- kindly avoid syntax error
- brackets are missing
- quotes are missing
- Name error
  - black color is the variable
  - red color is the string
  - green color keywords
- Attribute error , when we call package

  random.ranm
- indentation error : space

- python is a step by step process
- If we get any error , kernel will not execute the other lines
- Forv example we written 1000lines of code
- There is an error at 500 line
- Then remaining 500 lines will not execute
- But you know that that error not a logical error
- eventhough we got the error we wants to make sure run our code
- Second use case: we need to capture the error also

In [2]:
```
a=eval(input("Enter the number1:"))
b=eval(input("Enter the number2:"))
print("The division of a and b is:",a/b)
print('Hello')
```

```
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
Cell In[2], line 3
      1 a=eval(input("Enter the number1:"))
      2 b=eval(input("Enter the number2:"))
----> 3 print("The division of a and b is:",a/b)
      4 print('Hello')

ZeroDivisionError: division by zero
```

try-except

- try and except both are keywords
- try block has original code
- except block is used to capture the error
- it is similar to if-else
- try-except should come together
- First the code will run in try block, whenever the error occures immediaty it wil redirect to except block

In [4]:
```
try:
    a=eval(input("Enter the number1:"))
    b=eval(input("Enter the number2:"))
    print("The division of a and b is:",a/b)
    print('Hello')

except:
    print("error is coming")
    print("error is zero division error")
```

```
error is coming
error is zero division error
```

In [6]:
```
try:
    a=eval(input("Enter the number1:"))
    b=eval(input("Enter the number2:"))
    print("The division of a and b is:",a/b)
    print('Hello')

except:
    print("error is coming")
    print("error is name error")
```

```
error is coming
error is name error
```

In [ ]:
```
```

In [9]:
```
try:
    a=eval(input("Enter the number1:"))
    b=eval(input("Enter the number2:"))
    print("The division of a and b is:",a/b)
    print('Hello')

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

```
name 'aa' is not defined
```

Case-1: Miss the brackets

In [10]:
```
try:
    a=eval(input("Enter the number1:")
    b=eval(input("Enter the number2:"))
    print("The division of a and b is:",a/b)
```

```
    print('Hello')

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

```
  Cell In[10], line 2
    a=eval(input("Enter the number1:")
          ^
SyntaxError: '(' was never closed
```

Case-2: miss the quotes

In [11]:
```
try:
    a=eval(input("Enter the number1:))
    b=eval(input("Enter the number2:"))
    print("The division of a and b is:",a/b)
    print('Hello')

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

```
  Cell In[11], line 2
    a=eval(input("Enter the number1:))
                 ^
SyntaxError: unterminated string literal (detected at line 2)
```

In [ ]:
```
# do the even-odd program by taking a random number
import random
num=random.randint(1,100)
if num%2==0:
    print(f"The given {num} is even")
else:
    print(f"The given {num} is odd")
```

In [12]:
```
try:
    import random
    num=random.randint(1,100)
    if num%2==0:
        print(f"The given {num} is even")
    else:
        print(f"The given {num} is odd")

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

```
module 'random' has no attribute 'randin'
```

In [ ]:
```
may I know why this * is coming while executing

*  kernel  it is

[]
[*]
[<num>]
```

In [ ]:
```
try:

except:
```

```
finally:
```