When forced exit occurs using break and return statements

Example:

```
def division(a,b):
    try:
        c=a/b
        return c
    except ZeroDivisionError:
        return 0
    finally:
        print("inside finally block")

def main():
    n1=int(input("enter first number")) # 5
    n2=int(input("enter second number")) # 2
    res=division(n1,n2)
    print(res)

main()
```

Output:

```
enter first number5
enter second number2
inside finally block
2.5
>>>
========== RESTART: C:/Users/user/Desktop/python6pm/py168.py
==========
enter first number5
enter second number0
inside finally block
0
>>>
```

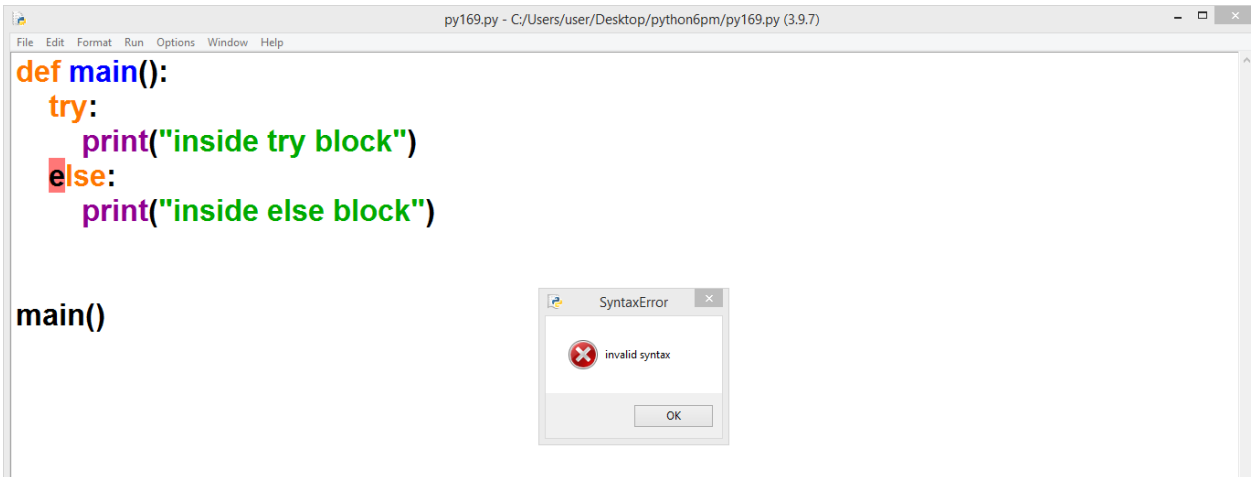
Try..else..except..finally

else block is executed after execution of try block

Syntax:

```
try:
   statement-1
```

```
  statement-2
except <error-type>:
  statement-3
else:
  statement-4
```



We cannot define else block without except block

Example:

```
def main():
    try:
        print("inside try block")
    except:
        print("inside except block")
    else:
        print("inside else block")

main()
```

Output:

```
inside try block
inside else block
```

Example:

```
def main():
    try:
        print("inside try block")
        x=int(input("enter any number"))
```

```
        print(x)
    except:
        print("inside except block")
    else:
        print("inside else block")

main()
```

Output:

```
inside try block
enter any number5
5
inside else block
>>>
============ RESTART: C:/Users/user/Desktop/python6pm/py169.py
===========
inside try block
enter any numberabc
inside except block
>>>
```

The block of code which has to be executed on execution of try block that code is included with else block.

Example:

```
def main():
    try:
        a=int(input("enter first number"))
        b=int(input("enter second number"))
        c=a/b
    except ZeroDivisionError:
        print("cannot divide number with zero")
    except ValueError:
        print("input values must be integers")
    else:
        print(f'result is {c}')

main()
```

Output:

```
enter first number5
enter second number2
result is 2.5
>>>
========== RESTART: C:/Users/user/Desktop/python6pm/py170.py
==========
enter first number5
enter second number0
cannot divide number with zero
>>>
========== RESTART: C:/Users/user/Desktop/python6pm/py170.py
==========
enter first number5
enter second numberabc
input values must be integers
>>>
```

raise keyword

“raise” keyword used for generating error explicitly