In [2]:

```
#10
try:
    a=int(input('enter a number:'))
    if a%2!=0:
        raise TypeError('odd not allow')
    else:
        print(1/a)
except Exception as er:
    print(er)
```

```
enter a number:4
0.25
```

In [3]:

```
#9th
try:
    a=int(input('enter list size'))
    ls=list()
    for i in range(a):
        ls.append(int(input('Element:')))
    s=int(input('enter index:'))
    if len(ls)>=s:
        print(f'index element={len(ls)}')
    else:
        raise IndexError('error')
except IndexError as er:
    print(er)
except ValueError as er:
    print('no integer input')
finally:
    print('THIS IS THE END')
```

```
enter list size3
Element:1
Element:2
Element:3
enter index:3
index element=3
THIS IS THE END
```

In [4]:

```
#6th
try:
    a=int(input())
    b=int(input())
    if a==b:
        raise ZeroDivisionError("zero divisior")
    print((a+b)/(a-b))
except ValueError as er:
    print('error occured,input again')
except ZeroDivisonError as er:
    print(er)
```

```
3
4
-7.0
```

In [5]:

```
#5
try:
    import cmath
    a=int(input())
    if a==0:
        raise TypeError('not quadratic')
    b=int(input())
    c=int(input())
    d=(b**2)-(4*a*c)
    if d>0:
        root1=(-b-cmath.sqrt(d)/(2*a))
        root2=(-b+cmath.sqrt(d)/(2*a))
    elif d==0:
        root1-(-b/(2*a))
        root2-(-b/(2*a))
    else:
        raise ValueError('no Real Roots')
    print(root1,root2)
except Exception as er:
    print(er)
```

```
1
2
3
no Real Roots
```

In [6]:

```
#2
try:
    import docker
except:
    print('library not imported')
try:
    a={1:'hi',2:'hello',3:'how are you'}
    print(dict[7])
except:
    print('key not found')
try:
    print(ar)
except NameError as ir:
    print(ir)
try:
    p=3/0
except:
    print('ZERO DIVISION ERROR')
```

```
library not imported
key not found
name 'ar' is not defined
ZERO DIVISION ERROR
```

In [8]:

```
#11
try:
    a=int(input())
    b=int(input())
    if b==0:
        raiseZeroDivisionError('denominator cannot be zero')
    else:
        print(a/b)
except Exception as er:
    print(er)
```

```
2
3
0.6666666666666666
```

In [ ]:

```

```