

UNIVERSIDAD SAN PABLO DE GUATEMALA

Facultad de Ciencias Empresariales

Escuela de Ingeniería

Ingeniería en Ciencias y Sistemas de la Computación



MANEJO DE ERRORES Y EXCEPCIONES

Trabajo presentado en el curso de Programación II
Impartido por Lic. Mynor Alfonso López De León.

Yellsmy Lilibeth Toj García

2000579

Guatemala, 9 de septiembre de 2021.

MANEJO DE ERRORES Y EXCEPCIONES

Hay (al menos) dos tipos diferentes de errores: errores de sintaxis y excepciones.

1. Errores de sintaxis:

Son errores de interpretación que, al momento de ejecutar el código, inmediatamente marca el error.

2. Excepciones: Cuando una declaración o expresión es sintácticamente correcta puede detectar ciertos errores durante su ejecución, estos errores se llaman excepciones.

	Utilidad	Diferencias
Errores	Es muy útil saber que tenemos un error en la sintaxis y en Python siempre nos mostrará la (o las) línea/s en las que estamos equivocados sintácticamente inmediatamente corramos el código.	Estos errores únicamente son de sintaxis y se manifiestan al inicio de la ejecución del programa.
Excepciones	El manejo de las excepciones es de vital importancia ya que al presentar un error de estos, el resto del código no se ejecutará y esto sería catastrófico cuando ejecutamos un programa con cientos o miles de líneas de código, al saber manejar las excepciones hacemos que aunque se presente un error de estos, el programa continúe con su curso hasta terminar.	Estos errores, a diferencia de los de sintaxis, se manifiestan durante la ejecución de un programa y no al inicio (a menos que el error esté en las primeras líneas de código).

Ejemplos

Error de sintaxis:

```
while True
    print("Feliz día")
    break
```

```
File "c:\Users\Lenovo5\Desktop\UNIVERSIDAD SAN PABLO DE GUATEMALA\PROGRAMACIÓN II\ERRORES Y EXCEPCIONES.py", line 1
    while True
        ^
SyntaxError: invalid syntax
```

Da error de sintaxis porque faltan los dos puntos ":" después del while True.

Excepciones:

```
operacion = input("Ingrese operación: ")
if operacion == "suma":
    suma = 4 + num1
    print(suma)
Traceback (most recent call last):
  File "c:\Users\Lenovo5\Desktop\UNIVERSIDAD SAN PABLO DE GUATEMALA\PROGRAMACIÓN II\ERRORES Y EXCEPCIONES.py", line 3, in <module>
    suma = 4 + num1
NameError: name 'num1' is not defined
```

Esto debido a que "num1" es una variable no definida.

Manejo:

Para evitar que el programa se detenga cuando encuentra una excepción debemos colocar el bloque de código donde se podría generar la misma a continuación de la sentencia try. El programa intentará (try) ejecutar dicho código y si no puede hacerlo, atrapará el error y responderá como lo indiquemos a continuación de la(s) sentencia(s) except.

```
operacion = input("Ingrese operación: ")
if operacion == "suma":
    try:
        suma = 4 + num1
        print(suma)
    except NameError:
        print("Error, No ha definido la variable que está utilizando")
print("Operación terminada")
PS C:\Users\Lenovo5\Desktop\UNIVERSIDAD SAN PABLO DE GUATEMALA\PROGRAMACIÓN II> & C:/Users/Lenovo5/AppData/Local/Programs/Python/Python39/python.exe "c:/Users/Lenovo5/Desktop/UNIVERSIDAD SAN PABLO DE GUATEMALA/PROGRAMACIÓN II/ERRORES Y EXCEPCIONES.py"
```

```
Ingrese operación: suma
Error, No ha definido la variable que está utilizando
Operación terminada
```

Aplicación al código en grupo (Jugadores)

La variable “edad” en la que el jugador debe ingresar su edad al momento de registrarse, es de tipo entero “int” por lo tanto, al momento de ingresar la edad con letras da un error de tipo “ValueError”, para que esto no afecte el resto del código se está usando try y except para manejar este error, y que al momento de escribir la edad con letra, da el mensaje “Error, su edad debe ser en números y no letras, inténtelo nuevamente”, e inicia el registro nuevamente.

```
while True:
    try:
        estado = "Activa"
        nombre = input("Ingrese nombre: ")
        edad = int(input("Ingrese su edad: "))
        nacionalidad = input("Ingrese su nacionalidad: ")
        apodo = input("Ingrese su apodo: ")
        contraseña = input("Cree su contraseña: ")
        repit_contraseña = input("Confirme su contraseña: ")
        if contraseña != repit_contraseña:
            print("La contraseña no coincide, intente nuevamente")
        contraseña = input("Cree su contraseña: ")
        repit_contraseña = input("Confirme su contraseña: ")
        if contraseña != repit_contraseña:
            print("Error al crear cuenta")
            print("Las contraseñas no coinciden nuevamente")
            print("Reinicie su registro")
            print("Elija la opción Crear cuenta nuevamente")

        Datos += nombre + "|"
        Datos += apodo + "|"
        Datos += nacionalidad + "|"
        Datos += str(edad) + "|"
        Datos += contraseña + "|"
        Datos += estado + "\n"
        self.cargarUsuario(nombre, nacionalidad, contraseña, apodo, edad, estado)
        break
    except ValueError:
        print("Error, su edad debe ser en números y no en letras, inténtelo nuevamente")
```

```
Ingrese nombre: María
Ingrese su edad: veinticinco
```

```
Error, su edad debe ser en números y no en letras, inténtelo nuevamente  
Ingrese nombre: María  
Ingrese su edad: 25  
Ingrese su nacionalidad: Mexicana  
Ingrese su apodo: De los ángeles  
Cree su contraseña: 777  
Confirme su contraseña: 777
```