note: you may need to restart the kernel to use updated packages.
```

In [59]:
```
import mysql.connector
connection=mysql.connector.connect(
    host="localhost",
    user="root",
    password="root",
    database="company")
mycursor=connection.cursor()
```

In [9]:
```
if connection.is_connected():
    print("is connected")
```

```
is connected
```

In [11]:
```
mycursor.execute("show databases")
for x in cursor:
    print(x)
```

```
('company',)
('information_schema',)
('mani_marketing',)
('mysql',)
('performance_schema',)
('studentmanagement',)
('sys',)
('world',)
```

In [61]:
```
mycursor.execute("show tables")
myrsesult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
('company_data',)
('departments',)
('employees',)
```

In [ ]:
```
#create a table
mycursor.execute('''CREATE TABLE company_data (
    id INT PRIMARY KEY,
    company_name VARCHAR(255),
    location VARCHAR(255),
    industry VARCHAR(100),
    number_of_employees INT,
    revenue DECIMAL(15, 2))''')
for x in cursor:
    print(x)
```

In [25]:
```
sql_insert="insert into company_data(id,company_name,location,industry,number_of_employees,revenue)values(%s,%s
values=[ (61, 'Tech Innovate', 'San Francisco', 'Technology', 450, 16000000.00),
    (62, 'Fresh Farms', 'Fresno', 'Agriculture', 200, 2500000.00),
    (63, 'Wellness Center', 'Houston', 'Healthcare', 320, 8500000.00),
    (64, 'Safe Invest', 'New York', 'Finance', 400, 13500000.00),
    (65, 'Knowledge Tree', 'Boston', 'Education', 90, 1900000.00),
    (66, 'QuickMart', 'Dallas', 'Retail', 700, 23000000.00),
    (67, 'BioFuel Corp', 'Phoenix', 'Energy', 500, 17000000.00),
    (68, 'Fast Motors', 'Detroit', 'Automotive', 450, 21000000.00),
    (69, 'Gourmet Foods', 'Miami', 'Food & Beverage', 250, 3500000.00),
    (70, 'Sky Travels', 'New York', 'Travel', 55, 1200000.00),
    (71, 'Tech Solutions', 'San Jose', 'Technology', 600, 24000000.00),
    (72, 'Harvest More', 'Little Rock', 'Agriculture', 170, 1400000.00),
    (73, 'HealthLine', 'Austin', 'Healthcare', 290, 7800000.00),
    (74, 'Capital Trust', 'Charlotte', 'Finance', 350, 13000000.00),
    (75, 'Bright Minds', 'Minneapolis', 'Education', 130, 2300000.00),
    (76, 'Market Plaza', 'Columbus', 'Retail', 850, 25000000.00),
    (77, 'Green Energy', 'Portland', 'Energy', 380, 18000000.00),
    (78, 'MotorWorld', 'Indianapolis', 'Automotive', 480, 20000000.00),
    (79, 'Taste Buds', 'San Diego', 'Food & Beverage', 210, 4100000.00),
    (80, 'Dream Vacations', 'Orlando', 'Travel', 35, 900000.00)
]
mycursor.executemany(sql_insert,values)
connection.commit()
```

In [43]:
```
mycursor.execute("select*from company_data")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
(1, 'Tech Corp', 'San Francisco', 'Technology', 500, Decimal('15000000.00'))
(2, 'FinServ', 'New York', 'Finance', 300, Decimal('8000000.00'))
(3, 'HealthWorks', 'Los Angeles', 'Healthcare', 250, Decimal('5000000.00'))
(4, 'EduLearn', 'Boston', 'Education', 100, Decimal('2000000.00'))
(5, 'Retail King', 'Chicago', 'Retail', 1000, Decimal('22000000.00'))
(6, 'AgriGrowth', 'Des Moines', 'Agriculture', 150, Decimal('1200000.00'))
(7, 'EnergyMax', 'Houston', 'Energy', 400, Decimal('18000000.00'))
(8, 'AutoMotive', 'Detroit', 'Automotive', 600, Decimal('25000000.00'))
(9, 'Foodies', 'New Orleans', 'Food & Beverage', 200, Decimal('5000000.00'))
(10, 'TravelEase', 'Orlando', 'Travel', 75, Decimal('3000000.00'))
(11, 'Tech Hub', 'San Jose', 'Technology', 800, Decimal('32000000.00'))
(12, 'Green Fields', 'Fresno', 'Agriculture', 90, Decimal('900000.00'))
(13, 'MediCare', 'Houston', 'Healthcare', 350, Decimal('10000000.00'))
(14, 'FinWorld', 'Miami', 'Finance', 450, Decimal('15000000.00'))
(15, 'Bright Edu', 'Seattle', 'Education', 120, Decimal('1800000.00'))
(16, 'HomeWare', 'Dallas', 'Retail', 550, Decimal('8000000.00'))
(17, 'Solar Energy', 'Phoenix', 'Energy', 300, Decimal('12000000.00'))
(18, 'MotorWorks', 'St. Louis', 'Automotive', 350, Decimal('18000000.00'))
(19, 'Bakers Delight', 'San Diego', 'Food & Beverage', 150, Decimal('2000000.00'))
(20, 'Adventure Co', 'Denver', 'Travel', 60, Decimal('1500000.00'))
(21, 'CloudNet', 'San Francisco', 'Technology', 700, Decimal('26000000.00'))
(22, 'GrainCorp', 'Omaha', 'Agriculture', 130, Decimal('1400000.00'))
(23, 'HealNow', 'Austin', 'Healthcare', 300, Decimal('8000000.00'))
(24, 'Invest Corp', 'Charlotte', 'Finance', 250, Decimal('12000000.00'))
(25, 'Learn Academy', 'Philadelphia', 'Education', 180, Decimal('2200000.00'))
(26, 'Mega Mart', 'Atlanta', 'Retail', 750, Decimal('20000000.00'))
(27, 'WindPower', 'Portland', 'Energy', 350, Decimal('16000000.00'))
(28, 'DriveSafe', 'Nashville', 'Automotive', 450, Decimal('22000000.00'))
(29, 'Cafe Treats', 'Austin', 'Food & Beverage', 100, Decimal('3000000.00'))
(30, 'Global Tours', 'New York', 'Travel', 50, Decimal('2000000.00'))
(31, 'InfoTech', 'San Jose', 'Technology', 600, Decimal('24000000.00'))
(32, 'Farm Fresh', 'Little Rock', 'Agriculture', 160, Decimal('1300000.00'))
(33, 'HealthFirst', 'Los Angeles', 'Healthcare', 400, Decimal('12000000.00'))
(34, 'FinanceNow', 'Dallas', 'Finance', 500, Decimal('17000000.00'))
(35, 'EduWave', 'Minneapolis', 'Education', 140, Decimal('2500000.00'))
(36, 'ShopSmart', 'Columbus', 'Retail', 900, Decimal('23000000.00'))
(37, 'PowerGrid', 'Oklahoma City', 'Energy', 380, Decimal('19000000.00'))
(38, 'AutoWorld', 'Indianapolis', 'Automotive', 520, Decimal('24000000.00'))
(39, 'Sweet Bakery', 'Miami', 'Food & Beverage', 180, Decimal('3500000.00'))
(40, 'JetSet', 'Seattle', 'Travel', 40, Decimal('1000000.00'))
(41, 'ByteNet', 'San Francisco', 'Technology', 900, Decimal('35000000.00'))
(42, 'Harvest', 'Boise', 'Agriculture', 110, Decimal('1100000.00'))
(43, 'LifeCare', 'Boston', 'Healthcare', 360, Decimal('9000000.00'))
(44, 'TradeWise', 'Denver', 'Finance', 320, Decimal('11000000.00'))
(45, 'EduBright', 'Portland', 'Education', 160, Decimal('2000000.00'))
(46, 'RetailMart', 'Charlotte', 'Retail', 800, Decimal('21000000.00'))
(47, 'SolarWave', 'Las Vegas', 'Energy', 260, Decimal('13000000.00'))
(48, 'CarZone', 'Memphis', 'Automotive', 580, Decimal('25000000.00'))
(49, 'Gourmet Cafe', 'San Francisco', 'Food & Beverage', 130, Decimal('4000000.00'))
(50, 'Wanderlust', 'Chicago', 'Travel', 65, Decimal('1800000.00'))
(51, 'SoftTech', 'San Jose', 'Technology', 700, Decimal('29000000.00'))
(52, 'Green Thumb', 'Salt Lake City', 'Agriculture', 190, Decimal('1500000.00'))
(53, 'MedCentral', 'Houston', 'Healthcare', 500, Decimal('15000000.00'))
(54, 'InvestSmart', 'Miami', 'Finance', 600, Decimal('20000000.00'))
(55, 'LearnEasy', 'Phoenix', 'Education', 220, Decimal('2800000.00'))
(56, 'ShopVille', 'New York', 'Retail', 850, Decimal('24000000.00'))
(57, 'EnergyStream', 'Austin', 'Energy', 420, Decimal('18000000.00'))
(58, 'AutoPlus', 'Cleveland', 'Automotive', 640, Decimal('28000000.00'))
(59, 'Deli Delight', 'Boston', 'Food & Beverage', 210, Decimal('4500000.00'))
(60, 'Explore Co', 'Los Angeles', 'Travel', 55, Decimal('1200000.00'))
(61, 'Tech Innovate', 'San Francisco', 'Technology', 450, Decimal('16000000.00'))
(62, 'Fresh Farms', 'Fresno', 'Agriculture', 200, Decimal('2500000.00'))
(63, 'Wellness Center', 'Houston', 'Healthcare', 320, Decimal('8500000.00'))
(64, 'Safe Invest', 'New York', 'Finance', 400, Decimal('13500000.00'))
(65, 'Knowledge Tree', 'Boston', 'Education', 90, Decimal('1900000.00'))
(66, 'QuickMart', 'Dallas', 'Retail', 700, Decimal('23000000.00'))
(67, 'BioFuel Corp', 'Phoenix', 'Energy', 500, Decimal('17000000.00'))
(68, 'Fast Motors', 'Detroit', 'Automotive', 450, Decimal('21000000.00'))
(69, 'Gourmet Foods', 'Miami', 'Food & Beverage', 250, Decimal('3500000.00'))
(70, 'Sky Travels', 'New York', 'Travel', 55, Decimal('1200000.00'))
(71, 'Tech Solutions', 'San Jose', 'Technology', 600, Decimal('24000000.00'))
(72, 'Harvest More', 'Little Rock', 'Agriculture', 170, Decimal('1400000.00'))
(73, 'HealthLine', 'Austin', 'Healthcare', 290, Decimal('7800000.00'))
(74, 'Capital Trust', 'Charlotte', 'Finance', 350, Decimal('13000000.00'))
(75, 'Bright Minds', 'Minneapolis', 'Education', 130, Decimal('2300000.00'))
(76, 'Market Plaza', 'Columbus', 'Retail', 850, Decimal('25000000.00'))
(77, 'Green Energy', 'Portland', 'Energy', 380, Decimal('18000000.00'))
(78, 'MotorWorld', 'Indianapolis', 'Automotive', 480, Decimal('20000000.00'))
(79, 'Taste Buds', 'San Diego', 'Food & Beverage', 210, Decimal('4100000.00'))
(80, 'Dream Vacations', 'Orlando', 'Travel', 35, Decimal('900000.00'))
```

In [45]:
```
mycursor.execute("select company_name,location from company_data")
myresult=mycursor.fetchall()
```

```
for x in myresult:
    print(x)
```

```
('Tech Corp', 'San Francisco')
('FinServ', 'New York')
('HealthWorks', 'Los Angeles')
('EduLearn', 'Boston')
('Retail King', 'Chicago')
('AgriGrowth', 'Des Moines')
('EnergyMax', 'Houston')
('AutoMotive', 'Detroit')
('Foodies', 'New Orleans')
('TravelEase', 'Orlando')
('Tech Hub', 'San Jose')
('Green Fields', 'Fresno')
('MediCare', 'Houston')
('FinWorld', 'Miami')
('Bright Edu', 'Seattle')
('HomeWare', 'Dallas')
('Solar Energy', 'Phoenix')
('MotorWorks', 'St. Louis')
('Bakers Delight', 'San Diego')
('Adventure Co', 'Denver')
('CloudNet', 'San Francisco')
('GrainCorp', 'Omaha')
('HealNow', 'Austin')
('Invest Corp', 'Charlotte')
('Learn Academy', 'Philadelphia')
('Mega Mart', 'Atlanta')
('WindPower', 'Portland')
('DriveSafe', 'Nashville')
('Cafe Treats', 'Austin')
('Global Tours', 'New York')
('InfoTech', 'San Jose')
('Farm Fresh', 'Little Rock')
('HealthFirst', 'Los Angeles')
('FinanceNow', 'Dallas')
('EduWave', 'Minneapolis')
('ShopSmart', 'Columbus')
('PowerGrid', 'Oklahoma City')
('AutoWorld', 'Indianapolis')
('Sweet Bakery', 'Miami')
('JetSet', 'Seattle')
('ByteNet', 'San Francisco')
('Harvest', 'Boise')
('LifeCare', 'Boston')
('TradeWise', 'Denver')
('EduBright', 'Portland')
('RetailMart', 'Charlotte')
('SolarWave', 'Las Vegas')
('CarZone', 'Memphis')
('Gourmet Cafe', 'San Francisco')
('Wanderlust', 'Chicago')
('SoftTech', 'San Jose')
('Green Thumb', 'Salt Lake City')
('MedCentral', 'Houston')
('InvestSmart', 'Miami')
('LearnEasy', 'Phoenix')
('ShopVille', 'New York')
('EnergyStream', 'Austin')
('AutoPlus', 'Cleveland')
('Deli Delight', 'Boston')
('Explore Co', 'Los Angeles')
('Tech Innovate', 'San Francisco')
('Fresh Farms', 'Fresno')
('Wellness Center', 'Houston')
('Safe Invest', 'New York')
('Knowledge Tree', 'Boston')
('QuickMart', 'Dallas')
('BioFuel Corp', 'Phoenix')
('Fast Motors', 'Detroit')
('Gourmet Foods', 'Miami')
('Sky Travels', 'New York')
('Tech Solutions', 'San Jose')
('Harvest More', 'Little Rock')
('HealthLine', 'Austin')
('Capital Trust', 'Charlotte')
('Bright Minds', 'Minneapolis')
('Market Plaza', 'Columbus')
('Green Energy', 'Portland')
('MotorWorld', 'Indianapolis')
('Taste Buds', 'San Diego')
('Dream Vacations', 'Orlando')
```

In [71]:
```
# if you execute single row like you use fetchone() function.
mycursor.execute("select*from company_data")
myresult=mycursor.fetchone()
print(myresult)
```

```
(1, 'Tech Corp', 'San Francisco', 'Technology', 500, Decimal('15000000.00'))
```

In [45]:
```
import mysql.connector
connection=mysql.connector.connect(
    host="localhost",
    user="root",
    password="root",
    database="company")
mycursor=connection.cursor()
```

pyhton mysql where

->selecting records from a table,you can fliter the selection by using "where" statement.

mysql querys like ->select*from company_data where id="13"-only 13 id data show ->select*from company_data where company_name='fast motors"-that company name data will show it. ->you give anyone where statement and specific column name its show that coloumn data.

In [11]:
```
mycursor.execute("select*from company_data where location='new york' ")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
(2, 'FinServ', 'New York', 'Finance', 300, Decimal('8000000.00'))
(30, 'Global Tours', 'New York', 'Travel', 50, Decimal('2000000.00'))
(56, 'ShopVille', 'New York', 'Retail', 850, Decimal('24000000.00'))
(64, 'Safe Invest', 'New York', 'Finance', 400, Decimal('13500000.00'))
(70, 'Sky Travels', 'New York', 'Travel', 55, Decimal('1200000.00'))
```

In [ ]:
```
#wildcard characters
you can also select the records that start,ends with a given letter.
->using % to represent wildcard characters.
```

In [13]:
```
# example: location names and i given starting letter "N" show only all location letter N names.

mycursor.execute("select*from company_data where location like '%N' ")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
(4, 'EduLearn', 'Boston', 'Education', 100, Decimal('2000000.00'))
(7, 'EnergyMax', 'Houston', 'Energy', 400, Decimal('18000000.00'))
(13, 'MediCare', 'Houston', 'Healthcare', 350, Decimal('10000000.00'))
(23, 'HealNow', 'Austin', 'Healthcare', 300, Decimal('8000000.00'))
(29, 'Cafe Treats', 'Austin', 'Food & Beverage', 100, Decimal('3000000.00'))
(43, 'LifeCare', 'Boston', 'Healthcare', 360, Decimal('9000000.00'))
(53, 'MedCentral', 'Houston', 'Healthcare', 500, Decimal('15000000.00'))
(57, 'EnergyStream', 'Austin', 'Energy', 420, Decimal('18000000.00'))
(59, 'Deli Delight', 'Boston', 'Food & Beverage', 210, Decimal('4500000.00'))
(63, 'Wellness Center', 'Houston', 'Healthcare', 320, Decimal('8500000.00'))
(65, 'Knowledge Tree', 'Boston', 'Education', 90, Decimal('1900000.00'))
(73, 'HealthLine', 'Austin', 'Healthcare', 290, Decimal('7800000.00'))
```

prevent sql injection

->this is to prevent sql injections,which is a common web hacking tech to destroy or misuse your database. -> mysql.connector module has method to escape query values.

In [23]:
```
mysql="select*from company_data where location=%s "
loc=("New york",)
mycursor.execute(mysql,loc)
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
(2, 'FinServ', 'New York', 'Finance', 300, Decimal('8000000.00'))
(30, 'Global Tours', 'New York', 'Travel', 50, Decimal('2000000.00'))
(56, 'ShopVille', 'New York', 'Retail', 850, Decimal('24000000.00'))
(64, 'Safe Invest', 'New York', 'Finance', 400, Decimal('13500000.00'))
(70, 'Sky Travels', 'New York', 'Travel', 55, Decimal('1200000.00'))
```

In [ ]:
```
python mysql orderby:
```

```
->simply say short the result.
-> we use ascending "asc"keyword or descending "desc" keyword.
```

In [29]:

```
#example: show data order by company names like alphabetic order

mycursor.execute("select*from company_data order by company_name")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
(20, 'Adventure Co', 'Denver', 'Travel', 60, Decimal('1500000.00'))
(6, 'AgriGrowth', 'Des Moines', 'Agriculture', 150, Decimal('1200000.00'))
(8, 'AutoMotive', 'Detroit', 'Automotive', 600, Decimal('25000000.00'))
(58, 'AutoPlus', 'Cleveland', 'Automotive', 640, Decimal('28000000.00'))
(38, 'AutoWorld', 'Indianapolis', 'Automotive', 520, Decimal('24000000.00'))
(19, 'Bakers Delight', 'San Diego', 'Food & Beverage', 150, Decimal('2000000.00'))
(67, 'BioFuel Corp', 'Phoenix', 'Energy', 500, Decimal('17000000.00'))
(15, 'Bright Edu', 'Seattle', 'Education', 120, Decimal('1800000.00'))
(75, 'Bright Minds', 'Minneapolis', 'Education', 130, Decimal('2300000.00'))
(41, 'ByteNet', 'San Francisco', 'Technology', 900, Decimal('35000000.00'))
(29, 'Cafe Treats', 'Austin', 'Food & Beverage', 100, Decimal('3000000.00'))
(74, 'Capital Trust', 'Charlotte', 'Finance', 350, Decimal('13000000.00'))
(48, 'CarZone', 'Memphis', 'Automotive', 580, Decimal('25000000.00'))
(21, 'CloudNet', 'San Francisco', 'Technology', 700, Decimal('26000000.00'))
(59, 'Deli Delight', 'Boston', 'Food & Beverage', 210, Decimal('4500000.00'))
(80, 'Dream Vacations', 'Orlando', 'Travel', 35, Decimal('900000.00'))
(28, 'DriveSafe', 'Nashville', 'Automotive', 450, Decimal('22000000.00'))
(45, 'EduBright', 'Portland', 'Education', 160, Decimal('2000000.00'))
(4, 'EduLearn', 'Boston', 'Education', 100, Decimal('2000000.00'))
(35, 'EduWave', 'Minneapolis', 'Education', 140, Decimal('2500000.00'))
(7, 'EnergyMax', 'Houston', 'Energy', 400, Decimal('18000000.00'))
(57, 'EnergyStream', 'Austin', 'Energy', 420, Decimal('18000000.00'))
(60, 'Explore Co', 'Los Angeles', 'Travel', 55, Decimal('1200000.00'))
(32, 'Farm Fresh', 'Little Rock', 'Agriculture', 160, Decimal('1300000.00'))
(68, 'Fast Motors', 'Detroit', 'Automotive', 450, Decimal('21000000.00'))
(34, 'FinanceNow', 'Dallas', 'Finance', 500, Decimal('17000000.00'))
(2, 'FinServ', 'New York', 'Finance', 300, Decimal('8000000.00'))
(14, 'FinWorld', 'Miami', 'Finance', 450, Decimal('15000000.00'))
(9, 'Foodies', 'New Orleans', 'Food & Beverage', 200, Decimal('5000000.00'))
(62, 'Fresh Farms', 'Fresno', 'Agriculture', 200, Decimal('2500000.00'))
(30, 'Global Tours', 'New York', 'Travel', 50, Decimal('2000000.00'))
(49, 'Gourmet Cafe', 'San Francisco', 'Food & Beverage', 130, Decimal('4000000.00'))
(69, 'Gourmet Foods', 'Miami', 'Food & Beverage', 250, Decimal('3500000.00'))
(22, 'GrainCorp', 'Omaha', 'Agriculture', 130, Decimal('1400000.00'))
(77, 'Green Energy', 'Portland', 'Energy', 380, Decimal('18000000.00'))
(12, 'Green Fields', 'Fresno', 'Agriculture', 90, Decimal('900000.00'))
(52, 'Green Thumb', 'Salt Lake City', 'Agriculture', 190, Decimal('1500000.00'))
(42, 'Harvest', 'Boise', 'Agriculture', 110, Decimal('1100000.00'))
(72, 'Harvest More', 'Little Rock', 'Agriculture', 170, Decimal('1400000.00'))
(23, 'HealNow', 'Austin', 'Healthcare', 300, Decimal('8000000.00'))
(33, 'HealthFirst', 'Los Angeles', 'Healthcare', 400, Decimal('12000000.00'))
(73, 'HealthLine', 'Austin', 'Healthcare', 290, Decimal('7800000.00'))
(3, 'HealthWorks', 'Los Angeles', 'Healthcare', 250, Decimal('5000000.00'))
(16, 'HomeWare', 'Dallas', 'Retail', 550, Decimal('8000000.00'))
(31, 'InfoTech', 'San Jose', 'Technology', 600, Decimal('24000000.00'))
(24, 'Invest Corp', 'Charlotte', 'Finance', 250, Decimal('12000000.00'))
(54, 'InvestSmart', 'Miami', 'Finance', 600, Decimal('20000000.00'))
(40, 'JetSet', 'Seattle', 'Travel', 40, Decimal('1000000.00'))
(65, 'Knowledge Tree', 'Boston', 'Education', 90, Decimal('1900000.00'))
(25, 'Learn Academy', 'Philadelphia', 'Education', 180, Decimal('2200000.00'))
(55, 'LearnEasy', 'Phoenix', 'Education', 220, Decimal('2800000.00'))
(43, 'LifeCare', 'Boston', 'Healthcare', 360, Decimal('9000000.00'))
(76, 'Market Plaza', 'Columbus', 'Retail', 850, Decimal('25000000.00'))
(53, 'MedCentral', 'Houston', 'Healthcare', 500, Decimal('15000000.00'))
(13, 'MediCare', 'Houston', 'Healthcare', 350, Decimal('10000000.00'))
(26, 'Mega Mart', 'Atlanta', 'Retail', 750, Decimal('20000000.00'))
(18, 'MotorWorks', 'St. Louis', 'Automotive', 350, Decimal('18000000.00'))
(78, 'MotorWorld', 'Indianapolis', 'Automotive', 480, Decimal('20000000.00'))
(37, 'PowerGrid', 'Oklahoma City', 'Energy', 380, Decimal('19000000.00'))
(66, 'QuickMart', 'Dallas', 'Retail', 700, Decimal('23000000.00'))
(5, 'Retail King', 'Chicago', 'Retail', 1000, Decimal('22000000.00'))
(46, 'RetailMart', 'Charlotte', 'Retail', 800, Decimal('21000000.00'))
(64, 'Safe Invest', 'New York', 'Finance', 400, Decimal('13500000.00'))
(36, 'ShopSmart', 'Columbus', 'Retail', 900, Decimal('23000000.00'))
(56, 'ShopVille', 'New York', 'Retail', 850, Decimal('24000000.00'))
(70, 'Sky Travels', 'New York', 'Travel', 55, Decimal('1200000.00'))
(51, 'SoftTech', 'San Jose', 'Technology', 700, Decimal('29000000.00'))
(17, 'Solar Energy', 'Phoenix', 'Energy', 300, Decimal('12000000.00'))
(47, 'SolarWave', 'Las Vegas', 'Energy', 260, Decimal('13000000.00'))
(39, 'Sweet Bakery', 'Miami', 'Food & Beverage', 180, Decimal('3500000.00'))
(79, 'Taste Buds', 'San Diego', 'Food & Beverage', 210, Decimal('4100000.00'))
(1, 'Tech Corp', 'San Francisco', 'Technology', 500, Decimal('15000000.00'))
(11, 'Tech Hub', 'San Jose', 'Technology', 800, Decimal('32000000.00'))
(61, 'Tech Innovate', 'San Francisco', 'Technology', 450, Decimal('16000000.00'))
(71, 'Tech Solutions', 'San Jose', 'Technology', 600, Decimal('24000000.00'))
(44, 'TradeWise', 'Denver', 'Finance', 320, Decimal('11000000.00'))
(10, 'TravelEase', 'Orlando', 'Travel', 75, Decimal('3000000.00'))
(50, 'Wanderlust', 'Chicago', 'Travel', 65, Decimal('1800000.00'))
(63, 'Wellness Center', 'Houston', 'Healthcare', 320, Decimal('8500000.00'))
(27, 'WindPower', 'Portland', 'Energy', 350, Decimal('16000000.00'))
```

In [31]:
```
#example: data showing revenue like descending order
```

```
mycursor.execute("select*from company_data order by revenue desc")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
(41, 'ByteNet', 'San Francisco', 'Technology', 900, Decimal('35000000.00'))
(11, 'Tech Hub', 'San Jose', 'Technology', 800, Decimal('32000000.00'))
(51, 'SoftTech', 'San Jose', 'Technology', 700, Decimal('29000000.00'))
(58, 'AutoPlus', 'Cleveland', 'Automotive', 640, Decimal('28000000.00'))
(21, 'CloudNet', 'San Francisco', 'Technology', 700, Decimal('26000000.00'))
(8, 'AutoMotive', 'Detroit', 'Automotive', 600, Decimal('25000000.00'))
(48, 'CarZone', 'Memphis', 'Automotive', 580, Decimal('25000000.00'))
(76, 'Market Plaza', 'Columbus', 'Retail', 850, Decimal('25000000.00'))
(56, 'ShopVille', 'New York', 'Retail', 850, Decimal('24000000.00'))
(71, 'Tech Solutions', 'San Jose', 'Technology', 600, Decimal('24000000.00'))
(38, 'AutoWorld', 'Indianapolis', 'Automotive', 520, Decimal('24000000.00'))
(31, 'InfoTech', 'San Jose', 'Technology', 600, Decimal('24000000.00'))
(36, 'ShopSmart', 'Columbus', 'Retail', 900, Decimal('23000000.00'))
(66, 'QuickMart', 'Dallas', 'Retail', 700, Decimal('23000000.00'))
(5, 'Retail King', 'Chicago', 'Retail', 1000, Decimal('22000000.00'))
(28, 'DriveSafe', 'Nashville', 'Automotive', 450, Decimal('22000000.00'))
(46, 'RetailMart', 'Charlotte', 'Retail', 800, Decimal('21000000.00'))
(68, 'Fast Motors', 'Detroit', 'Automotive', 450, Decimal('21000000.00'))
(78, 'MotorWorld', 'Indianapolis', 'Automotive', 480, Decimal('20000000.00'))
(26, 'Mega Mart', 'Atlanta', 'Retail', 750, Decimal('20000000.00'))
(54, 'InvestSmart', 'Miami', 'Finance', 600, Decimal('20000000.00'))
(37, 'PowerGrid', 'Oklahoma City', 'Energy', 380, Decimal('19000000.00'))
(7, 'EnergyMax', 'Houston', 'Energy', 400, Decimal('18000000.00'))
(57, 'EnergyStream', 'Austin', 'Energy', 420, Decimal('18000000.00'))
(18, 'MotorWorks', 'St. Louis', 'Automotive', 350, Decimal('18000000.00'))
(77, 'Green Energy', 'Portland', 'Energy', 380, Decimal('18000000.00'))
(34, 'FinanceNow', 'Dallas', 'Finance', 500, Decimal('17000000.00'))
(67, 'BioFuel Corp', 'Phoenix', 'Energy', 500, Decimal('17000000.00'))
(27, 'WindPower', 'Portland', 'Energy', 350, Decimal('16000000.00'))
(61, 'Tech Innovate', 'San Francisco', 'Technology', 450, Decimal('16000000.00'))
(53, 'MedCentral', 'Houston', 'Healthcare', 500, Decimal('15000000.00'))
(1, 'Tech Corp', 'San Francisco', 'Technology', 500, Decimal('15000000.00'))
(14, 'FinWorld', 'Miami', 'Finance', 450, Decimal('15000000.00'))
(64, 'Safe Invest', 'New York', 'Finance', 400, Decimal('13500000.00'))
(47, 'SolarWave', 'Las Vegas', 'Energy', 260, Decimal('13000000.00'))
(74, 'Capital Trust', 'Charlotte', 'Finance', 350, Decimal('13000000.00'))
(33, 'HealthFirst', 'Los Angeles', 'Healthcare', 400, Decimal('12000000.00'))
(24, 'Invest Corp', 'Charlotte', 'Finance', 250, Decimal('12000000.00'))
(17, 'Solar Energy', 'Phoenix', 'Energy', 300, Decimal('12000000.00'))
(44, 'TradeWise', 'Denver', 'Finance', 320, Decimal('11000000.00'))
(13, 'MediCare', 'Houston', 'Healthcare', 350, Decimal('10000000.00'))
(43, 'LifeCare', 'Boston', 'Healthcare', 360, Decimal('9000000.00'))
(63, 'Wellness Center', 'Houston', 'Healthcare', 320, Decimal('8500000.00'))
(23, 'HealNow', 'Austin', 'Healthcare', 300, Decimal('8000000.00'))
(2, 'FinServ', 'New York', 'Finance', 300, Decimal('8000000.00'))
(16, 'HomeWare', 'Dallas', 'Retail', 550, Decimal('8000000.00'))
(73, 'HealthLine', 'Austin', 'Healthcare', 290, Decimal('7800000.00'))
(9, 'Foodies', 'New Orleans', 'Food & Beverage', 200, Decimal('5000000.00'))
(3, 'HealthWorks', 'Los Angeles', 'Healthcare', 250, Decimal('5000000.00'))
(59, 'Deli Delight', 'Boston', 'Food & Beverage', 210, Decimal('4500000.00'))
(79, 'Taste Buds', 'San Diego', 'Food & Beverage', 210, Decimal('4100000.00'))
(49, 'Gourmet Cafe', 'San Francisco', 'Food & Beverage', 130, Decimal('4000000.00'))
(39, 'Sweet Bakery', 'Miami', 'Food & Beverage', 180, Decimal('3500000.00'))
(69, 'Gourmet Foods', 'Miami', 'Food & Beverage', 250, Decimal('3500000.00'))
(10, 'TravelEase', 'Orlando', 'Travel', 75, Decimal('3000000.00'))
(29, 'Cafe Treats', 'Austin', 'Food & Beverage', 100, Decimal('3000000.00'))
(55, 'LearnEasy', 'Phoenix', 'Education', 220, Decimal('2800000.00'))
(35, 'EduWave', 'Minneapolis', 'Education', 140, Decimal('2500000.00'))
(62, 'Fresh Farms', 'Fresno', 'Agriculture', 200, Decimal('2500000.00'))
(75, 'Bright Minds', 'Minneapolis', 'Education', 130, Decimal('2300000.00'))
(25, 'Learn Academy', 'Philadelphia', 'Education', 180, Decimal('2200000.00'))
(30, 'Global Tours', 'New York', 'Travel', 50, Decimal('2000000.00'))
(19, 'Bakers Delight', 'San Diego', 'Food & Beverage', 150, Decimal('2000000.00'))
(4, 'EduLearn', 'Boston', 'Education', 100, Decimal('2000000.00'))
(45, 'EduBright', 'Portland', 'Education', 160, Decimal('2000000.00'))
(65, 'Knowledge Tree', 'Boston', 'Education', 90, Decimal('1900000.00'))
(50, 'Wanderlust', 'Chicago', 'Travel', 65, Decimal('1800000.00'))
(15, 'Bright Edu', 'Seattle', 'Education', 120, Decimal('1800000.00'))
(20, 'Adventure Co', 'Denver', 'Travel', 60, Decimal('1500000.00'))
(52, 'Green Thumb', 'Salt Lake City', 'Agriculture', 190, Decimal('1500000.00'))
(22, 'GrainCorp', 'Omaha', 'Agriculture', 130, Decimal('1400000.00'))
(72, 'Harvest More', 'Little Rock', 'Agriculture', 170, Decimal('1400000.00'))
(32, 'Farm Fresh', 'Little Rock', 'Agriculture', 160, Decimal('1300000.00'))
(60, 'Explore Co', 'Los Angeles', 'Travel', 55, Decimal('1200000.00'))
(70, 'Sky Travels', 'New York', 'Travel', 55, Decimal('1200000.00'))
(6, 'AgriGrowth', 'Des Moines', 'Agriculture', 150, Decimal('1200000.00'))
(42, 'Harvest', 'Boise', 'Agriculture', 110, Decimal('1100000.00'))
(40, 'JetSet', 'Seattle', 'Travel', 40, Decimal('1000000.00'))
(12, 'Green Fields', 'Fresno', 'Agriculture', 90, Decimal('900000.00'))
(80, 'Dream Vacations', 'Orlando', 'Travel', 35, Decimal('900000.00'))
```

In [ ]:
```
pyhton mysql delete
```

```
delete-statement using only we can delete select record not all records.
  ->we can use wildcard statement .
```

In [37]:
```
#example:delete id number 12
mycursor.execute("delete from company_data where id=12 ")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

In [39]:
```
#delete id 12 then showing all data expect 12 id record.
mycursor.execute("select*from company_data")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
(1, 'Tech Corp', 'San Francisco', 'Technology', 500, Decimal('15000000.00'))
(2, 'FinServ', 'New York', 'Finance', 300, Decimal('8000000.00'))
(3, 'HealthWorks', 'Los Angeles', 'Healthcare', 250, Decimal('5000000.00'))
(4, 'EduLearn', 'Boston', 'Education', 100, Decimal('2000000.00'))
(5, 'Retail King', 'Chicago', 'Retail', 1000, Decimal('22000000.00'))
(6, 'AgriGrowth', 'Des Moines', 'Agriculture', 150, Decimal('1200000.00'))
(7, 'EnergyMax', 'Houston', 'Energy', 400, Decimal('18000000.00'))
(8, 'AutoMotive', 'Detroit', 'Automotive', 600, Decimal('25000000.00'))
(9, 'Foodies', 'New Orleans', 'Food & Beverage', 200, Decimal('5000000.00'))
(10, 'TravelEase', 'Orlando', 'Travel', 75, Decimal('3000000.00'))
(11, 'Tech Hub', 'San Jose', 'Technology', 800, Decimal('32000000.00'))
(13, 'MediCare', 'Houston', 'Healthcare', 350, Decimal('10000000.00'))
(14, 'FinWorld', 'Miami', 'Finance', 450, Decimal('15000000.00'))
(15, 'Bright Edu', 'Seattle', 'Education', 120, Decimal('1800000.00'))
(16, 'HomeWare', 'Dallas', 'Retail', 550, Decimal('8000000.00'))
(17, 'Solar Energy', 'Phoenix', 'Energy', 300, Decimal('12000000.00'))
(18, 'MotorWorks', 'St. Louis', 'Automotive', 350, Decimal('18000000.00'))
(19, 'Bakers Delight', 'San Diego', 'Food & Beverage', 150, Decimal('2000000.00'))
(20, 'Adventure Co', 'Denver', 'Travel', 60, Decimal('1500000.00'))
(21, 'CloudNet', 'San Francisco', 'Technology', 700, Decimal('26000000.00'))
(22, 'GrainCorp', 'Omaha', 'Agriculture', 130, Decimal('1400000.00'))
(23, 'HealNow', 'Austin', 'Healthcare', 300, Decimal('8000000.00'))
(24, 'Invest Corp', 'Charlotte', 'Finance', 250, Decimal('12000000.00'))
(25, 'Learn Academy', 'Philadelphia', 'Education', 180, Decimal('2200000.00'))
(26, 'Mega Mart', 'Atlanta', 'Retail', 750, Decimal('20000000.00'))
(27, 'WindPower', 'Portland', 'Energy', 350, Decimal('16000000.00'))
(28, 'DriveSafe', 'Nashville', 'Automotive', 450, Decimal('22000000.00'))
(29, 'Cafe Treats', 'Austin', 'Food & Beverage', 100, Decimal('3000000.00'))
(30, 'Global Tours', 'New York', 'Travel', 50, Decimal('2000000.00'))
(31, 'InfoTech', 'San Jose', 'Technology', 600, Decimal('24000000.00'))
(32, 'Farm Fresh', 'Little Rock', 'Agriculture', 160, Decimal('1300000.00'))
(33, 'HealthFirst', 'Los Angeles', 'Healthcare', 400, Decimal('12000000.00'))
(34, 'FinanceNow', 'Dallas', 'Finance', 500, Decimal('17000000.00'))
(35, 'EduWave', 'Minneapolis', 'Education', 140, Decimal('2500000.00'))
(36, 'ShopSmart', 'Columbus', 'Retail', 900, Decimal('23000000.00'))
(37, 'PowerGrid', 'Oklahoma City', 'Energy', 380, Decimal('19000000.00'))
(38, 'AutoWorld', 'Indianapolis', 'Automotive', 520, Decimal('24000000.00'))
(39, 'Sweet Bakery', 'Miami', 'Food & Beverage', 180, Decimal('3500000.00'))
(40, 'JetSet', 'Seattle', 'Travel', 40, Decimal('1000000.00'))
(41, 'ByteNet', 'San Francisco', 'Technology', 900, Decimal('35000000.00'))
(42, 'Harvest', 'Boise', 'Agriculture', 110, Decimal('1100000.00'))
(43, 'LifeCare', 'Boston', 'Healthcare', 360, Decimal('9000000.00'))
(44, 'TradeWise', 'Denver', 'Finance', 320, Decimal('11000000.00'))
(45, 'EduBright', 'Portland', 'Education', 160, Decimal('2000000.00'))
(46, 'RetailMart', 'Charlotte', 'Retail', 800, Decimal('21000000.00'))
(47, 'SolarWave', 'Las Vegas', 'Energy', 260, Decimal('13000000.00'))
(48, 'CarZone', 'Memphis', 'Automotive', 580, Decimal('25000000.00'))
(49, 'Gourmet Cafe', 'San Francisco', 'Food & Beverage', 130, Decimal('4000000.00'))
(50, 'Wanderlust', 'Chicago', 'Travel', 65, Decimal('1800000.00'))
(51, 'SoftTech', 'San Jose', 'Technology', 700, Decimal('29000000.00'))
(52, 'Green Thumb', 'Salt Lake City', 'Agriculture', 190, Decimal('1500000.00'))
(53, 'MedCentral', 'Houston', 'Healthcare', 500, Decimal('15000000.00'))
(54, 'InvestSmart', 'Miami', 'Finance', 600, Decimal('20000000.00'))
(55, 'LearnEasy', 'Phoenix', 'Education', 220, Decimal('2800000.00'))
(56, 'ShopVille', 'New York', 'Retail', 850, Decimal('24000000.00'))
(57, 'EnergyStream', 'Austin', 'Energy', 420, Decimal('18000000.00'))
(58, 'AutoPlus', 'Cleveland', 'Automotive', 640, Decimal('28000000.00'))
(59, 'Deli Delight', 'Boston', 'Food & Beverage', 210, Decimal('4500000.00'))
(60, 'Explore Co', 'Los Angeles', 'Travel', 55, Decimal('1200000.00'))
(61, 'Tech Innovate', 'San Francisco', 'Technology', 450, Decimal('16000000.00'))
(62, 'Fresh Farms', 'Fresno', 'Agriculture', 200, Decimal('2500000.00'))
(63, 'Wellness Center', 'Houston', 'Healthcare', 320, Decimal('8500000.00'))
(64, 'Safe Invest', 'New York', 'Finance', 400, Decimal('13500000.00'))
(65, 'Knowledge Tree', 'Boston', 'Education', 90, Decimal('1900000.00'))
(66, 'QuickMart', 'Dallas', 'Retail', 700, Decimal('23000000.00'))
(67, 'BioFuel Corp', 'Phoenix', 'Energy', 500, Decimal('17000000.00'))
(68, 'Fast Motors', 'Detroit', 'Automotive', 450, Decimal('21000000.00'))
(69, 'Gourmet Foods', 'Miami', 'Food & Beverage', 250, Decimal('3500000.00'))
(70, 'Sky Travels', 'New York', 'Travel', 55, Decimal('1200000.00'))
(71, 'Tech Solutions', 'San Jose', 'Technology', 600, Decimal('24000000.00'))
(72, 'Harvest More', 'Little Rock', 'Agriculture', 170, Decimal('1400000.00'))
(73, 'HealthLine', 'Austin', 'Healthcare', 290, Decimal('7800000.00'))
(74, 'Capital Trust', 'Charlotte', 'Finance', 350, Decimal('13000000.00'))
(75, 'Bright Minds', 'Minneapolis', 'Education', 130, Decimal('2300000.00'))
(76, 'Market Plaza', 'Columbus', 'Retail', 850, Decimal('25000000.00'))
(77, 'Green Energy', 'Portland', 'Energy', 380, Decimal('18000000.00'))
(78, 'MotorWorld', 'Indianapolis', 'Automotive', 480, Decimal('20000000.00'))
(79, 'Taste Buds', 'San Diego', 'Food & Beverage', 210, Decimal('4100000.00'))
(80, 'Dream Vacations', 'Orlando', 'Travel', 35, Decimal('900000.00'))
```

python mysql drop table:

->you can delete an entire data existing table by using drop statement: i will show another table employees data and use drop statement

In [41]:
```
mycursor.execute("select*from employees")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
(1, 'mani', 'it', datetime.date(2003, 4, 2), Decimal('350.00'), 'mai@gmail.com', '9346092985', 'guntur', 'IND',
datetime.date(2003, 4, 12))
(2, 'tinku', 'vasu', datetime.date(2001, 4, 2), Decimal('200.00'), 'tinku@gmail.com', '9498833080', 'snk', 'IND'
, None)
(3, 'sai', 'D.S.', datetime.date(2002, 4, 2), Decimal('350.00'), 'sai@gmail.com', '9014797137', 'zphs', 'IND', d
atetime.date(2004, 2, 4))
(4, 'srinu', 'ec', datetime.date(2004, 3, 2), Decimal('500.00'), 'srinu@gmail.com', '9014577137', 'zphs', 'IND',
None)
(5, 'ravi', 'vasu', datetime.date(2004, 5, 2), Decimal('400.00'), 'ravi@gmail.com', '9876543210', 'zphs', 'IND',
None)
(6, 'parasu', 'vasu', datetime.date(2005, 6, 2), Decimal('500.00'), 'parasu@gmail.com', None, 'zphs', 'IND', dat
etime.date(2004, 2, 4))
(7, 'vamsi', 'vasu', datetime.date(2003, 7, 2), Decimal('600.00'), 'vamsi@gmail.com', None, 'zphs', 'IND', None)
(8, 'manoj', 'vasu', datetime.date(2002, 8, 2), Decimal('600.00'), 'manoj@gmail.com', None, 'zphs', 'IND', datet
ime.date(2004, 2, 4))
(9, 'gopi', 'it', datetime.date(1999, 4, 2), Decimal('450.00'), 'gopi@gmail.com', '9876543201', 'snk', 'IND', da
tetime.date(2023, 2, 4))
(10, 'gopi', 'mech', datetime.date(1999, 4, 2), Decimal('450.00'), 'venkatappa@gmail.com', '9876543201', 'snk',
'IND', datetime.date(2023, 2, 4))
(11, 'go', 'it', datetime.date(1999, 4, 2), Decimal('450.00'), 'go2gmail.com', '9876543201', 'snk', 'world', dat
etime.date(2023, 2, 4))
(12, 'go', 'it', datetime.date(1999, 4, 2), Decimal('450.00'), 'go@gmail.com', '9876543201', 'snk', 'world', dat
etime.date(2023, 2, 4))
(13, 'hari', 'hr', datetime.date(2024, 12, 23), Decimal('4000.00'), 'hari@gmail.com', '3849388398', 'gid', 'pak'
, datetime.date(2024, 9, 13))
(14, 'hari', 'hr', datetime.date(2024, 12, 23), Decimal('4000.00'), 'hari@gmail.com', '3849388398', 'gid', 'pak'
, datetime.date(2024, 9, 13))
```

In [43]:
```
mycursor.execute("drop table employees")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

In [47]:
```
#drop statement table will exsit now show only one table
mycursor.execute("show tables")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
('company_data',)
('departments',)
('employees',)
```

pyhton mysql update table ->by using update statement you can existing records in table. -> and also we can use sql injection (%s)

In [17]:
```
#example:
mycursor.execute("update  company_data set location='snk'  where location='new york'")
print(mycursor.rowcount,"records(s) affected")
mycursor.execute("select*from company_data where location='snk' ")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
0 records(s) affected
(2, 'FinServ', 'snk', 'Finance', 300, Decimal('8000000.00'))
(30, 'Global Tours', 'snk', 'Travel', 50, Decimal('2000000.00'))
(56, 'ShopVille', 'snk', 'Retail', 850, Decimal('24000000.00'))
(64, 'Safe Invest', 'snk', 'Finance', 400, Decimal('13500000.00'))
(70, 'Sky Travels', 'snk', 'Travel', 55, Decimal('1200000.00'))
```

python mysql limit

->only limit of records returned from data table. ->use 'limit"

In [19]:
```
mycursor.execute("select *from company_data limit 10")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
(1, 'Tech Corp', 'San Francisco', 'Technology', 500, Decimal('15000000.00'))
(2, 'FinServ', 'snk', 'Finance', 300, Decimal('8000000.00'))
(3, 'HealthWorks', 'Los Angeles', 'Healthcare', 250, Decimal('5000000.00'))
(4, 'EduLearn', 'Boston', 'Education', 100, Decimal('2000000.00'))
(5, 'Retail King', 'Chicago', 'Retail', 1000, Decimal('22000000.00'))
(6, 'AgriGrowth', 'Des Moines', 'Agriculture', 150, Decimal('1200000.00'))
(7, 'EnergyMax', 'Houston', 'Energy', 400, Decimal('18000000.00'))
(8, 'AutoMotive', 'Detroit', 'Automotive', 600, Decimal('25000000.00'))
(9, 'Foodies', 'New Orleans', 'Food & Beverage', 200, Decimal('5000000.00'))
(10, 'TravelEase', 'Orlando', 'Travel', 75, Decimal('3000000.00'))
```

use "offset" keyword ->if you want to return 10 are up to us records. ->starting from the six record ,you can use 'offset'

In [25]:
```
mycursor.execute("select*from company_data limit 10 offset 5")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
(6, 'AgriGrowth', 'Des Moines', 'Agriculture', 150, Decimal('1200000.00'))
(7, 'EnergyMax', 'Houston', 'Energy', 400, Decimal('18000000.00'))
(8, 'AutoMotive', 'Detroit', 'Automotive', 600, Decimal('25000000.00'))
(9, 'Foodies', 'New Orleans', 'Food & Beverage', 200, Decimal('5000000.00'))
(10, 'TravelEase', 'Orlando', 'Travel', 75, Decimal('3000000.00'))
(11, 'Tech Hub', 'San Jose', 'Technology', 800, Decimal('32000000.00'))
(13, 'MediCare', 'Houston', 'Healthcare', 350, Decimal('10000000.00'))
(14, 'FinWorld', 'Miami', 'Finance', 450, Decimal('15000000.00'))
(15, 'Bright Edu', 'Seattle', 'Education', 120, Decimal('1800000.00'))
(16, 'HomeWare', 'Dallas', 'Retail', 550, Decimal('8000000.00'))
```

python mysql join

-> join two are more tables ->combine rows from two aare more tables. ->based on related column between them,by using "join" statement. ->1.inner join ->2.left join ->3.right join

In [77]:
```
mycursor.execute("select*from departments")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
(1, 'HR')
(2, 'Engineering')
(3, 'Marketing')
```

In [81]:
```
mycursor.execute("select*from employees")
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
(101, 'Alice', 1)
(102, 'Bob', 2)
(103, 'Charlie', 3)
(104, 'David', 2)
```

In [87]:
```
#inner join
mycursor.execute('''
select emp_name,dept_name from employees as emp
inner join departments as dept
on emp.department_id=dept.dept_id ''')
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
('Alice', 'HR')
('Bob', 'Engineering')
('David', 'Engineering')
('Charlie', 'Marketing')
```

In [91]:
```
#left join
mycursor.execute('''
select emp_name,dept_name from employees as emp
left join departments as dept
on emp.department_id=dept.dept_id ''')
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
('Alice', 'HR')
('Bob', 'Engineering')
('Charlie', 'Marketing')
('David', 'Engineering')
```

In [93]:
```
#right join
mycursor.execute('''
select emp_name,dept_name from employees as emp
right join departments as dept
on emp.department_id=dept.dept_id''')
myresult=mycursor.fetchall()
for x in myresult:
    print(x)
```

```
('Alice', 'HR')
('Bob', 'Engineering')
('David', 'Engineering')
('Charlie', 'Marketing')
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: