Tutorial de Excepciones

Las excepciones son la herramienta que implementa Python para manejar los errores potenciales de un programa. Cuando una porción de código quiere indicar que ocurrió un error, se dice que debe lanzar una excepción, mientras que, cuando otra quiere saber si surgió un error y actuar en consecuencia, se dice que debe capturarla dicha excepción.

Es importante destacar que, si una excepción es lanzada y ningún código la captura, el programa finaliza de manera automática.

Un ejemplo podría ser la función: int().

```
int("20")
20
```

Esta función lanza una excepción cuando el argumento pasado no puede ser representado como un número entero. Tal como hemos visto en el curso.

Como se observa, las excepciones están generalmente acompañadas de un mensaje que nos indica qué salió mal y en dónde. Consideremos ahora un ejemplo más real de aplicación:

```
edad = int(input("Escribe tu edad: "))

if edad >= 18:
    print("Eres un adulto.")

else:
    print("Aún no eres un adulto.")
```

Aquí pronto vemos cuál es el problema. Si el usuario escribe cualquier cosa que no sea un número en la consola, el programa arrojará una excepción y terminará. Para resolver este inconveniente, necesitamos capturar la excepción, lo cual haremos con las palabras reservadas try y except.



```
while True:
    try:
        edad = int(input("Escribe tu edad: "))
        break
    except ValueError:
        print("¡Debes ingresar un valor numérico!")

if edad >= 18:
    print("Eres un adulto.")
else:
    print("Aún no eres un adulto.")
```

Salida por consola:

Escribe tu edad: cua ¡Debes ingresar un v	•
Escribe tu edad:	

Analicemos en mayor detalle el código anterior. Primero hemos incluido un bucle while, que se ejecuta infinitamente mientras la condición sea verdadera (True). Esto nos permitirá que el programa siga preguntando la edad hasta que el usuario ingrese un número.

Para ello "envolvimos" las llamadas a int() e input() dentro de un bloque de código try/except. Siguiendo la lógica que se detalla a continuación: cuando una excepción es capturada, la ejecución del código salta desde donde haya ocurrido (la llamada a int() al bloque de código dentro de la cláusula except.

Así, cuando int() lanza ValueError, el código salta a la llamada a print() sin llegar a ejecutar la palabra reservada break, la cual da término al bucle. Por esta razón el proceso se repite siempre que ocurra una excepción del tipo ValueError. Ahora bien, si la llamada a int() es exitosa, break llega a ejecutarse y el código continúa en el condicional.

Existen otros tipos de Excepciones, por ejemplo:

TypeError, KeyError, IndexError, NameError, RuntimeError, ZeroDivisionError

Por ejemplo, IndexError es lanzada cuando intentamos acceder a un elemento de una lista o tupla que está por fuera de sus límites.

```
---> 2 lenguajes[5]
```

IndexError: list index out of range

Un mismo código puede lanzar múltiples excepciones. Por ejemplo, int(), como hemos visto, lanza ValueError cuando el argumento no puede ser convertido a un número entero, pero también TypeError cuando el argumento no es una cadena.

```
TypeError
call last)
Input In [4], in <cell line: 1>()
----> 1 int([2, 4])
TypeError: int() argument must be a string, a bytes-like object or a number, not 'list'
```

