VINOD KUMAR VERMA, PGT(CS), KV OEF KANPUR &
SACHIN BHARDWAJ, PGT(CS), KV NO.1 TEZPUR

INTERFACE PYTHON WITH MYSQL

Connecting Python application with MySQL

Introduction

- □ Every application required data to be stored for future reference to manipulate data. Today every application stores data in database for this purpose
- □ For example, reservation system stores passengers details for reserving the seats and later on for sending some messages or for printing tickets etc.
- □ In school student details are saved for many reasons like attendance, fee collections, exams, report card etc.
- □ Python allows us to connect all types of database like Oracle, SQL Server, MySQL.
- □ In our syllabus we have to understand how to connect Python programs with MySQL

Pre-requisite to connect Python with MySQL

- Before we connect python program with any database like MySQL we need to build a bridge to connect Python and MySQL.
- To build this bridge so that data can travel both ways we need a connector called “mysql.connector”.
- We can install “mysql.connector” by using following methods:
  - At command prompt (Administrator login)
    - Type “pip install mysql.connector” and press enter
      - (internet connection in required)
      - This connector will work only for MySQL 5.7.3 or later
  - Or open “https://dev.mysql.com/downloads/connector/python/”

  and download connector as per OS and Python version

Connecting to MySQL from Python

- Once the connector is installed you are ready to connect your python program to MySQL.
- The following steps to follow while connecting your python program with MySQL
  - Open python
  - Import the package required (import mysql.connector)
  - Open the connection to database
  - Create a cursor instance
  - Execute the query and store it in resultset
  - Extract data from resultset
  - Clean up the environment

Importing mysql.connector

```
import mysql.connector
```

Or

```
import mysql.connector as ms
```

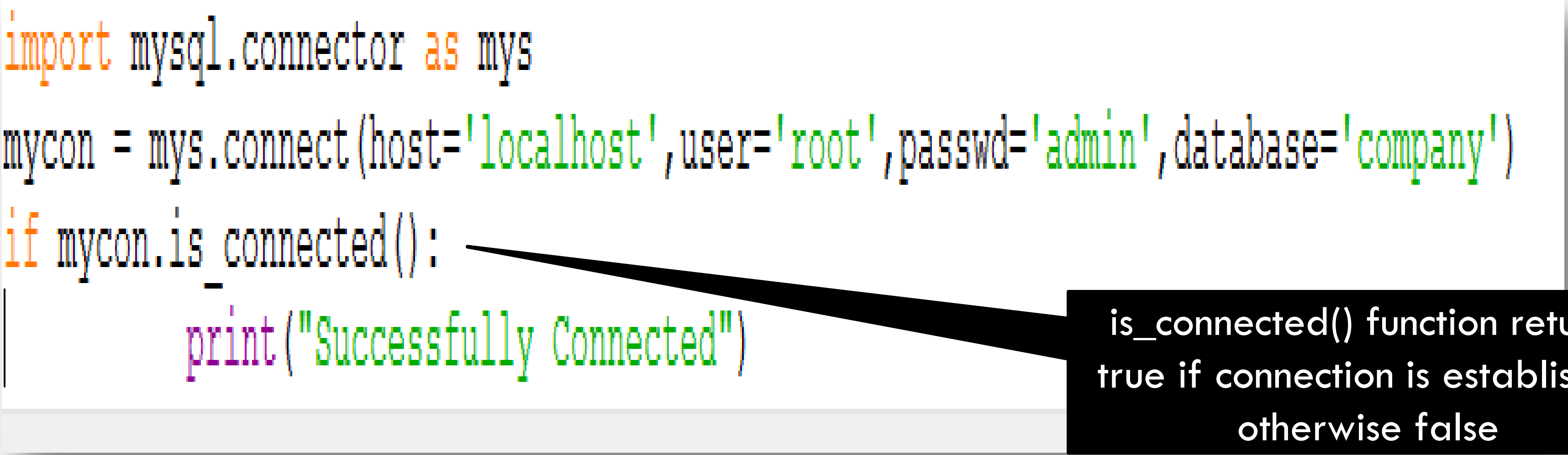
Here “ms” is an alias, so every time we can use “ms” in place of “mysql.connector”

Open a connection to MySQL Database

- To create connection, connect() function is used
- Its syntax is:
  - connect(**host**=**<server_name>**,**user**=**<user_name>**, **passwd**=**<password>**[,**database**=**<database>**])

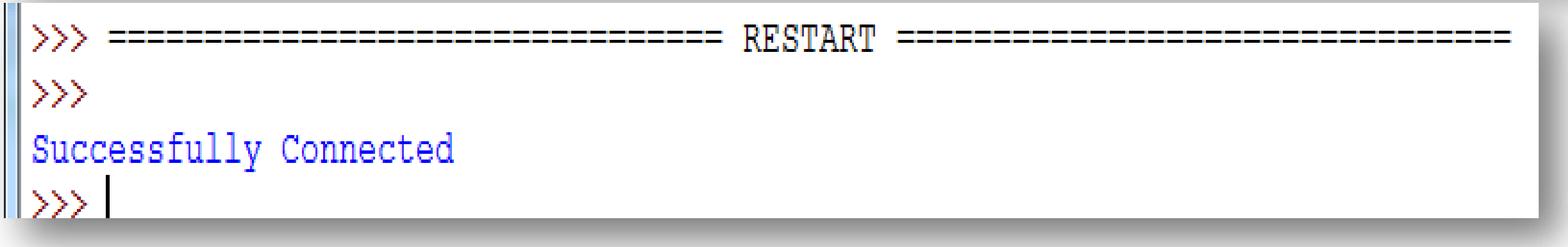
- Here server_name means database servername, generally it is given as "localhost"
- User_name means user by which we connect with mysql generally it is given as "root"
- Password is the password of user "root"
- Database is the name of database whose data(table) we want to use

Example: To establish connection with MySQL



```
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
if mycon.is_connected():
        print("Successfully Connected")
```

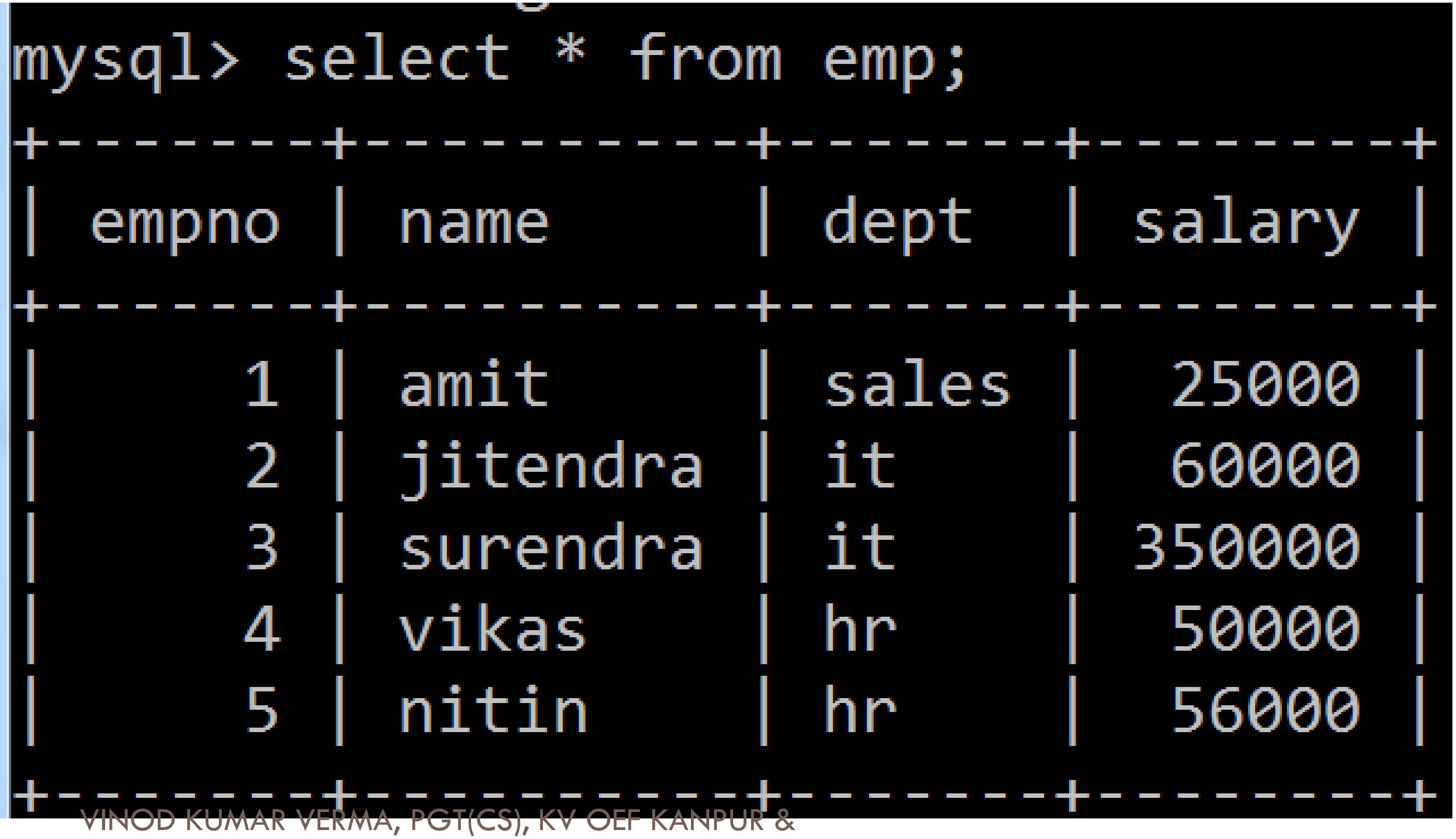
is_connected() function returns true if connection is established otherwise false



```
>>> ============================== RESTART ==============================
>>>
Successfully Connected
>>> 
```

"mys" is an alias of package "mysql.connector"
"mycon" is connection object which stores connection established with MySQL
"connect()" function is used to connect with mysql by specifying parameters like host, user, passwd, database

Table to work (emp)



```
mysql> select * from emp;
+-------+----------+--------+--------+
| empno | name     | dept   | salary |
+-------+----------+--------+--------+
|     1 | amit     | sales  |  25000 |
|     2 | jitendra | it     |  60000 |
|     3 | surendra | it     | 350000 |
|     4 | vikas    | hr     |  50000 |
|     5 | nitin    | hr     |  56000 |
+-------+----------+--------+--------+
```

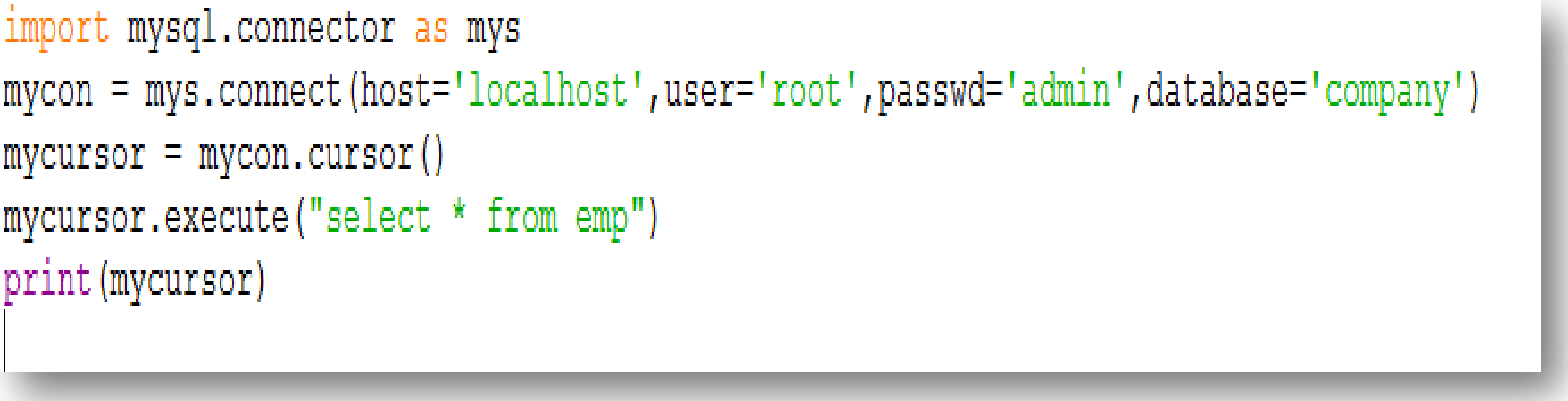
Creating **Cursor**

- It is a useful control structure of database connectivity.
- When we fire a query to database, it is executed and resultset (set of records) is sent over he connection in one go.
- We may want to access data one row at a time, but query processing cannot happens as one row at a time, so cursor help us in performing this task. Cursor stores all the data as a temporary container of returned data and we can fetch data one row at a time from Cursor.

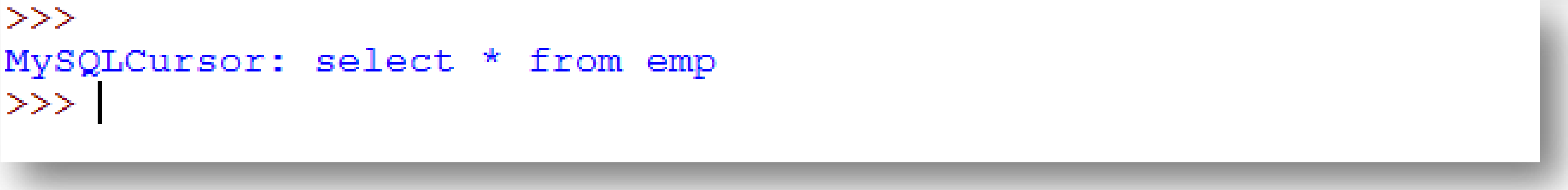
Creating **Cursor and Executing Query**

- **TO CREATE CURSOR**
  - Cursor_name = connectionObject.cursor()
  - For e.g.
  - mycursor = mycon.cursor()
- **TO EXECUTE QUERY**
- We use **execute()** function to send query to connection
- Cursor_name.execute(query)
- For e.g.
- mycursor.execute('select * from emp')

Example - Cursor



```python
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
mycursor = mycon.cursor()
mycursor.execute("select * from emp")
print(mycursor)
```



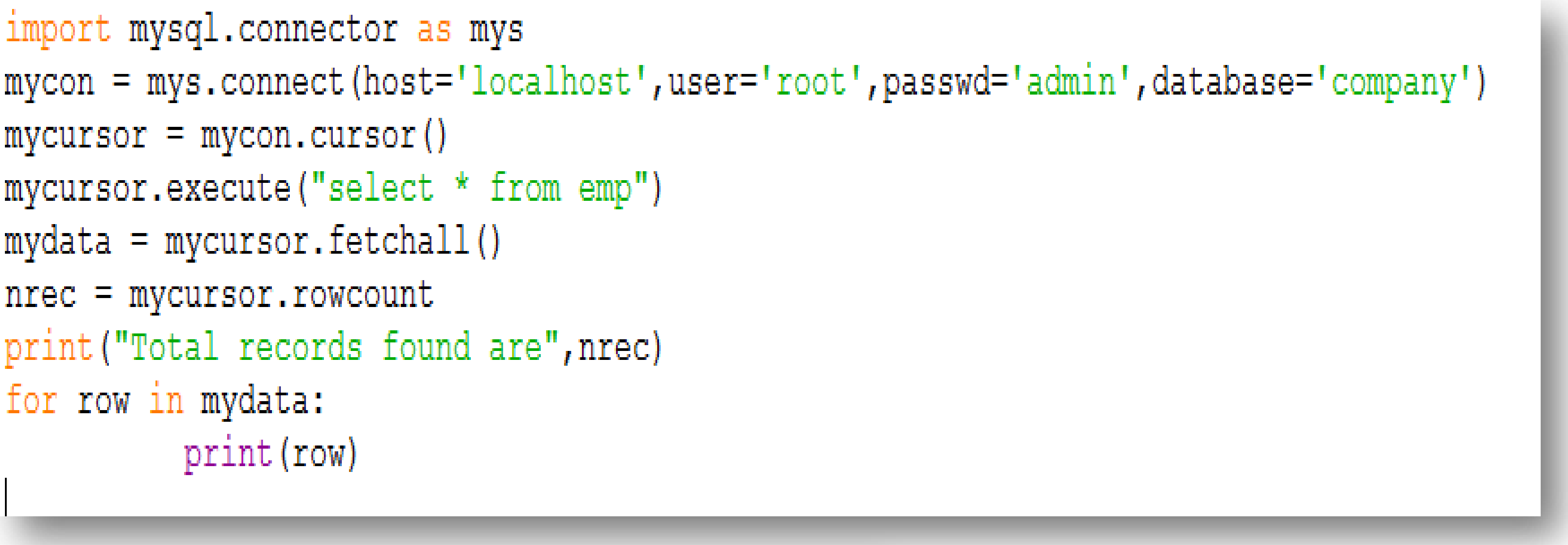
```
>>>
MySQLCursor: select * from emp
>>> 
```

Output shows cursor is created and query is fired and stored, but no data is coming. To fetch data we have to use functions like fetchall(), fetchone(), fetchmany() are used

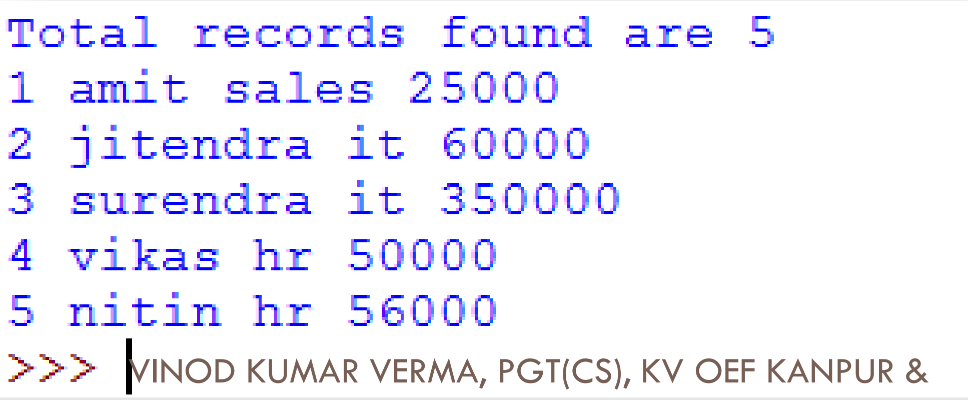
Fetching(extracting) data from ResultSet

- To extract data from cursor following functions are used:
  - **fetchall()** : it will return all the record in the form of tuple.
  - **fetchone()** : it return one record from the result set. i.e. first time it will return first record, next time it will return second record and so on. If no more record it will return **None**
  - **fetchmany(n)** : it will return n number of records. It no more record it will return an **empty tuple.**
  - **rowcount** : it will return number of rows retrieved from the cursor so far.

Example – fetchall()

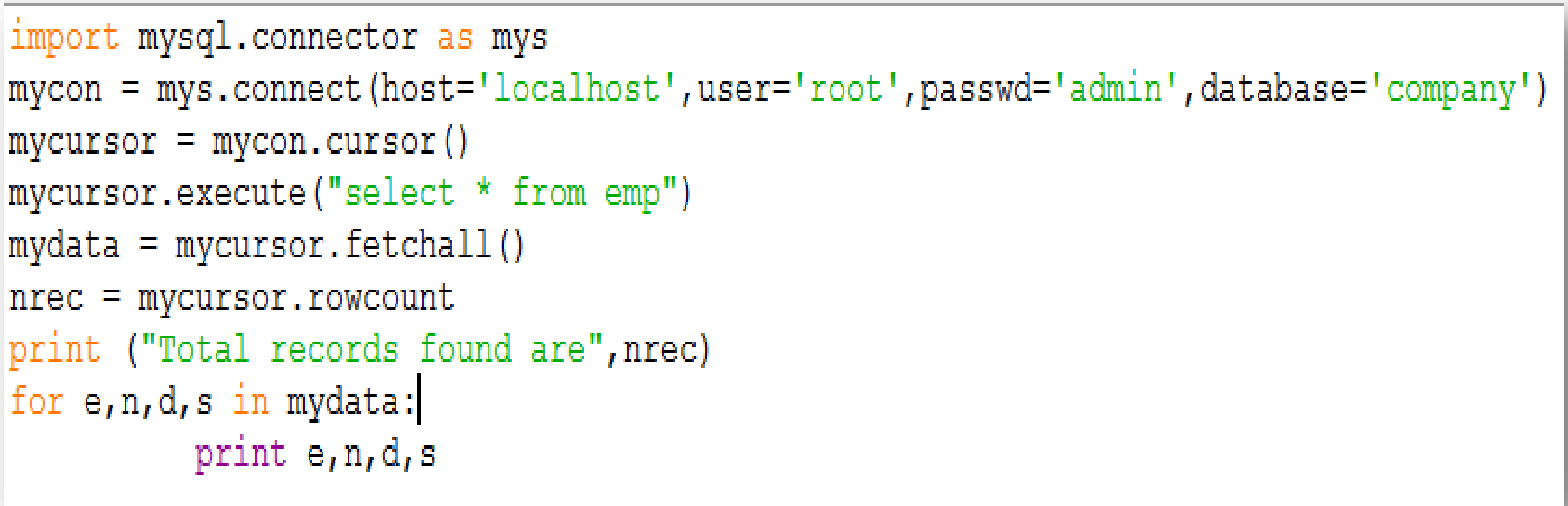


```
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
mycursor = mycon.cursor()
mycursor.execute("select * from emp")
mydata = mycursor.fetchall()
nrec = mycursor.rowcount
print("Total records found are",nrec)
for row in mydata:
        print(row)
```

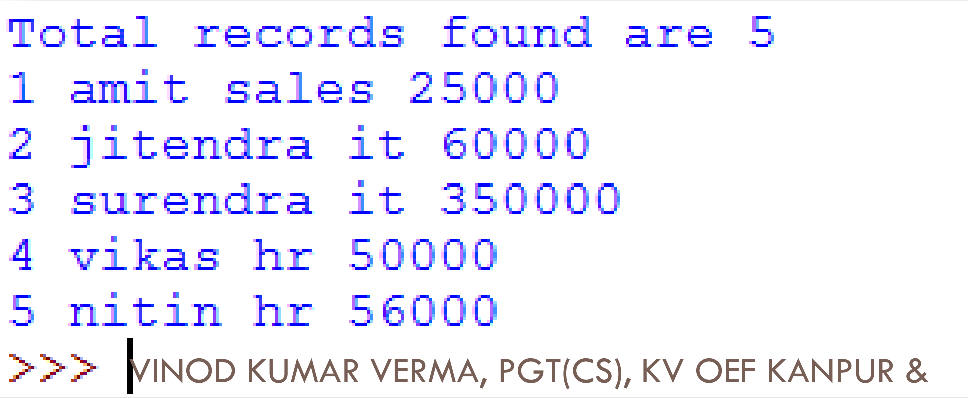


```
Total records found are 5
1 amit sales 25000
2 jitendra it 60000
3 surendra it 350000
4 vikas hr 50000
5 nitin hr 56000
>>> 
```

Example 2 – fetchall()

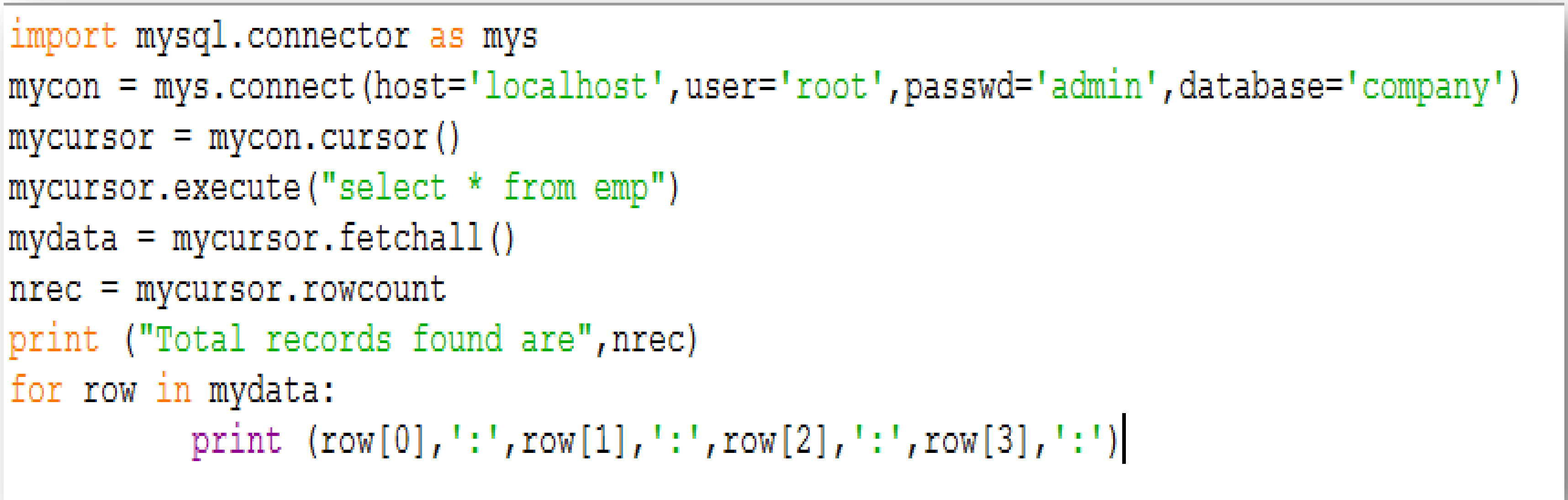


```
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
mycursor = mycon.cursor()
mycursor.execute("select * from emp")
mydata = mycursor.fetchall()
nrec = mycursor.rowcount
print ("Total records found are",nrec)
for e,n,d,s in mydata:
          print e,n,d,s
```

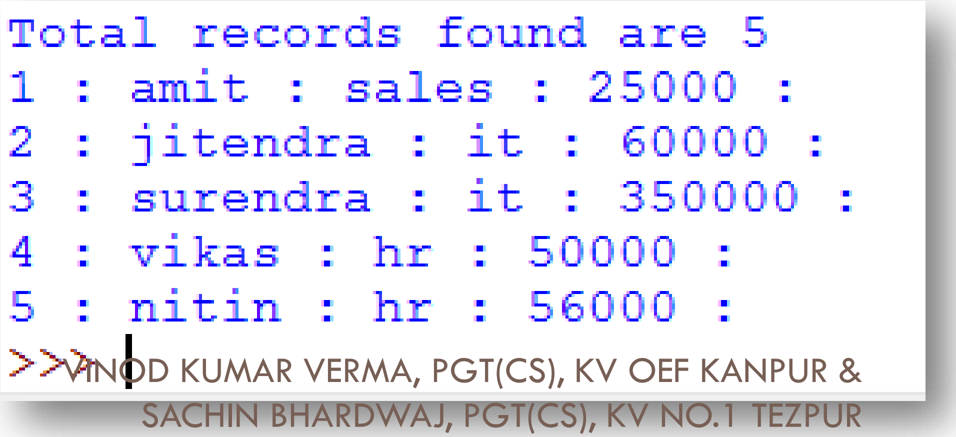


```
Total records found are 5
1 amit sales 25000
2 jitendra it 60000
3 surendra it 350000
4 vikas hr 50000
5 nitin hr 56000
>>> 
```

Example 3 – fetchall()

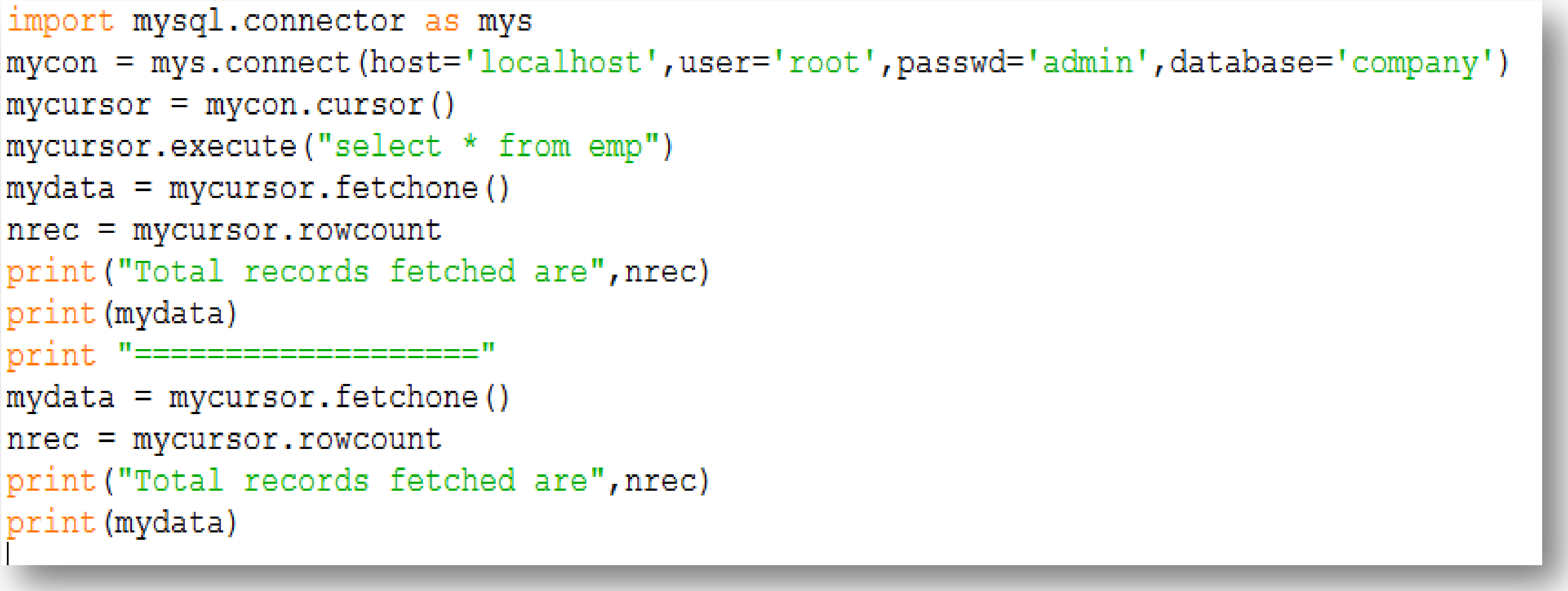


```
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
mycursor = mycon.cursor()
mycursor.execute("select * from emp")
mydata = mycursor.fetchall()
nrec = mycursor.rowcount
print ("Total records found are",nrec)
for row in mydata:
        print (row[0],':',row[1],':',row[2],':',row[3],':')
```

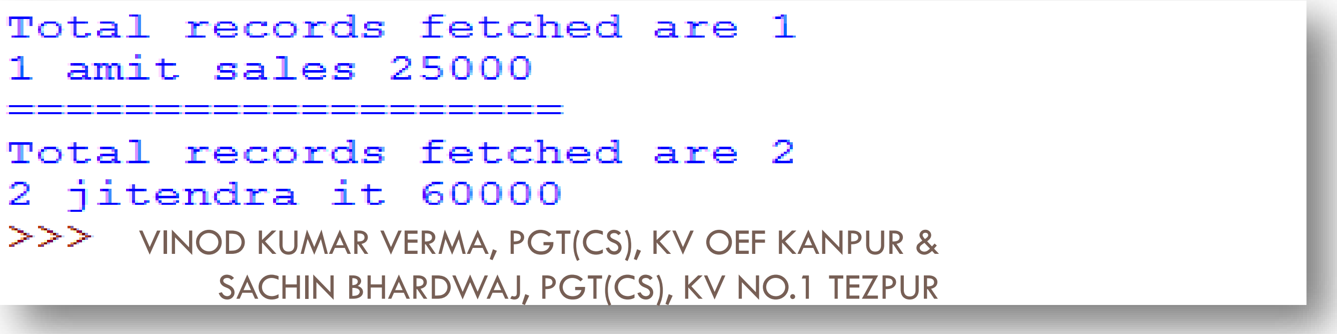


```
Total records found are 5
1 : amit : sales : 25000 :
2 : jitendra : it : 60000 :
3 : surendra : it : 350000 :
4 : vikas : hr : 50000 :
5 : nitin : hr : 56000 :
>>>
```

Example 4: fetchone()

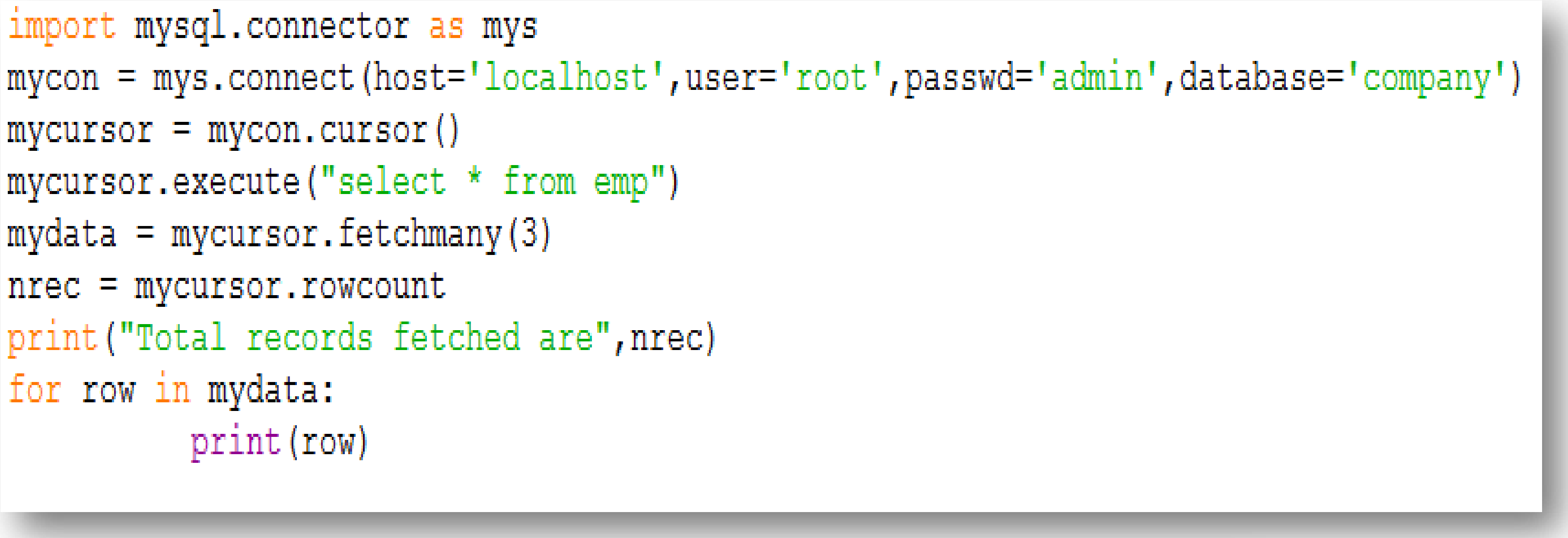


```
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
mycursor = mycon.cursor()
mycursor.execute("select * from emp")
mydata = mycursor.fetchone()
nrec = mycursor.rowcount
print("Total records fetched are",nrec)
print(mydata)
print "==================="
mydata = mycursor.fetchone()
nrec = mycursor.rowcount
print("Total records fetched are",nrec)
print(mydata)
```

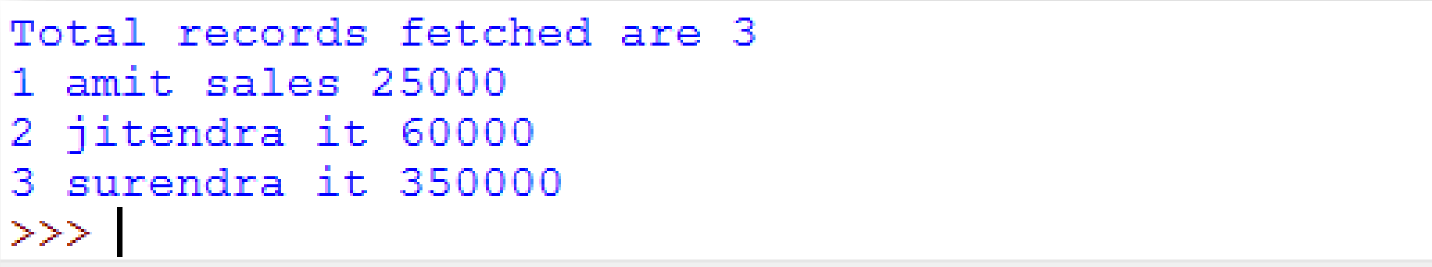


```
Total records fetched are 1
1 amit sales 25000
===================
Total records fetched are 2
2 jitendra it 60000
>>>
```

Example 5: fetchmany(n)

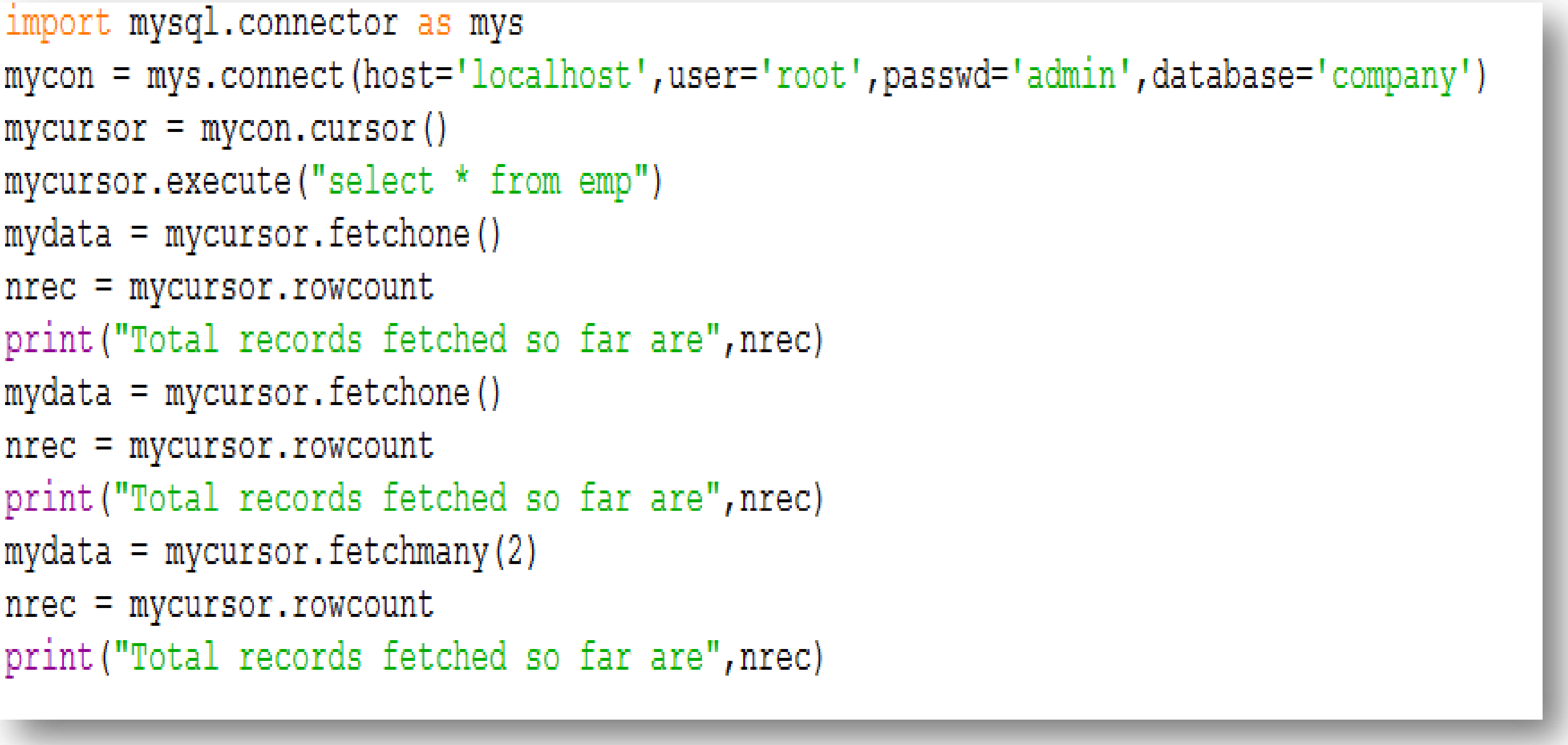


```
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
mycursor = mycon.cursor()
mycursor.execute("select * from emp")
mydata = mycursor.fetchmany(3)
nrec = mycursor.rowcount
print("Total records fetched are",nrec)
for row in mydata:
        print(row)
```



```
Total records fetched are 3
1 amit sales 25000
2 jitendra it 60000
3 surendra it 350000
>>> 
```

Guess the output

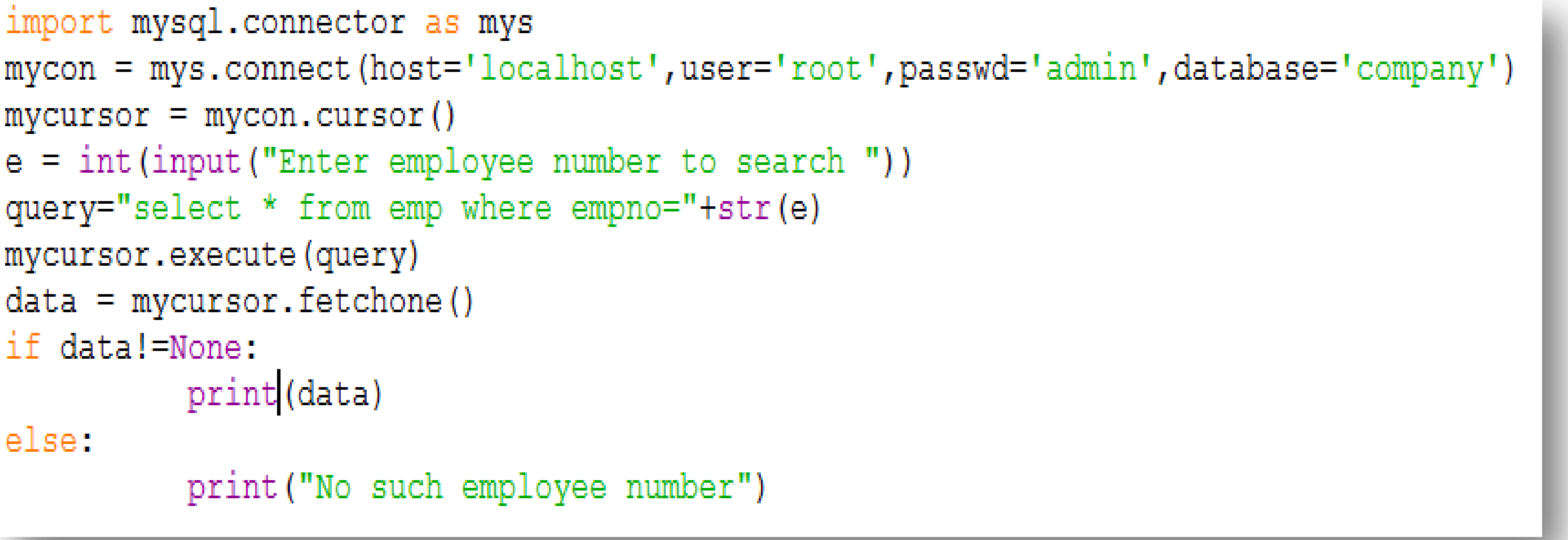


```
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
mycursor = mycon.cursor()
mycursor.execute("select * from emp")
mydata = mycursor.fetchone()
nrec = mycursor.rowcount
print("Total records fetched so far are",nrec)
mydata = mycursor.fetchone()
nrec = mycursor.rowcount
print("Total records fetched so far are",nrec)
mydata = mycursor.fetchmany(2)
nrec = mycursor.rowcount
print("Total records fetched so far are",nrec)
```

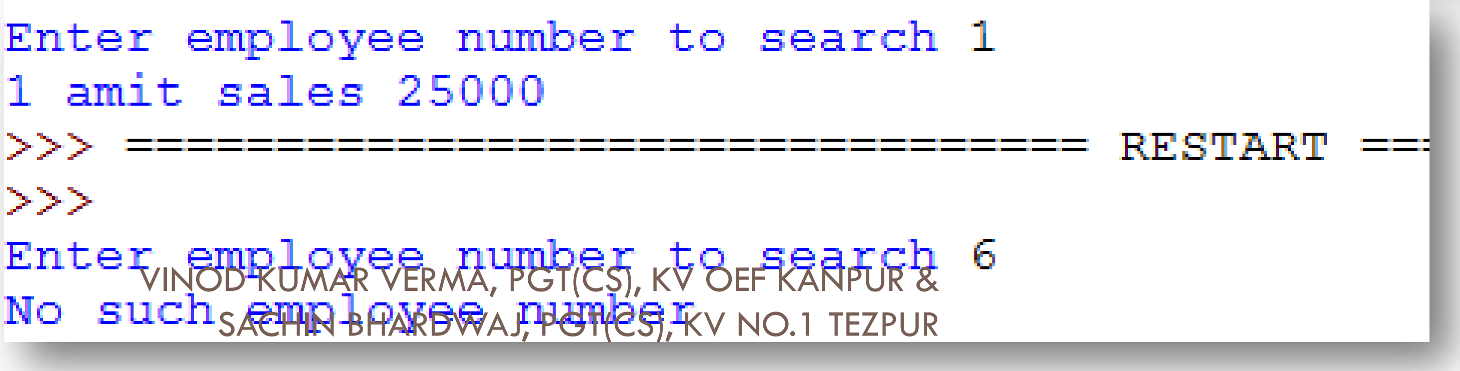
Parameterized Query

- We can pass values to query to perform dynamic search like we want to search for any employee number entered during runtime or to search any other column values.
- To Create Parameterized query we can use various methods like:
  - Concatenating dynamic variable to query in which values are entered.
  - String template with % formatting
  - String template with {} and format function

Concatenating variable with query



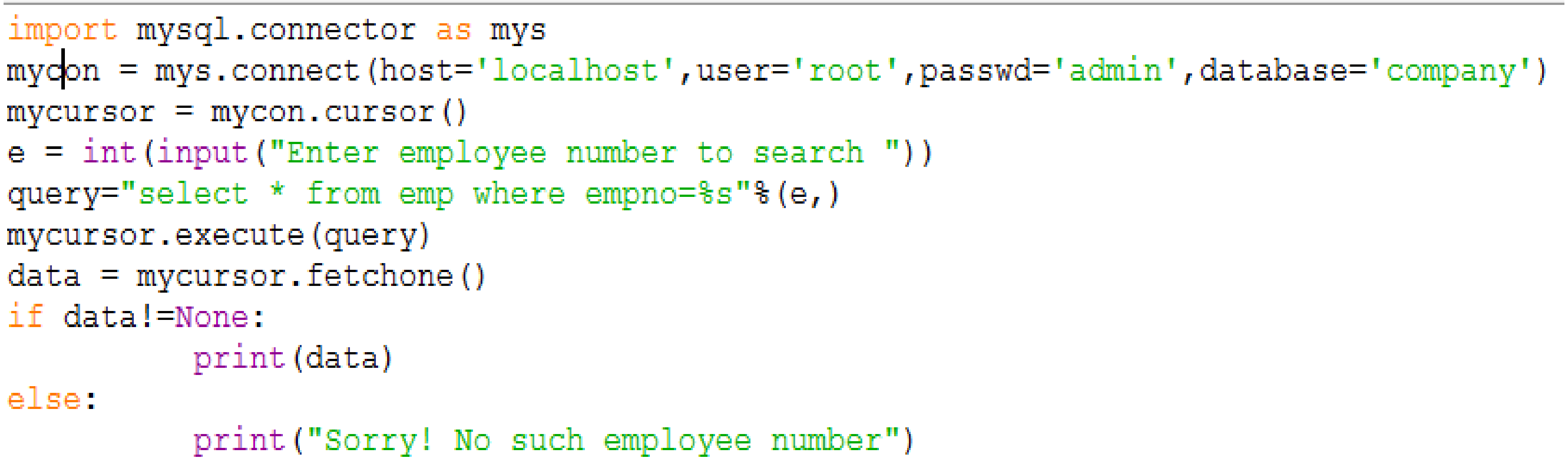
```
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
mycursor = mycon.cursor()
e = int(input("Enter employee number to search "))
query="select * from emp where empno="+str(e)
mycursor.execute(query)
data = mycursor.fetchone()
if data!=None:
        print(data)
else:
        print("No such employee number")
```



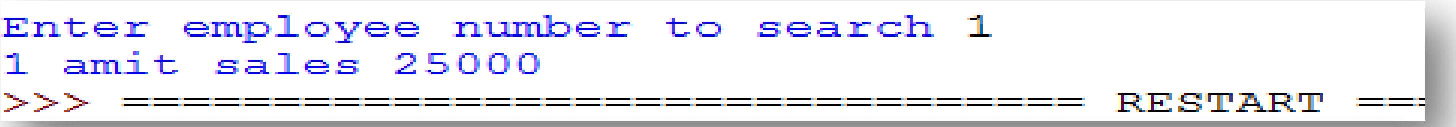
```
Enter employee number to search 1
1 amit sales 25000
>>> ============================== RESTART ===
>>>
Enter employee number to search 6
No such employee number
```

String template with %s formatting

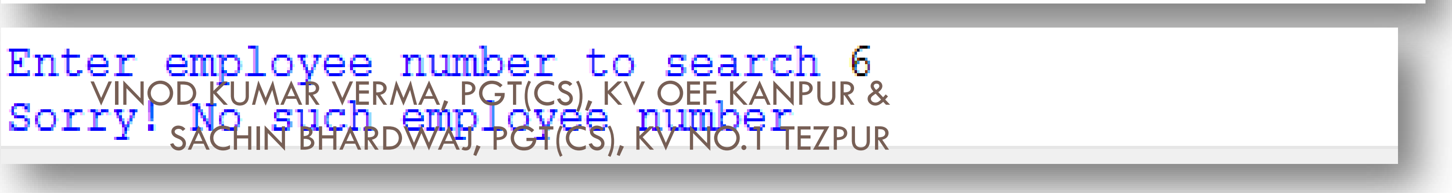
- In this method we will use %s in place of values to substitute and then pass the value for that place.



```
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
mycursor = mycon.cursor()
e = int(input("Enter employee number to search "))
query="select * from emp where empno=%s"%(e,)
mycursor.execute(query)
data = mycursor.fetchone()
if data!=None:
          print(data)
else:
          print("Sorry! No such employee number")
```

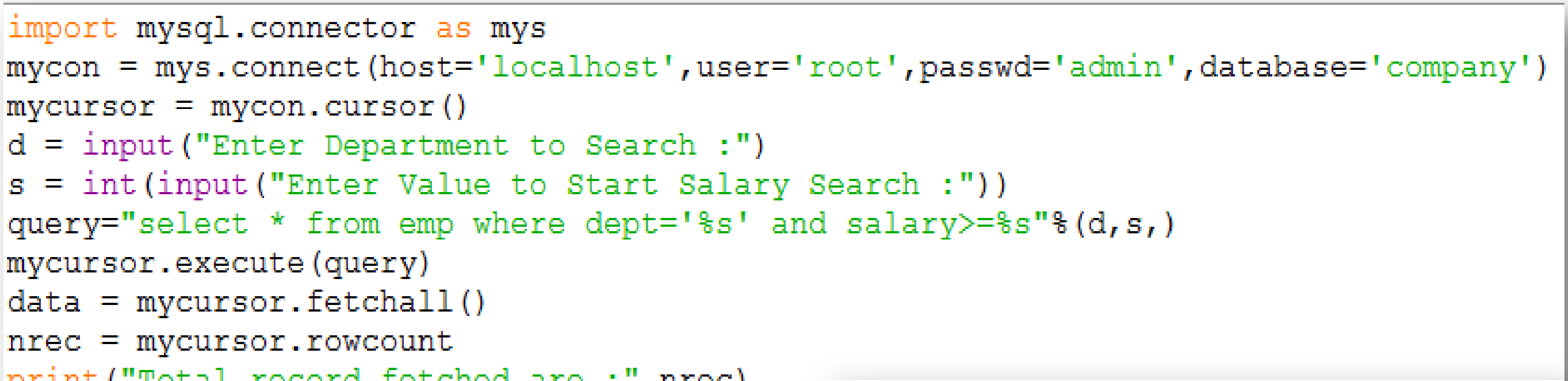


```
Enter employee number to search 1
1 amit sales 25000
>>> ============================== RESTART ===
```

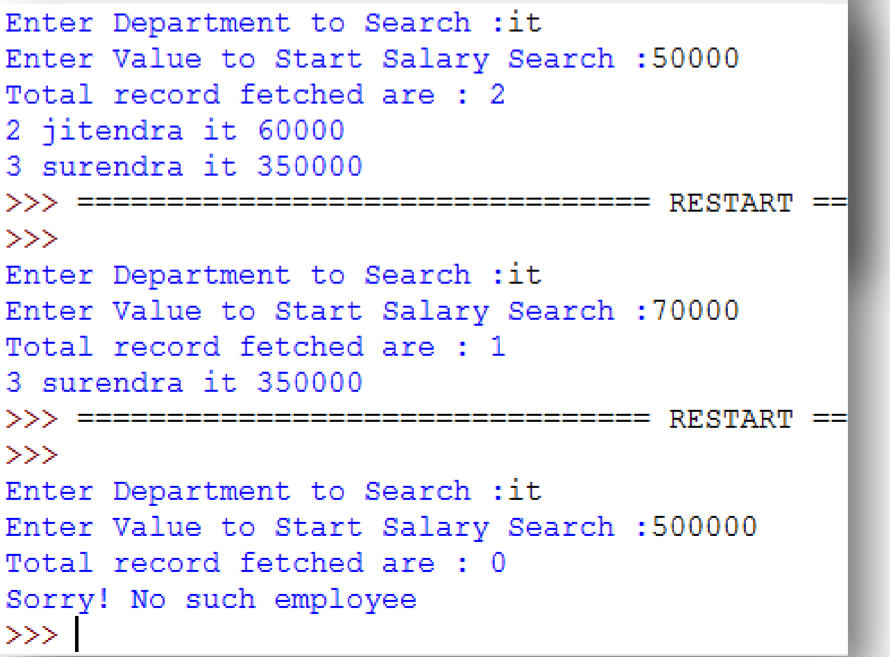


```
Enter employee number to search 6
Sorry! No such employee number
```

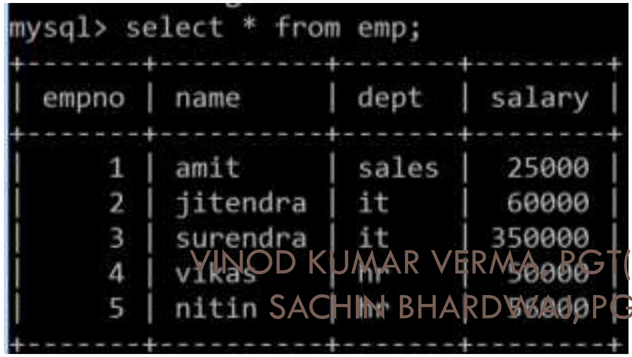
String template with %s formatting



```
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
mycursor = mycon.cursor()
d = input("Enter Department to Search :")
s = int(input("Enter Value to Start Salary Search :"))
query="select * from emp where dept='%s' and salary>=%s"%(d,s,)
mycursor.execute(query)
data = mycursor.fetchall()
nrec = mycursor.rowcount
print("Total record fetched are :",nrec)
if nrec!=0:
          for row in data:
                    print(row)
else:
          print("Sorry! No such employee ")
```



```
Enter Department to Search :it
Enter Value to Start Salary Search :50000
Total record fetched are : 2
2 jitendra it 60000
3 surendra it 350000
>>> ================================ RESTART ==
>>>
Enter Department to Search :it
Enter Value to Start Salary Search :70000
Total record fetched are : 1
3 surendra it 350000
>>> ================================ RESTART ==
>>>
Enter Department to Search :it
Enter Value to Start Salary Search :500000
Total record fetched are : 0
Sorry! No such employee
>>> |
```

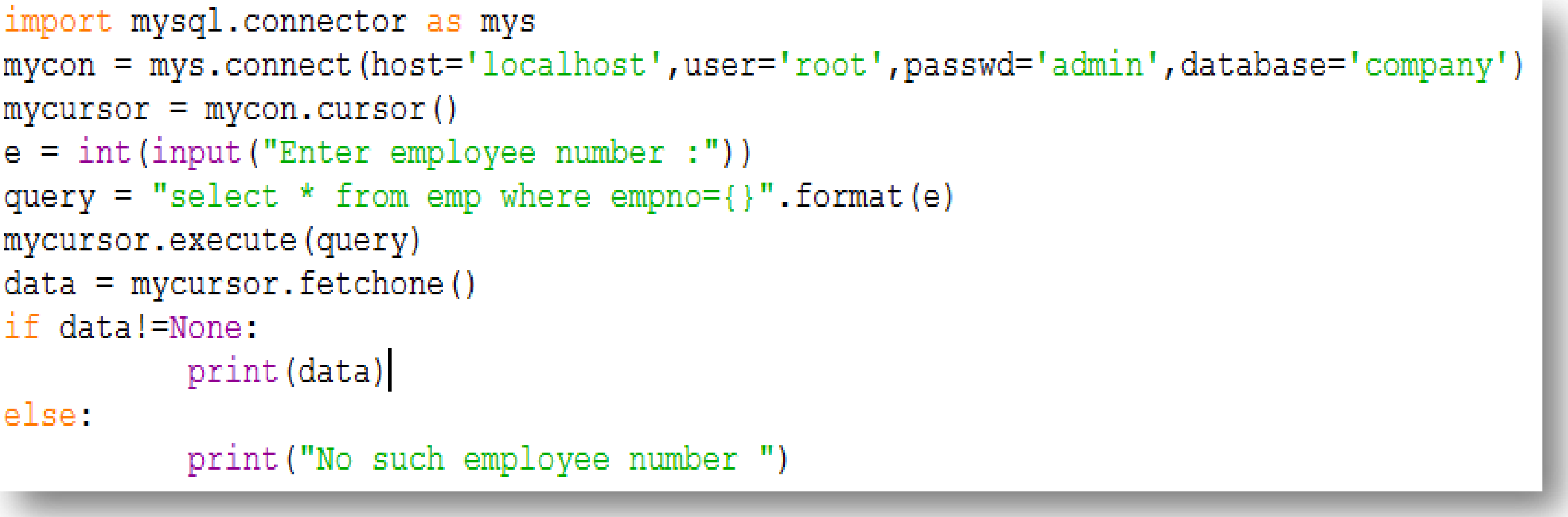


```
mysql> select * from emp;
+-------+----------+-------+--------+
| empno | name     | dept  | salary |
+-------+----------+-------+--------+
|     1 | amit     | sales |  25000 |
|     2 | jitendra | it    |  60000 |
|     3 | surendra | it    | 350000 |
|     4 | vikas    | hr    |  50000 |
|     5 | nitin    | hr    |  56000 |
+-------+----------+-------+--------+
```

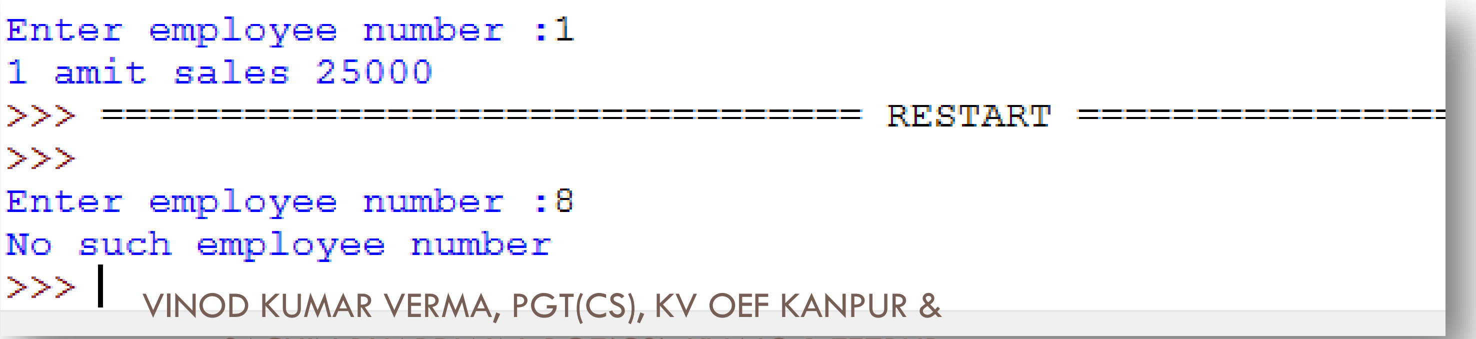
String template with {} and format()

- In this method in place of %s we will use {} and to pass values for these placeholder format() is used. Inside we can optionally give 0,1,2… values for e.g. {0},{1} but its not mandatory. we can also optionally pass named parameter inside {} so that while passing values through format function we need not to remember the order of value to pass. For e.g. {roll},{name} etc.

String template with {} and format()

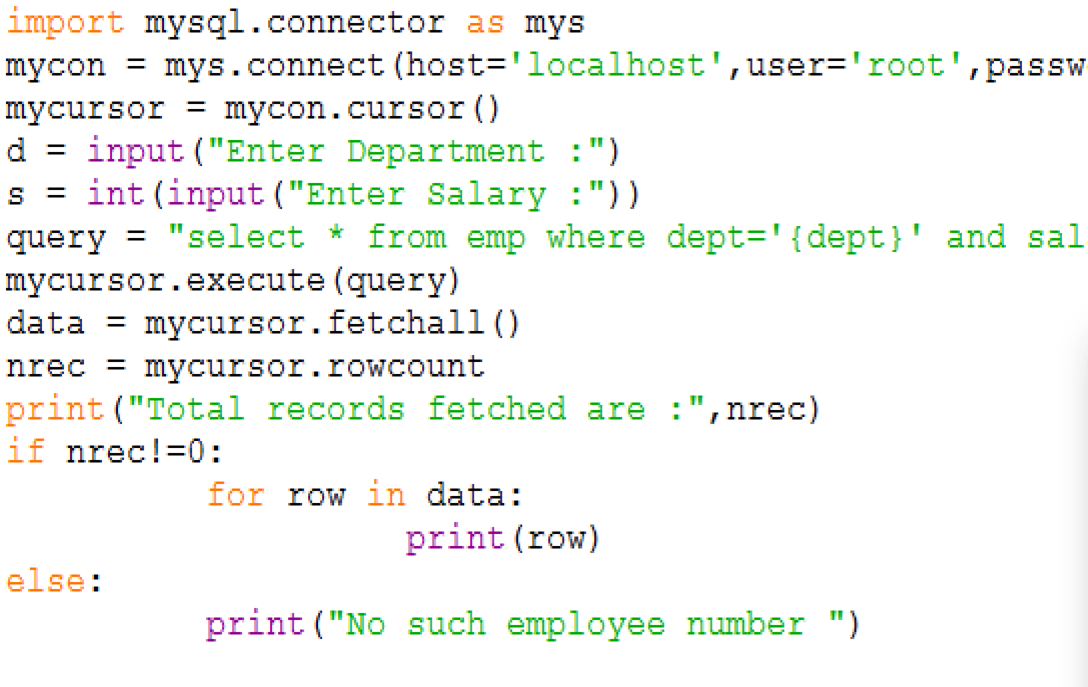


```
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
mycursor = mycon.cursor()
e = int(input("Enter employee number :"))
query = "select * from emp where empno={}".format(e)
mycursor.execute(query)
data = mycursor.fetchone()
if data!=None:
        print(data)
else:
        print("No such employee number ")
```

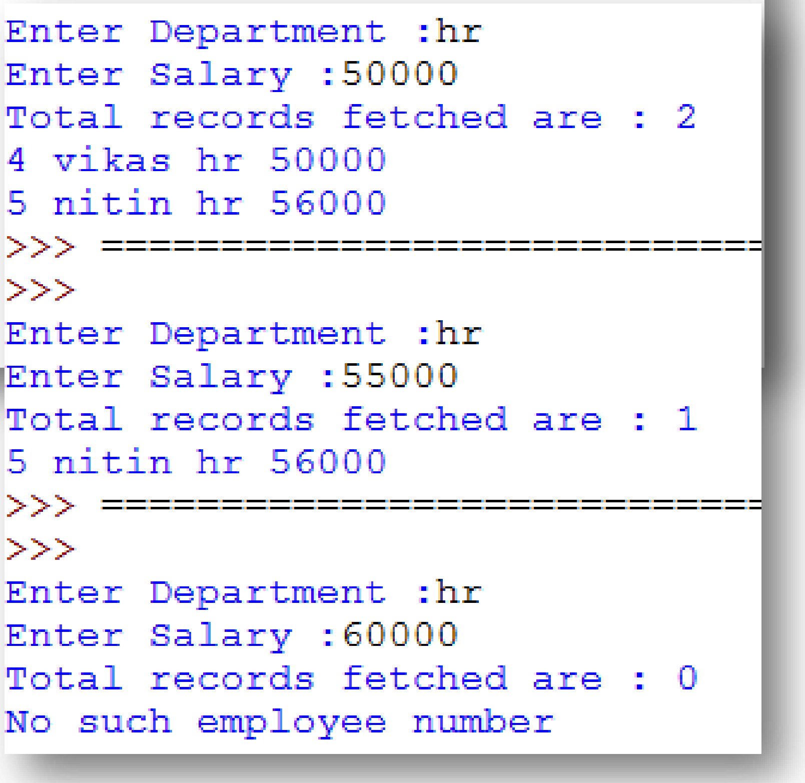


```
Enter employee number :1
1 amit sales 25000
>>> =============================== RESTART ===============
>>>
Enter employee number :8
No such employee number
>>>
```

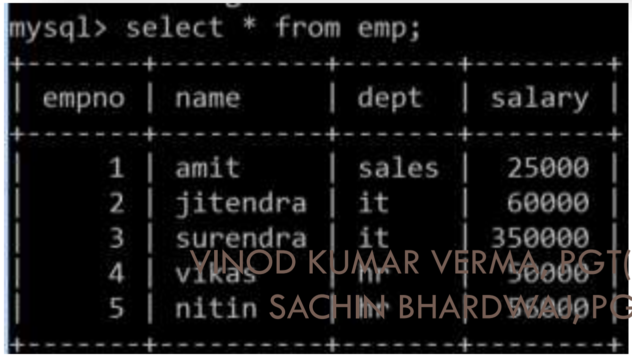
String template with {} and format()



```
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
mycursor = mycon.cursor()
d = input("Enter Department :")
s = int(input("Enter Salary :"))
query = "select * from emp where dept='{dept}' and salary>={salary}".format(salary=s,dept=d)
mycursor.execute(query)
data = mycursor.fetchall()
nrec = mycursor.rowcount
print("Total records fetched are :",nrec)
if nrec!=0:
          for row in data:
                    print(row)
else:
          print("No such employee number ")
```



```
Enter Department :hr
Enter Salary :50000
Total records fetched are : 2
4 vikas hr 50000
5 nitin hr 56000
>>> ============================
>>>
Enter Department :hr
Enter Salary :55000
Total records fetched are : 1
5 nitin hr 56000
>>> ============================
>>>
Enter Department :hr
Enter Salary :60000
Total records fetched are : 0
No such employee number
```

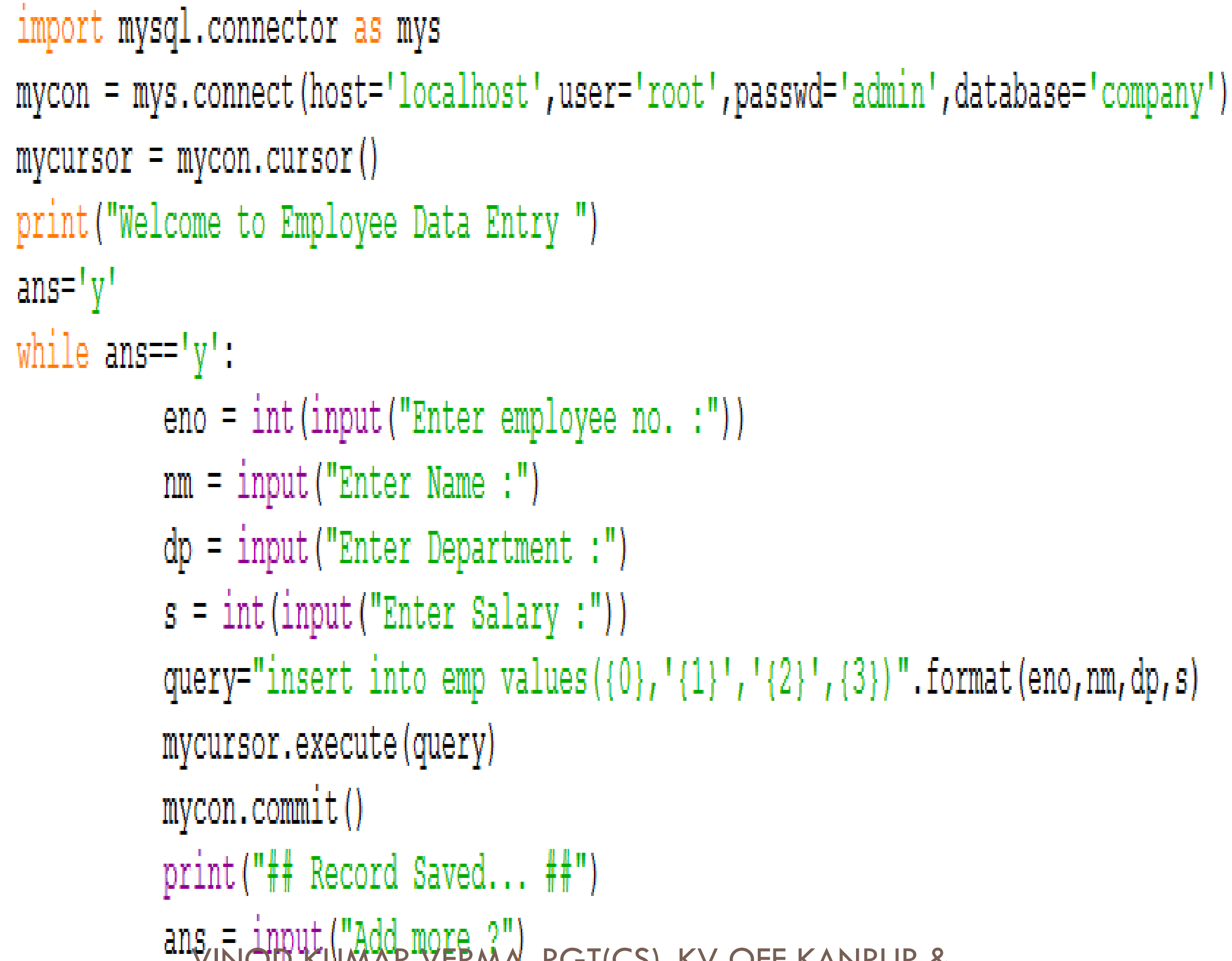


```
mysql> select * from emp;
+-------+-----------+--------+--------+
| empno | name      | dept   | salary |
+-------+-----------+--------+--------+
|     1 | amit      | sales  |  25000 |
|     2 | jitendra  | it     |  60000 |
|     3 | surendra  | it     | 350000 |
|     4 | vikas     | hr     |  50000 |
|     5 | nitin     | hr     |  56000 |
+-------+-----------+--------+--------+
```

Inserting data in MySQL table from Python

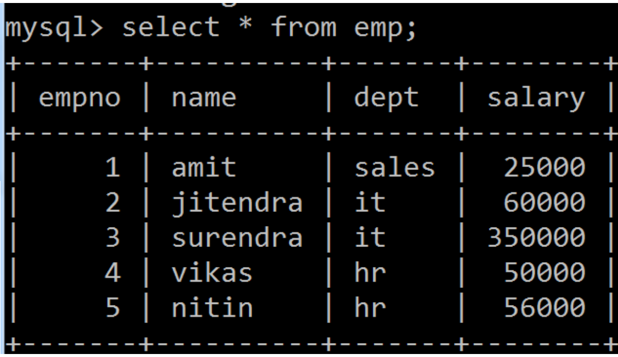
- **INSERT** and **UPDATE** operation are executed in the same way we execute **SELECT** query using **execute**() but one thing to remember, after executing insert or update query we must **commit** our query using **connection object** with **commit**().
- For e.g. (if our connection object name is mycon)
- mycon.commit()

Example : inserting data

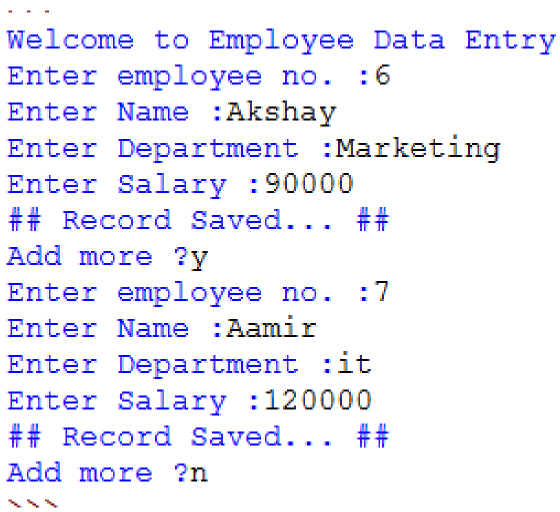


```python
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
mycursor = mycon.cursor()
print("Welcome to Employee Data Entry ")
ans='y'
while ans=='y':
        eno = int(input("Enter employee no. :"))
        nm = input("Enter Name :")
        dp = input("Enter Department :")
        s = int(input("Enter Salary :"))
        query="insert into emp values({0},'{1}','{2}',{3})".format(eno,nm,dp,s)
        mycursor.execute(query)
        mycon.commit()
        print("## Record Saved... ##")
        ans = input("Add more ?")
```

BEFORE PROGRAM EXECUTION

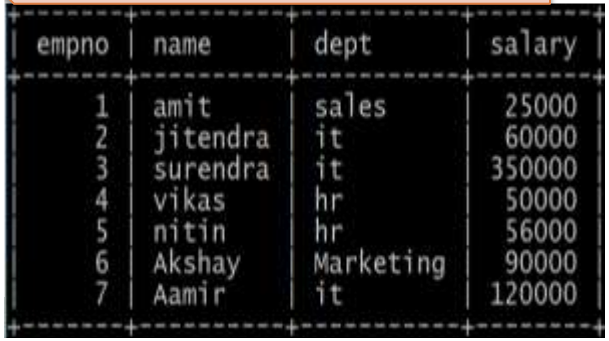


```
mysql> select * from emp;
+-------+----------+-------+--------+
| empno | name     | dept  | salary |
+-------+----------+-------+--------+
|     1 | amit     | sales |  25000 |
|     2 | jitendra | it    |  60000 |
|     3 | surendra | it    | 350000 |
|     4 | vikas    | hr    |  50000 |
|     5 | nitin    | hr    |  56000 |
+-------+----------+-------+--------+
```



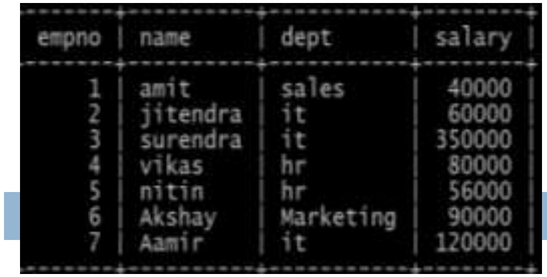
```
Welcome to Employee Data Entry
Enter employee no. :6
Enter Name :Akshay
Enter Department :Marketing
Enter Salary :90000
## Record Saved... ##
Add more ?y
Enter employee no. :7
Enter Name :Aamir
Enter Department :it
Enter Salary :120000
## Record Saved... ##
Add more ?n
```

AFTER PROGRAM EXECUTION



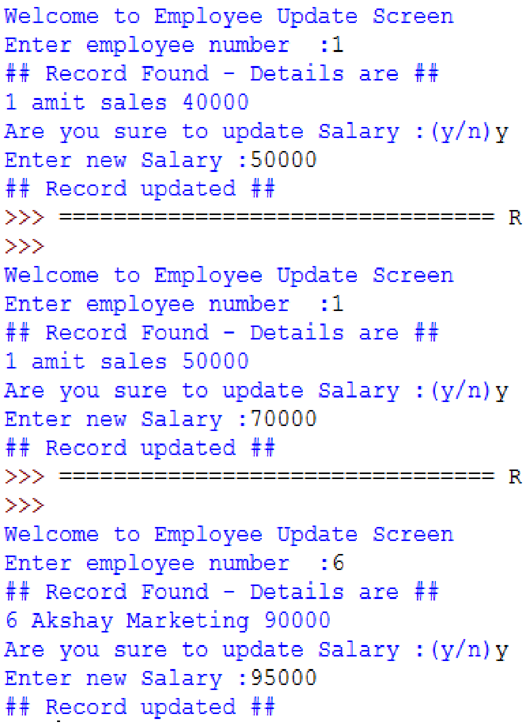
```
+-------+----------+-----------+--------+
| empno | name     | dept      | salary |
+-------+----------+-----------+--------+
|     1 | amit     | sales     |  25000 |
|     2 | jitendra | it        |  60000 |
|     3 | surendra | it        | 350000 |
|     4 | vikas    | hr        |  50000 |
|     5 | nitin    | hr        |  56000 |
|     6 | Akshay   | Marketing |  90000 |
|     7 | Aamir    | it        | 120000 |
+-------+----------+-----------+--------+
```

Example: Updating record

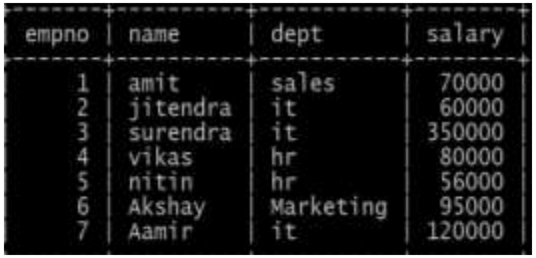


```
+-------+----------+-----------+--------+
| empno | name     | dept      | salary |
+-------+----------+-----------+--------+
|     1 | amit     | sales     |  40000 |
|     2 | jitendra | it        |  60000 |
|     3 | surendra | it        | 350000 |
|     4 | vikas    | hr        |  80000 |
|     5 | nitin    | hr        |  56000 |
|     6 | Akshay   | Marketing |  90000 |
|     7 | Aamir    | it        | 120000 |
+-------+----------+-----------+--------+
```

```python
import mysql.connector as mys
mycon = mys.connect(host='localhost',user='root',passwd='admin',database='company')
mycursor = mycon.cursor()
print("Welcome to Employee Update Screen ")
eno = int(input("Enter employee number  :"))
query = "select * from emp where empno={}".format(eno)
mycursor.execute(query)
data = mycursor.fetchone()
if data!=None:
        print("## Record Found - Details are ##")
        print(data)
        ans = input("Are you sure to update Salary :(y/n)")
        if ans=='y':
                s = int(input("Enter new Salary :"))
                query="update emp set salary={} where empno={}".format(s,eno)
                mycursor.execute(query)
                mycon.commit()
                print("## Record updated ##")
else:
        print("Sorry! No Such Empno exists")
mycon.close()
```



```
Welcome to Employee Update Screen
Enter employee number  :1
## Record Found - Details are ##
1 amit sales 40000
Are you sure to update Salary :(y/n)y
Enter new Salary :50000
## Record updated ##
>>> ================================ R
>>>
Welcome to Employee Update Screen
Enter employee number  :1
## Record Found - Details are ##
1 amit sales 50000
Are you sure to update Salary :(y/n)y
Enter new Salary :70000
## Record updated ##
>>> ================================ R
>>>
Welcome to Employee Update Screen
Enter employee number  :6
## Record Found - Details are ##
6 Akshay Marketing 90000
Are you sure to update Salary :(y/n)y
Enter new Salary :95000
## Record updated ##
```



```
+-------+----------+-----------+--------+
| empno | name     | dept      | salary |
+-------+----------+-----------+--------+
|     1 | amit     | sales     |  70000 |
|     2 | jitendra | it        |  60000 |
|     3 | surendra | it        | 350000 |
|     4 | vikas    | hr        |  80000 |
|     5 | nitin    | hr        |  56000 |
|     6 | Akshay   | Marketing |  95000 |
|     7 | Aamir    | it        | 120000 |
+-------+----------+-----------+--------+
```