

# Exception Handling

In [ ]: Exections are events that disrupt the normal flow of python execution.

```
In [6]: def main(x, y):
        try:
            z = x / y
            return z
        except Exception as e:
            print(f"Error :{e}")
            return None

        print(main(100,0))
```

Error : division by zero  
None

In [ ]:

```
In [ ]: # Exceptions/errors

        1- synatx error
        2- logical error
```

In [7]: # 1- synatx error

```
for i in range(10)
    print(i)
```

```
Cell In[7], line 3
    for i in range(10)
                ^
SyntaxError: expected ':'
```

In [8]: if a > 3

```
Cell In[8], line 1
    if a > 3
        ^
SyntaxError: expected ':'
```

In [ ]: # Indent error

```
In [9]: for i in range(10):
        print(i)
```

```
Cell In[9], line 2
    print(i)
    ^
IndentationError: expected an indented block after 'for' statement on line 1
```

In [19]: # Logical error

```
def cal_average(a,b):  
    return a + b /2
```

In [20]: `cal_average(5,10)` # BODMAS ---> BODMAS stands for Brackets, Orders (or Of), Div

Out[20]: 10.0

In [13]: `5+ 10`  
`15 /2`

Out[13]: 7.5

In [22]: `# Built-IN Error`

```
print(age)
```

```
-----  
NameError                                Traceback (most recent call last)  
Cell In[22], line 3  
      1 # Built-IN Error  
----> 3 print(age)  
  
NameError: name 'age' is not defined
```

In [23]: `lst = [1,2,3,3,4]`  
`lst[7]`

```
-----  
IndexError                                Traceback (most recent call last)  
Cell In[23], line 2  
      1 lst = [1,2,3,3,4]  
----> 2 lst[7]  
  
IndexError: list index out of range
```

In [26]: `import OS`

```
-----  
ModuleNotFoundError                       Traceback (most recent call last)  
Cell In[26], line 1  
----> 1 import OS  
  
ModuleNotFoundError: No module named 'OS'
```

In [27]: `open('demo.txt','r')`

```

-----
FileNotFoundError                                Traceback (most recent call last)
Cell In[27], line 1
----> 1 open( , )

File ~\AppData\Local\Programs\Python\Python313\Lib\site-packages\IPython\core\interactiveshell.py:326, in _modified_open(file, *args, **kwargs)
    319 if file in {0, 1, 2}:
    320     raise ValueError(
    321         f"IPython won't let you open fd={file} by default "
    322         "as it is likely to crash IPython. If you know what you are doing, "
    323         "you can use builtins' open."
    324     )
--> 326 return io_open(file, *args, **kwargs)

FileNotFoundError: [Errno 2] No such file or directory: 'demo.txt'

```

```

In [28]: try:
        open('demo.txt', 'r')

        except Exception as e:
            print(e)

        finally:
            pass

```

[Errno 2] No such file or directory: 'demo.txt'

```

In [31]: try:
        print(10 + '5')
        except Exception as e:
            print(f" {e}")
        finally:
            pass

```

unsupported operand type(s) for +: 'int' and 'str'  
pass

```

In [33]: # Custom Exceptions

class ValueError(Exception):
    pass

```

```

In [35]: s = int(input("Pass the number less than 10 :"))
        if s < 10:
            raise ValueError("Value is too small")

```

```

-----
ValueError                                Traceback (most recent call last)
Cell In[35], line 3
      1 s = int(input("Pass the number less than 10 :"))
      2 if s < 10:
----> 3     raise ValueError("Value is too small")

ValueError: Value is too small

```

```

In [ ]: # Nested try and except block

```

```
In [ ]: try:
        try:
            z = m1 / m2
        except Exception as e:
            print(e)

    except expection as e:
        print(e)
```