

Universidad Nacional del Altiplano
Facultad de Ingeniería Estadística e Informática

Docente:

Ing. Coyla Idme Leonel

Alumno:

Ticona Miramira Roberto Angel

Manejo de excepciones

El manejo de excepciones en la POO en python es un mecanismo que permite a los objetos y métodos identificar errores en tiempo de ejecución, interrumpir el flujo normal del programa y ejecutar bloques específicos de código como: **try except, else y finally**. Para tratarlos de manera controlada, garantizando la estabilidad y confiabilidad del sistema.

Ejemplos

1.- Crear la clase Division y probar las excepciones para la división de números entre 0.

» CÓDIGO

```
1 class Division:
2     def __init__(self, a, b):
3         self.a = a
4         self.b = b
5
6     def dividir(self):
7         try:
8             resultado = self.a / self.b
9             return resultado
10        except ZeroDivisionError:
11            return "Error: No se puede dividir entre cero."
12        except Exception as e:
13            return f"Ocurrió un error: {e}"
14        finally:
15            print("Operación finalizada.")
16
17 operacion = Division(10, 0)
18 print(operacion.dividir())
```

» EJECUCIÓN

```
1 Operación finalizada.
2 Error: No se puede dividir entre cero.
```

2.- Crear un programa que calcule la hipotenusa de un triángulo rectángulo, utilizando excepciones.

» CÓDIGO

```
1 import math
2
3 def calcular_hipotenusa(cateto_a, cateto_b):
4     return math.sqrt(cateto_a ** 2 + cateto_b ** 2)
5
6 def main():
7     try:
```

```
8     a = float(input("Ingrese el valor del cateto a: "))
9     b = float(input("Ingrese el valor del cateto b: "))
10    if a <= 0 or b <= 0:
11        raise ValueError("Los catetos deben ser números positivos")
12    hipotenusa = calcular_hipotenusa(a, b)
13    print(f"La hipotenusa es: {hipotenusa:.2f}")
14    except ValueError as ve:
15        print("Error", ve)
16    except Exception as e:
17        print("Ocurrió un error inesperado", e)
18
19 if __name__=="__main__":
20     main()
```

» EJECUCIÓN

```
1 Ingrese el valor del cateto a: -3
2 Ingrese el valor del cateto b: -1
3 Error Los catetos deben ser números positivos
```

3.- Crear la secuencia de fibonacci, utilizando excepciones.

» CÓDIGO

```
1 def fibonacci(n):
2     a = 0
3     b = 1
4     for i in range (n):
5         print(a)
6         c = a + b
7         b = a
8         a = c
9
10 def main():
11     try:
12         n = int(input("Ingrese el valor de n: "))
13         if n <= 0:
14             raise ValueError("El valor de n tiene que ser positivo")
15         fibonacci(n)
16     except ValueError as ve:
17         print("Error", ve)
18     except Exception as e:
19         print("Ocurrió un error inesperado", e)
20
21 if __name__=="__main__":
22     main()
```

» EJECUCIÓN

```
1 Ingrese el valor de n: -5
2 Error El valor de n tiene que ser positivo
```