

Error / Exception Handling



Error / Exception Handling

In Python Errors can be of two types

- Syntax Errors
- Exceptions

Syntax Errors:

We have already seen that Python has it's own convention of writing code which is known as syntax. These errors are caused by wrong syntax.

Example :

```
a = "Exception"  
if ("e" in a.lower())  
    print("e is present in word")
```

Exceptions:

These errors occur when your syntax is correct but the code resulted in error.

Example

```
number = 45  
div = number/0  
print(div)
```

```
ZeroDivisionError                                Traceback (most recent call last)  
~\AppData\Local\Temp\ipykernel_14372\2336418019.py in <module>  
      1 number = 45  
----> 2 div = number/0  
      3 print(div)  
  
ZeroDivisionError: division by zero
```

How to deal with this?

try ,Except ,finally Statement

try :

The code block which we want to execute is placed inside try block.

except:

Exceptions which we want to handle are given to except statement and logic to handle those exceptions are written inside except block.

else:

If no exceptions are found then code in this block will be executed.

finally:

Code block which is written inside finally block will always execute irrespective of whether we have found any exceptions or not.

try ,Except ,finally Statement

Syntax:

try:

Code in try block

except exception:

Code in except block

else:

Code in else block

finally:

Code which should be always executed