

Checkpoint 3 – Preguntas teóricas:

- **¿Cuáles son los tipos de Datos en Python?**

Los tipos de datos en Python son booleanos, números, strings, bytes y bytes arrays, none, listas, tuplas, sets y diccionarios.

- Los booleanos toman valores True o False.
- Los números se dividen en tres categorías, enteros (-1, 2, 3), flotantes (1.2, 3.4, -0.001) y complejos (1j, 3+4j)
- Los string almacenan texto, se delimitan por comilla simples o dobles, “hola” “”hola mundo”, pueden contener frases completas o párrafos, un número con comillas se considerará un string.
- Los bytes son secuencias inmutables de números enteros (0-255) Se usan para datos binarios, archivos y redes.
- Los bytes arrays es un byte que tiene carácter mutable, se usan como datos binarios en procesos.
- None representa la ausencia de valor, puede ser el valor por defecto antes de que se introduzca un dato.
- Las listas son secuencias ordenadas y mutables de elementos, se representan mediante corchetes []. Ejemplo: ordenador = ["cpu", "pantalla", "teclado"].
- Las tuplas son secuencias ordenadas e inmutable de elementos, se representan mediante paréntesis (). Ejemplo: ordenador = ("cpu", "pantalla", "teclado").
- Los sets son secuencias no ordenadas y sin duplicados, son de gran utilidad en índices u otros valores que no admiten duplicados. Se representan con llaves {}. Ejemplo: ordenador = {"cpu", "pantalla", "teclado"}.
- Diccionarios: Almacenan datos estructurados asociando diferentes elementos, también se representan con llaves, pero se usa dos puntos para asociar los elementos. Ejemplo: ordenador = {"cpu": "Intel", "pantalla": "Philips", "teclado": "102 teclas"}.

- **¿Qué tipo de convención de nomenclatura deberíamos utilizar para las variables en Python?**

Existen varias clases de nomenclatura, las más habituales son:

- kebab-case - las partes de la variable se separan con guiones (my-super-var)
- snake_case - se utiliza un guión bajo para separar (my_super_var)
- CamelCase - cada palabra en la variable se escribe con mayúscula (MySuperVar)

Para Python lo habitual es nombrarla usando el método snake_case siguiendo la recomendación PEP8 <https://peps.python.org/pep-0008/#naming-conventions>

- **¿Qué es un Heredoc en Python?**

Heredoc (abreviatura de "documento aquí") es una forma de representar un bloque de texto en código, generalmente en lenguajes de programación. Le permite definir cadenas multilínea, ya sean párrafos, bloques de código, etc.

Mejoran la legibilidad, tanto del código como del output, y minimizan los errores. En algunas ocasiones también se utilizan como sustitutos a los comentarios, aunque no se trata de una buena práctica.

- **¿Qué es una interpolación de cadenas?**

La interpolación de cadenas es la inserción de elementos como variables o valores dentro de un texto, la forma de hacerlo es anteponiendo una `f` al string y rodear los elementos con llaves, de forma que el resultado sería similar a esto:

Concatenación = `f"texto {elemento}"`

- **¿Cuándo deberíamos usar comentarios en Python?**

Deberíamos usar comentarios cuando ayuden a que el código sea más legible, bien explicando partes del código, poniendo ejemplos, dividan áreas o sirvan para la depuración.

- **¿Cuáles son las diferencias entre aplicaciones monolíticas y de microservicios?**

La aplicación monolítica consiste en un desarrollo en un único programa y en una única plataforma. Los cambios en el código implican desplegar de nuevo la aplicación.

Los micro servicios consisten en desarrollos independientes que se comunican entre sí, se pueden realizar despliegues independientes y redundantes, lo que la convierte en una infraestructura más ágil que la monolítica