

LENGUAJES y HERRAMIENTA PARA CIENCIAS DE DATOS I

Excepciones definidas por el usuario



UCO
ONLINE

Definir una excepción

- Definir una clase que herede de `Exception`

```
1 class NotIntError(Exception):
2     pass
3
4 def comprueba(lista):
5     for value in lista:
6         if not isinstance(value, int):
7             raise NotIntError(value)
8
9
10 lista = (4, 7, 2.11, 9)
11 try:
12     comprueba(lista)
13 except NotIntError as exc:
14     print(f'Numero no entero encontrado: {exc.args[0]}')
```

Mensaje personalizado

- Definir el método `__init__`

```
1 class NotIntError(Exception):
2     def __init__(self, message='Numero no entero encontrado'):
3         super().__init__(message)
4
5 def comprueba(lista):
6
7     for value in lista:
8         if not isinstance(value, int):
9             raise NotIntError
10
11 lista = (4, 7, 2.11, 9)
12 try:
13     comprueba(lista)
14 except NotIntError as exc:
15     print(exc)
```

Mensaje personalizado

- Añadir más parámetros a `__init__`
- Redefinir el método `__str__`

```
1 class NotIntError(Exception):
2     def __init__(self, value, message='Numero no entero encontrado'):
3         self.value = value
4         self.message = message
5         super().__init__(message)
6
7     def __str__(self):
8         return f'{self.message}: {self.value}'
```

