

- 2.1.1.6 LABORATORIO: La función print()

•

LABORATORIO

Tiempo Estimado

5-10 minutos

Nivel de Dificultad

Muy fácil

Objetivos

- Familiarizarse con la función `print()` y sus capacidades de formato.
- Experimentar con el código de Python.

Escenario

El comando `print()`, el cual es una de las directivas más sencillas de Python, simplemente imprime una línea de texto en la pantalla.

En tu primer laboratorio:

- Utiliza la función `print()` para imprimir la línea `"¡Hola, Mundo!"` en la pantalla.
- Una vez hecho esto, utiliza la función `print()` nuevamente, pero esta vez imprime tu nombre.
- Elimina las comillas dobles y ejecuta el código. Observa la reacción de Python. ¿Qué tipo de error se produce?
- Luego, elimina los paréntesis, vuelve a poner las comillas dobles y vuelve a ejecutar el código. ¿Qué tipo de error se produce esta vez?
- Experimenta tanto como puedas. Cambia las comillas dobles a comillas simples, utiliza múltiples funciones `print()` en la misma línea y luego en líneas diferentes. Observa que es lo que ocurre.

- 2.1.1.18 LABORATORIO: La función print()

•

LABORATORIO

Tiempo Estimado

5-10 minutos

Nivel de Dificultad

Muy fácil

Objetivos

- Familiarizarse con la función de `print()` y sus capacidades de formato.
- Experimentar con el código de Python.

Escenario

Modifica la primera línea de código en el editor, utilizando las palabras clave `sep` y `end`, para que coincida con el resultado esperado. Recuerda, utilizar dos funciones `print()`.

No cambies nada en la segunda invocación de `print()`.

Salida Esperada

```
Fundamentos***Programación***en...Python
```

Tiempo Estimado

5-15 minutos

Nivel de Dificultad

Fácil

Objetivos

- Experimentar con el código Python existente.
- Descubrir y solucionar errores básicos de sintaxis.
- Familiarizarse con la función `print()` y sus capacidades de formato.

Escenario

Recomendamos que **juegues con el código** que hemos escrito para ti y que realices algunas correcciones (quizás incluso destructivas). Siéntete libre de modificar cualquier parte del código, pero hay una condición: aprende de tus errores y saca tus propias conclusiones.

Intenta:

- Minimizar el número de invocaciones de la función `print()` insertando la secuencia `\n` en las cadenas.
- Hacer la flecha dos veces más grande (pero mantener las proporciones).
- Duplicar la flecha, colocando ambas flechas lado a lado; nota: una cadena se puede multiplicar usando el siguiente truco: `"cadena" * 2` producirá `"cadenacadena"` (te contaremos más sobre ello pronto).
- Elimina cualquiera de las comillas y observa detenidamente la respuesta de Python; presta atención a donde Python ve un error: ¿es el lugar en donde realmente existe el error?
- Haz lo mismo con algunos de los paréntesis.
- Cambia cualquiera de las palabras `print` en otra cosa (por ejemplo de minúscula a mayúscula, `Print`) - ¿Qué sucede ahora?
- Reemplaza algunas de las comillas por apóstrofes; observa lo que pasa detenidamente.