

Python - levantando exceções

Você pode gerar exceções de diversas maneiras usando a instrução `raise`. A sintaxe geral para a instrução `raise` é a seguinte -

Sintaxe

```
raise [Exception [, args [, traceback]]]
```

Aqui, `Exception` é o tipo de exceção (por exemplo, `NameError`) e `argument` é um valor para o argumento de exceção. O argumento é opcional; se não for fornecido, o argumento de exceção será `Nenhum`.

O argumento final, `traceback`, também é opcional (e raramente usado na prática) e, se presente, é o objeto `traceback` usado para a exceção.

Exemplo

Uma exceção pode ser uma string, uma classe ou um objeto. A maioria das exceções que o núcleo do Python levanta são classes, com um argumento que é uma instância da classe. Definir novas exceções é bastante fácil e pode ser feito da seguinte forma -

```
def functionName( level ):
    if level <1:
        raise Exception(level)
        # The code below to this would not be executed
        # if we raise the exception
    return level
```

Note - Para capturar uma exceção, uma cláusula "exceto" deve referir-se à mesma exceção lançada como um objeto de classe ou uma string simples. Por exemplo, para capturar a exceção acima, devemos escrever a cláusula `except` da seguinte forma -

```
try:
    Business Logic here...
except Exception as e:
    Exception handling here using e.args...
else:
    Rest of the code here...
```

O exemplo a seguir ilustra o uso de geração de uma exceção -



```
def functionName( level ):
    if level <1:
        raise Exception(level)
        # The code below to this would not be executed
        # if we raise the exception
    return level

try:
    l=functionName(-10)
    print ("level=",l)
except Exception as e:
    print ("error in level argument",e.args[0])
```

Isso produzirá a seguinte **saída** -

```
error in level argument -10
```