

tipos de Datos en Python

En python hay varios tipos de datos cada uno tiene su funcionalidad y representa un tipo de informacion a continuacion redactare algunos de ellos con sus ejemplos:

string o cadena: es un valor de texto que se le da a una variable y su sintaxis es la siguiente
variable = 'string' o "string"

Enteros (int): Representan números enteros sin decimales

Flotantes (float): Representan números decimales

Booleanos (bool): Representan valores de verdad y falso es decir, verdadero (True) o falso (False).

Listas: Son colecciones ordenadas y modificables de elementos. Los elementos de una lista pueden ser de diferentes tipos y se acceden mediante índices. Por ejemplo, [11, 2, 54, 85], ['g', 'h', 'c'], [1, 'dos', True], y se puede poner listas dentro de otra lista Por ejemplo [1, 2, 3, ['adam', 'ali', 5], True].

Diccionarios: Son colecciones no ordenadas de pares clave-valor. Cada elemento del diccionario tiene una clave única asociada a un valor. Los valores se acceden mediante sus claves. Por ejemplo, {'nombre': 'Juan', 'edad': 30, 'ciudad': 'Madrid'}.

Tuplas: Son colecciones ordenadas e inmutables de elementos. Una vez creada, una tupla no puede modificarse. Se accede a los elementos de una tupla mediante índices. Por ejemplo, (1, 2, 3), ('a', 'b', 'c'), (1, 'dos', True), etc.

tipo de convención de nomenclatura deberíamos utilizar para las variables en Python

La comunidad suele seguir la convención de nomenclatura llamada snake_case para nombrar variables. En esta convención, todas las letras son minúsculas y las palabras se separan con guiones bajos y evitar palabras predefinidas para metodos o funciones..... Por ejemplo:

```
name_hijo = adam
precio_salida = 18
```

un Heredoc en Python

Un "Heredoc" es concepto en si no existe en python pero hay una forma de definir cadenas de texto multilinea utilizando esta syntaxis ''' o """ asi que todo lo que dentro se concidera como parte de la cadena

una interpolación de cadenas

La interpolación de cadenas es el proceso de insertar valores de variables dentro de una cadena de texto. Esto te permite crear cadenas dinámicas que incluyan información específica en ciertos puntos.

Por ejemplo el ejercicio 7

```
numero = 8
new_variable = f" mi hijo tiene {numero} años"
print(new_variable)
```

comentarios en Python

En Python, los comentarios son utilizados para documentar y explicar el código.

Proporcionar advertencias o notas importantes

Marcar código para futuras mejoras o cambios

Explicar el propósito de una sección de código

pero estoy de acuerdo con el mentor de que no hay que abusar ya no cambian cuando se ahce cambio en el codigo lo que puede hacer que el codigo sea muy confuso.

las diferencias entre aplicaciones monolíticas y de microservicios

las aplicaciones monolíticas son aquellas en las que todo el código se desarrolla, compila y despliega como una sola unidad.

las aplicaciones de microservicios se dividen en servicios independientes que se desarrollan, compilan y despliegan de forma independiente

	monolíticas	microservicios
Arquitectura	Todo en una sola unidad	Dividida en servicios independientes
Lenguajes de Programación	Utiliza las mismas tecnologías y lenguajes	Puede usar diferentes tecnologías y lenguajes para cada servicio
Tolerancia a Fallos	Un fallo afecta a toda la aplicación	Un fallo en un servicio no afecta a otros.