Q1. Explain why we have to use the Exception class while creating a Custom Exception.

Answer :

We use Exception class for our cusotm exceotion bcoz it is the base class for all the built in error. by inheriting Exception class we can create our own custom exceptions and can handle it by try and except

Q3. What errors are defined in the ArithmeticError class? Explain any two with an example.

Answer :

Arithematic errors are those errors which occur while performing mathematical operations.

for example :

1. ZeroDivisonError :

when a number is divided by zero then ZeroDivisonError occur

2. OverFlow Error :

when the result of any mathematical operation is too large to present then overflow error takes place

In [2]:

```
# For Example
try :
    10/0
except ZeroDivisionError as e:
    print(e)
```

```
division by zero
```

In [ ]:

```
j = 5

try :
    for i in range(1, 1000):
        j = j**i
except ArithmeticError as e:
    print(e)
```

Q4. Why LookupError class is used? Explain with an example KeyError and IndexError.

Answer :

LookupError forms a base class for all the exceptions that are raised when a key or index is not found in the dict or sequence resp.

In [14]:

```
# KeyError
dict_01 = {"harsh" : 95, "rohit" : 90, "haribol" : 100}

try :
    dict_01["krishna"]
except KeyError as e:
    print(e)
```

```
'krishna'
```

In [15]:

```
l = ["harsh", "haribol", "krishna"]
try :
    l[3]
except IndexError as e :
    print(e)
```

```
list index out of range
```

Q5. Explain ImportError. What is ModuleNotFoundError?

In [19]:

```
try :
    import harshit
except ImportError as e :
    print(e)
```

```
No module named 'harshit'
```

ModuleNotFoundError is a subclass of ImportError, ModuleNotFoundError is occur when the path is incorrect or invalid

Q6. List down some best practices for exception handling in python.

Answer :

1. Use always a specific exception

2. write a message with the exception

3. use logging

4. avoid multiple exceptions

5. Document all the error

6. cleaup all the resources