try-except

- Generally in real time we will write n number of python lines
- There is a situation we might get an error, that error might be un known
- this unknown error does not cause any harm
- even though we are getting error we need to proceed further
- also we need to catch that error
- Because we alredy known python is a sequential process
- Means the lines will execute step by step
- suppose you have an error at particular line, the code execution will stop at that line only
- but you know that error is not harmful
- so you want to continue the execution
- then we required try-exception

- try- except has two blocks
- try block as actual code
- except block will catch the error

In [2]:
```
n1=eval(input("enter the number1:"))
n2=eval(input("enter the number2:"))
add=n1+n2
print(add)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[2], line 2
      1 n1=eval(input("enter the number1:"))
----> 2 n2=eval(input("enter the number2:"))
      3 add=n1+n2
      4 print(add)

File <string>:1

NameError: name 'ten' is not defined
```

In [4]:
```
try:
    n1=eval(input("enter the number1:"))
    n2=eval(input("enter the number2:"))
    add=n1+n2
    print(add)
except:
    print('hello')
    print('you are getting error')
    print("check the code properly")
```

```
21
```

Note

Generally try except block will use to catch the errors

In [18]:

```
try:
    n1=eval(input("enter the number1:"))
    n2=eval(input("enter the number2:"))
    add=n1+n2
    div=n1/n2
    print(add)
except Exception as e:
    print(f"the error is {e}")
```

```
the error is division by zero
```

In [12]:

```
n1=eval(input("enter the number1:"))
n2=eval(input("enter the number2:"))
add=n1+n2
div=n1/n2
print(add)
```

```
  Cell In[12], line 3
    add=n1+n2
    ^
IndentationError: unexpected indent
```

In [ ]:

```
# Now onwards every code
# need to use try-execpt
```

In [ ]:

```
num=eval(input("enter the number:"))
if num%2==0:
    print(f"the {num} is even")
else:
    print(f"the {num} is odd")
```

In [20]:

```
try:
    num=eval(input("enter the number:"))
    if num%2==0:
        print(f"the {num} is even")
    else:
        print(f"the {num} is odd")
except Exception as e:
    print(e)
```

```
name 'ten' is not defined
```

In [21]:

```
import random
try:
    start=eval(input("enter the start value:"))
    end=eval(input("enter the end value:"))
    num=random.randint(start,end)
    if num%2==0:
        print(f"the {num} is even")
    else:
        print(f"the {num} is odd")
except Exception as e:
    print(e)
```

```
name 'ten' is not defined
```

In [ ]:

```
```