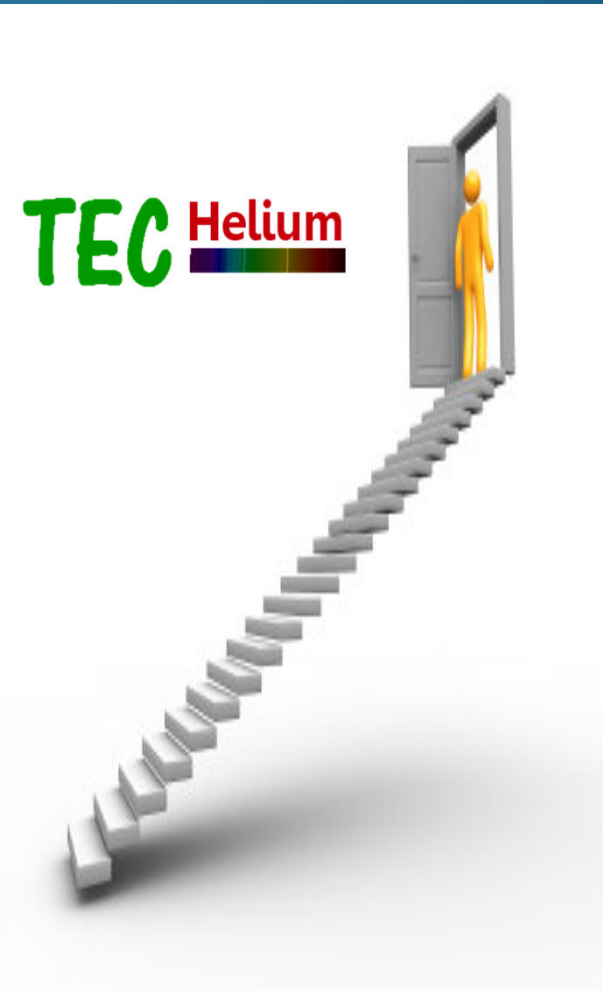


Lets Face it, now

Project Oriented Python

Exception Handling & Programs



Exception

- An exception is an event, which occurs during the execution of a program, that disrupts the normal flow of the program's instructions.
- In general, when a Python script encounters a situation that it can't cope with, it raises an exception.
- An exception is a Python object that represents an error.
- Handling Exception

try:

 You do your operations here;

except *ExceptionI*:

 If there is ExceptionI, then execute this block.

except *ExceptionII*:

 If there is ExceptionII, then execute this block.

else:

 Do this if no exception

Example

```
#!/usr/bin/python
```

```
try:
```

```
    fh = open("testfile", "w")
```

```
    fh.write("This is my test file for exception handling!!")
```

```
except IOError:
```

```
    print "Error: can't find file or read data"
```

```
else:
```

```
    print "Written content in the file successfully"
```

```
    fh.close()
```


The except clause with no exceptions



try:

You do your operations here;

.....

except:

If there is any exception, then execute this block.

.....

else:

If there is no exception then execute this block.

- This kind of a **try-except statement catches all the exceptions that occur.**
- **Not** considered a good programming practice though, because it catches all exceptions but does not identify the root cause of the problem that may occur.

Print Exception

```
def function_exception (file1):  
    try :  
        fa = open(file1,"r")  
    except Exception as ex:  
        print "Exception happened", ex  
  
>> function_exception("hcck")  
Exception happened [Errno 2] No such file or directory: 'hhck'
```

Thanks