Python_u_List Of General Use Of Exceptions_(Day 27)_[DATA MINDS]

October 3, 2023

LIST OF GENERAL USE OF EXCEPTIONS

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

```
division by zero
```

```python
[2]: try:
         int("Virat")
     except(ValueError,TypeError)as e:
         print(e)
```

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

```python
[3]: try:
         int("Virat")
     except:
         print("This will catch an error")
```

```
This will catch an error
```

```python
[6]: try:
         import Virat
     except ImportError as e:
         print(e)
     10/5
```

```
No module named 'Virat'
```

```
[6]: 2.0
```

```python
[10]: try:
          d={"key":"Virat",1:[2,3,4]}
          print(d["key1"])
      except KeyError as e:
```

```
    print(e)
```

'key1'

```
[13]: try:
          "Virat".test()
      except AttributeError as e:
          print(e)
```

'str' object has no attribute 'test'

```
[15]: try :
          l=[1,2,3,4,5]
          print(l[8])
      except IndexError as e:
          print(e)
```

list index out of range

```
[16]: try:
          123+"Virat"
      except TypeError as e:
          print(e)
```

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

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

[Errno 2] No such file or directory: 'test.txt'

```
[22]: dir(Exception)
```

```
[22]: ['__cause__',
       '__class__',
       '__context__',
       '__delattr__',
       '__dict__',
       '__dir__',
       '__doc__',
       '__eq__',
       '__format__',
       '__ge__',
       '__getattribute__',
       '__gt__',
```

```
    '__hash__',
    '__init__',
    '__init_subclass__',
    '__le__',
    '__lt__',
    '__ne__',
    '__new__',
    '__reduce__',
    '__reduce_ex__',
    '__repr__',
    '__setattr__',
    '__setstate__',
    '__sizeof__',
    '__str__',
    '__subclasshook__',
    '__suppress_context__',
    '__traceback__',
    'args',
    'with_traceback']
```

[26]:
```python
try:
    with open("test.txt","r") as f:
        test=f.read()
except Exception as e:
    print(e)
except FileNotFoundError as e:
    print("test",e)
```

```
[Errno 2] No such file or directory: 'test.txt'
```

[ ]: