Cómo ejecuto los ejemplos?

Para poder ejecutar los scripts, necesitas tener Python instalado en tu computador, abrir una consola (puede ser powershell, cmd, o cualquier terminal que dispongas), posicionarte en la carpeta donde tengas los ejercicios y ejecutar lo siguiente:

```
python ejercicio_01.py
```

Acorde al nombre del ejercicio que quieres probar. Verás en la pantalla el resultado o la interacción del script que estás ejecutando.

Por otro lado, también puedes usar Replit como lo hicimos en clase, copiar y pegar los scripts y ejecutarlos en `Run`.

Qué pasaría si...?

A fin de poder realizar código a prueba de errores, el programador debe pensar regularmente que el usuario podría usar mal su código (siendo esto más común de lo que pueda creerse...) así que necesitará gestionar esos errores de alguna manera (es decir, que aunque el usuario haga mal las cosas, el programa sea amigable con el).

Bajo esta premisa, qué pasaría si en el ejercicio 04 el usuario colocara un número con decimales? o si en vez de un número, te pasa una letra o un string que no pueda convertirse explícitamente a número entero? Lo mismo con el ejercicio 5 al preguntar su edad.

La respuesta es clara, el código se rompería, puesto que una de las condiciones de la funciónde converción de tipos int() requiere que su parámetro de entrada sea un elemento que pueda fácilmente convertir a número entero. En este caso, para evitar las aterradoras pantallas de error en Python, podemos "atrapar" el error y dar un comportamiento diferente al script en estos casos con dos keywords propias de Python (pero que cada lenguaje tiene sus propias versiones): try/except.

Explícitamente, su funcionamiento "intenta" realizar una porción de código (try) y ante cualquier error u excepción ejecuta una respuesta diferente (except). La sintaxis para esto sería:

```
try:
    user_try = int(input("Dame un numero: "))
except:
    print("No me has dado un numero valido :(")
```

Esto atrapará cualquier error, sin embargo, es recomendable atrapar todos los errores que puedan ocurrir explícitamente y dar una respuesta a cada uno. Por ejemplo, si ejecutas el código sin try/except, el error que verías en consola sería: ValueError: bla bla bla...

De esta forma, podemos dar una excepción única para este error, tal como sigue:

```
try:
    user_try = int(input("Dame un numero: "))

except ValueError:
    print("No me has dado un numero valido :(")

except:
    print("Esto saltará ante cualquier otro error.")
```

Existe una larga lista de Excepciones en Python, e incluso tú puedes crear tus propias Excepciones (aunque esto es un poco más avanzado). Te recomiendo investigar un poco al respecto, a ver cuantas puedes encontrar (una pista, si en vez de continuar con un script aprietas ctrl + c, verás una excepción con un nombre único también!)