

In []:

```
In [ ]: # 1. Establish connection between MySQL and Python.  
        #Create a new database.  
        #Create new table customers with name and address.  
        #Insert values into that table.  
        #Retrieve the values using a query.
```

```
In [1]: import mysql.connector
```

```
In [2]: conn = mysql.connector.connect(user = 'root',host = 'localhost',passwd='utkarsh@14'  
        print(conn)
```

<mysql.connector.connection_cext.CMySQLConnection object at 0x000002A07B5164E0>

```
In [3]: cursor=conn.cursor()
```

```
In [4]: cursor.execute("create database python")
```

```
In [6]: cursor.execute("use python")  
        query= "create table customers(id int primary key auto_increment,name varchar(50),m  
        cursor.execute(query)
```

```
In [10]: query="insert into customers(id,name) values(%s,%s)"  
         cursor.execute(query,(14,"Utkarsh"))
```

```
In [11]: conn.commit()
```

```
In [15]: l = [(80, 'nikhileshwar'), (78, 'nitish')]  
         query = "insert into customers(id, name) values(%s, %s)"  
         cursor.executemany(query, l)  
         conn.commit()
```

```
In [16]: cursor.reset()
```

In []:

In []:

In []:

In []:

```
mysql> show databases;
```

Database
classassignment
classicmodels
company
exam
hospitalmanagement
information_schema
mysql
performance_schema
practise
python
sakila
selfpractise
sys
test
world

```
+-----+
15 rows in set (0.00 sec)
```

```
mysql> desc python;
```

```
ERROR 1046 (3D000): No database selected
```

```
mysql> select * from python;
```

```
ERROR 1046 (3D000): No database selected
```

```
mysql> use python;
```

```
Database changed
```

```
mysql> select * from python;
```

```
ERROR 1146 (42S02): Table 'python.python' doesn't exist
```

```
mysql> select * from student;
```

```
ERROR 1146 (42S02): Table 'python.student' doesn't exist
```

```
mysql> select * from customers;
```

```
Empty set (0.00 sec)
```

```
mysql> select * from customers;
```

id	name	marks
14	Utkarsh	NULL
78	nitish	NULL
80	nikhileshwar	NULL

```
+-----+
3 rows in set (0.00 sec)
```

```
mysql>
```

```
In [1]: import cal
```


```
In [ ]: # 3. Write a python program to implement a simple calculator with a user defined  
module. Your module should have the functions for all arithmetic operations.  
Read the operands from the user.
```

```
In [7]: # main.py  
import cal  
  
# Read the operands from the user  
x = float(input("Enter the first number: "))  
y = float(input("Enter the second number: "))  
  
print("Sum: ", cal.add(x, y))  
print("Difference: ", cal.subtract(x, y))  
print("Product: ", cal.multiply(x, y))  
print("Quotient: ", cal.divide(x, y))
```

```
Sum: 30.0  
Difference: -10.0  
Product: 200.0  
Quotient: 0.5
```

2. Write a python program to implement multiple exceptions with else and Finally statements.

```
try:
    result = 10 / 0
except ZeroDivisionError:
    print("Error: Division by zero.")
except TypeError:
    print("Error: Unsupported operand type.")
else:
    print("The result is:", result)
finally:
    print("Executing the finally clause.")
```

 Error: Division by zero.
Executing the finally clause.

4. Find words in a sentence with more than 4 letters using list comprehension.

```
sentence = "No data is clean, but most is useful.."  
words = [word for word in sentence.split() if len(word) > 4]  
print(words)
```

```
['clean,', 'useful..']
```

5. Find all of the numbers from 1-1000 that are divisible by 7 using list comprehension.

```
num = [i for i in range(1, 1001) if i % 7 == 0]  
print(num)
```

```
[7, 14, 21, 28, 35, 42, 49, 56, 63, 70, 77, 84, 91, 98, 105, 112, 119, 126, 133, 140, 147, 154, 161, 168, 175, 182, 189, 196, 203, 210, 217, 224,
```

6. Write a Python program to sort a tuple by its float element, using lambda function. Sample data: (('item1', '12.20'), ('item2', '15.10'), ('item3', '24.5')) Expected Output: (('item3', '24.5'), ('item2', '15.10'), ('item1', '12.20'))

```
sample = (('item1', '12.20'), ('item2', '15.10'), ('item3', '24.5'))  
data = sorted(sample, key=lambda x: float(x[1]), reverse=True)  
print(data)
```

```
[('item3', '24.5'), ('item2', '15.10'), ('item1', '12.20')]
```

7. Write a python program to read a list of numbers and if an even number is there calculate the square and print the converted list. Use lambda and map functions

```
num = [1, 2, 3, 4, 5, 6]  
sqrt = map(lambda x: x**2 if x % 2 == 0 else x, num)  
print(list(sqrt))
```

```
[1, 4, 3, 16, 5, 36]
```