Exception

o To understand the exception firstly we should know about the error in program. Errors in program can be categorized into two types.
  ▪ *Compile Time Errors*
  ▪ *Run Time Errors*

Compile Time Errors:-
    Errors caught during compiled time is called Compile time errors. Compile time errors include library reference, syntax error or incorrect class import.

Run Time Errors:-
    The error that occurs during the run time of program is called run time error.
    They are also known as exceptions. Hence, we can say that exception is a runtime error that occurs because of user's mistake.

Reasons of Exception

Mismatched Input : Suppose that we are entering our name in place of age, causing exception because age is of data type int and name will be string.
File does not exist : Suppose that we are reading a text file easy.txt and that file does not exist in the system, causing exception.
*Divide by zero exception :*When a number is divided by zero then the output will be undefined(infinity).
IndentationError: If incorrect indentation is given.

Exception Handling

The process of handling the exception is called exception handling. There are three main keyword are used to solve the problem of exception.

try block : It is the place where actual code is written and exception occurs. when the code will lead to any error, that error/exception will get caught inside the except block.

except block : except block is intended to catch the error and handle the exception condition. We can have multiple except blocks to handle different types of exception and perform different actions when the exceptions occur.

finally block : This block executes either exception occurs or does not occurs.

Syntax of try-except

```
try:
    #block of statements
except:
    #block of statements


How will it work?
try:
    #run this code
except:
    #this code will run
    #if exception occurs
```

Example
#Program Without exception

```
try:
    a=10
    print("value of x:",a)
except:
    print("An exception occurred")
"""
**Output**
    value of x: 10
"""
```

#Program With exception

```
try:
    #note:here a is not defined
    print("value of x:",a)
except:
    print("An exception occurred")
"""
**Output**
    An exception occurred
"""
```

Syntax of try _ except _ else

```
try:
    #block of statements
except:
    #block of statements
else:
    #block of statements
How will it work?
try:
    #run this code
except:
    #this code will run
    #if exception occurs
else:
    #this code will run
    #if no exception occurs
```

Example

#Example1

```
try:
    a=10
    print("value of x:",a)
except:
    print("An exception occurred")
else:
    print("This is else statement")
"""
**Output**
value of x: 10
    This is else statement
"""
```

#Example 2

```
try:
    a=10
    b=0
    c=a/b
    print("value of c:",c)
except:
    print("Don't put zero in denominator")
else:
    print("This is else statement")
"""
**Output**
Don't put zero in denominator
"""
```

Syntax of try   except   else   finally

```
try:
    #block of statements
except:
    #block of statements
else:
    #block of statements
finally:
    #block of statements
How will it work?
try:
    #run this code
except:
    #this code will run
    #if exception occurs
else:
    #this code will run
    #if no exception occurs
finally:
    #this code will always run
```

Example

#Example 1

```
try:
    a=10
    b=2
    c=a/b
    print("value of c:",c)
except:
    print("Don't put zero in denominator")
else:
    print("This is else statement")
finally:
    print("This is finally block")
"""
**Output**
    value of c: 5.0
    This is else statement
    This is finally block
"""
```

#Example 2

```
try:
    a=10
    b=0
    c=a/b
    print("value of c:",c)
except:
    print("Don't put zero in denominator")
else:
    print("This is else statement")
finally:
    print("This is finally block")
"""
**Output**
    Don't put zero in denominator
    This is finally block
"""
```