

**UNIVERSIDAD DR. ANDRÉS BELLO**  
**REGIONAL CHALATENANGO**  
**FACULTAD DE TECNOLOGÍA E INNOVACIÓN**



**CARRERA:** INGENIERÍA EN SISTEMAS Y COMPUTACIÓN.

**DOCENTE:** ING. JONATHAN FRANCISCO CARBALLO CASTRO.

**ASIGNATURA:** FUNDAMENTOS DE PROGRAMACIÓN.

**CONTENIDO:** ACTIVIDAD ASÍNCRONA S-15.

**ALUMNOS:** WENDY MARISOL ESPINOZA RIVAS.  
KENIA GUADALUPE FIGUEROA ORTEGA.  
ROMILIO DAVID GALDÁMEZ ESCOBAR.  
AMANDA YURIDIA PORTILLO MARROQUÍN.  
BRANDON JOSUÉ RIVERA ALEMÁN.

**CICLO:** 01-23

**Chalatenango, 13 de mayo de 2023**

## INDICACIÓN:

Hacer una comparativa entre las versiones de Python 2 vs Python 3.

- Diferencia
- Correcciones (errores)

Se presenta un cuadro con las diferencias entre Python 2 y Python 3.

	PYTHON 2	PYTHON 3
<b>DIFERENCIAS</b>	<ul style="list-style-type: none"><li>- Retorna un entero al dividir dos enteros, incluso si el resultado es un número decimal.</li><li>- Usa la palabra clave "print" como una sentencia.</li><li>- Usa la sintaxis "except Exception, e:" para manejar excepciones.</li><li>- Las cadenas Unicode deben tener una "u" antes de la cadena.</li><li>- Algunas bibliotecas importantes no son compatibles con Python 2.</li><li>- Las funciones lambda solo pueden tener una expresión.</li><li>- No se pueden definir argumentos de palabras clave únicamente.</li><li>- Usa la función "xrange()" para generar rangos.</li><li>- No hay soporte para operaciones de matriz.</li><li>- No hay soporte para operaciones de matemáticas complejas.</li><li>- No hay soporte para asyncio.</li></ul>	<ul style="list-style-type: none"><li>- La división de dos enteros ahora retorna un número decimal, a menos que se use el operador de división entera (/).</li><li>- "Imprimir" es una función, lo que significa que debe usar la sintaxis de función para imprimir.</li><li>- Usa la sintaxis "except Exception as e:" para manejar excepciones.</li><li>- Las cadenas Unicode son la forma predeterminada y no necesitan un prefijo especial.</li><li>- Las bibliotecas se han actualizado para funcionar con Python 3.</li><li>- Las funciones lambda pueden tener múltiples expresiones y se pueden usar como funciones regulares.</li><li>- Los argumentos de palabras clave se pueden definir sin valores predeterminados.</li><li>- Usa la función "range()" para generar rangos.</li><li>- Soporte para operaciones de matriz agregada.</li><li>- Soporte para operaciones de matemáticas complejas agregadas.</li><li>- Soporte para asyncio agregado.</li></ul>

**En cuanto a las correcciones de errores, Python 3 incluye una serie de correcciones y mejoras con respecto a Python 2. Estas incluyen:**

- Una implementación de la división, que evita la división entera cuando no es apropiada y devuelve un resultado flotante.
- El manejo de cadenas de caracteres se ha mejorado significativamente, con el uso de Unicode como formato de cadena predeterminado.
- Se ha mejorado la seguridad y estabilidad de la biblioteca estándar, con la inclusión de nuevas funciones y la eliminación de funciones obsoletas o peligrosas.
- Se han mejorado las funciones de manejo de excepciones, permitiendo un manejo más preciso de las excepciones y una mejor depuración.
- Se han mejorado las funciones de manejo de archivos, haciendo que sea más fácil abrir y cerrar archivos y trabajar con ellos en general.
- Se ha mejorado la generación de números aleatorios, haciendo que sea más fácil generar números aleatorios de manera segura y controlada.