

Checkpoint 3

Mohamed ALWASSY ALWASY

¿Cuáles son los tipos de Datos en Python?

En python hay diferentes tipos de datos entre ellos:

str: se usa para declarar texto. Ejemplo: `saludo = 'Hola Mundo'`.

Int: es para declarar números enteros. Ejemplo: `numero_cualquiera = 1234`

Float: es para declarar números con decimales. Ejemplo: `decimal = 1.234`

Boolean: dispone de dos valores lógicos True y False. Ejemplo: `condicion_1 = True`

List: se usa para declarar más de un valor que pueden ser del mismo tipo por ejemplo esta es una lista que solo lleva strings (`any = ['hola', 'adios', 'buenos dias']`), o puede ser una lista que lleva diferentes tipos de datos ejemplo: `any = [123, 'hola', False]`

Tuples: que son como listas pero no se pueden modificar. Ejemplo: `any_touple = ('item_1', 'item_2'.....)`

Diccionarios: se usan para un amplio número de datos, y funcionan con clave-valor y eso quiere decir que el valor del elemento dentro del diccionario se obtiene mediante su clave. Ejemplo : `dic = {'nombre': 'Juan', 'edad': 30}`.

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

En python hay que respetar una serie de reglas en las que se incluye:

Utilizar nombres descriptivos: Los nombres de variables deben ser descriptivos y reflejar su propósito o contenido. Por ejemplo, en lugar de usar `x` o `temp`, es preferible utilizar nombres como `edad` o `lista_nombres`.

Utilizar minúsculas y guiones bajos para separar palabras: Para variables con nombres compuestos, utilizar minúsculas y guiones bajos para separar palabras. Por ejemplo, `nombre_usuario`, `edad_persona`.

Evitar caracteres especiales: Se recomienda evitar el uso de caracteres especiales como `$` o `%` en los nombres de variables.

Empezar con una letra o un guion bajo: Los nombres de variables deben comenzar con una letra (a-z, A-Z) o un guion bajo (_). No deben comenzar con un número.

Evitar palabras reservadas: No debes utilizar palabras reservadas de Python como nombres de variables. Por ejemplo, no uses if, for, while, etc.

¿Qué es un Heredoc en Python?

Es una forma de declarar cadenas multilinea de texto y se hace con el uso del símbolo `""" """`. Ejemplo `""" Este es un ejemplo de Heredoc en Python. Puedes incluir múltiples líneas de texto aquí sin necesidad de escapar caracteres especiales o utilizar concatenación de cadenas. """`

¿Qué es una interpolación de cadenas?

Es una forma de insertar valores de variables dentro de una cadena de texto, evitando así hacerlo manualmente, y esto se consigue en python con la función `str.format()`, con la letra `f`, o con el símbolo `%`. Ejemplo

```
nombre = "Juan"
```

```
edad = 30
```

```
cadena = "Hola, mi nombre es %s y tengo %d años." % (nombre, edad)
```

Otro ejemplo:

```
nombre = "Juan"
```

```
edad = 30
```

```
cadena = "Hola, mi nombre es {} y tengo {} años.".format(nombre, edad)
```

Otro ejemplo:

```
nombre = "Juan"
```

```
edad = 30
```

```
cadena = f"Hola, mi nombre es {nombre} y tengo {edad} años."
```

¿Cuándo deberíamos usar comentarios en Python?

En python usariamos comentarios para explicar codigo, marcar mejoras pendientes, hacer entender el codigo que esta escrito porque y como se ha hecho asi entre otras cosas, pero no es recomendable hacer muchos comentarios.

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

Las aplicaciones monoliticas todos los componentes estan en la misma aplicacion mientras las de microservicios estan separadas y si comunican a traves de API's, las monoliticas son mas rapidas que las de microservicios y la desventaja es que haciendo modificaciones en ellas puede afectar a todas las otras partes que componen la aplicacion mientras que en las de microservicios no ocurre esto si una parte de la aplicacion falla no afecta a los demas servicios de la aplicacion.