Practical No 8

Aim : Write A Program To Show Error Handling In Python.

1. try – except

```
import sys
while True:
    try:
        n1 = int(input("Enter A Number 1 : "))
        n2 = int(input("Enter A Number 2 : "))
        print("The Divison Is : ",n1/n2)
    except :
        print("There Is An Error",sys.exc_info()[0])
```

```
Enter A Number 1 : 1
Enter A Number 2 : 2
The Divison Is :  0.5
Enter A Number 1 : a
There Is An Error <class 'ValueError'>
Enter A Number 1 : 0
Enter A Number 2 : 2
The Divison Is :  0.0
Enter A Number 1 : 2
Enter A Number 2 : 0
There Is An Error <class 'ZeroDivisionError'>
```

2. Error as Argument

```
def Temp_convert(a):
    try :
        return int(a)
    except ValueError as Arg :
        print("The argument does not Contain numbers \n" , Arg)
```

```
Temp_convert("HEy")
The argument does not Contain numbers
 invalid literal for int() with base 10: 'HEy'
Temp_convert(7)
7
```

3. Custom Error

```
class NotEqualError(Exception):
    pass

n1 = 10
n2 = 9
try :
    if n1!= n2 :
        raise NotEqualError
    else:
        print("Numbers are Equal")
```

```
except NotEqualError:
    print("Numbers are not equal")
```

```
Numbers are not equal
```

4. Finally

```
def Divide(x,y):
    try :
        result = x/y
    except ZeroDivisionError:
        print("Division by Zero")

    else :
        print("result is : ",result)

    finally:
        print("Final clause is executing ")
```

```
Divide(2,0)
Division by Zero
Final clause is executing
Divide(2,1)
result is : 2.0
Final clause is executing
```