Untitled5

March 11, 2023

[1]:

```
a=1
```

[2]:

```
a/0
```

```
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
Cell In[2], line 1
----> 1 a/0

ZeroDivisionError: division by zero
```

[14]:

```
try:
    f=open("test.txt","r")
    f.write("this is my msg")
except Exception as e:
    print("there is some is with my code",e)
else :
    f.close()
    print("this is my print")
finally :
    print("i will always")
a=10
```

```
there is some is with my code not writable
i will always
```

[17]:

```
age=int(input("enter your age"))
```

```
enter your age -233
```

[19]:

```
class validateage(Exception):
    def __init__(self,msg):
        self.msg=msg
```

[2]:

```
def validate_age(age):
    if age<0:
        raise validateage("age should not be lesser than zero")
    elif age >200:
        raise validateage("age is high")
    else:
        print("age is valid")
```

[3]:
```
try:
    age=int(input("enter your age"))
    validate_age(age)
except validate_age as e:
    print(e)
```

```
enter your age 32
```

```
age is valid
```

[14]:
```
try:
    a=10
    10/0
except ZeroDivisionError as e:
    print(e)
```

```
division by zero
```

[15]:
```
try:
    int("sudh")
except (ValueError,TypeError) as e:
    print(e)
```

```
invalid literal for int() with base 10: 'sudh'
```

[16]:
```
try:
    int("sudh")
except :
    print("This will catch my error")
```

```
This will catch my error
```

[17]:
```
try:
    d={1:[1,2,3,4],"kk":"pk"}
    d["key10"]
except KeyError   as e:
    print(e)
```

```
'key10'
```

[18]:
```
try:
    "kishor".test()
```

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

```
'str' object has no attribute 'test'
```

[19]:

```
try:
    l=[1,2,3,4]
    l[10]
except IndexError as e:
    print(e)
```

```
list index out of range
```

[20]:

```
try:
    123+"kk"
except TypeError as e:
    print(e)
```

```
unsupported operand type(s) for +: 'int' and 'str'
```

[22]:

```
try:
    with open("test.txt",'r') as f:
        f.read()
except FileNotFoundError as e:
    print(e)
```

[23]:

```
def test(file):
    try:
        with open(file,'r') as f:
            f.read()
    except Exception as e:
        print("test",e)
    except FileNotFoundError as e:
        print("this is my file not found type error",e)
```

[24]:

```
try:
    a=10
    10/0
except ZeroDivisionError as e:
    print(e)
```

```
division by zero
```

[ ]:

```
```