

Actividad No 1.

INTRODUCCIÓN A PHP



9 de octubre de 2024

CEAI - SENA

Samuel Ortiz Rivera

Tabla de contenido

[1. ¿Cómo convertir el código PHP en bytecode? 2](#_Toc179374905)

[2. Ejemplos de PHP a bytecode 2](#_Toc179374906)

[3. ¿Cómo funciona el compilador JIT? 4](#_Toc179374907)

[4. ¿Qué es un servidor web? Arquitectura del servidor Web. 4](#_Toc179374908)

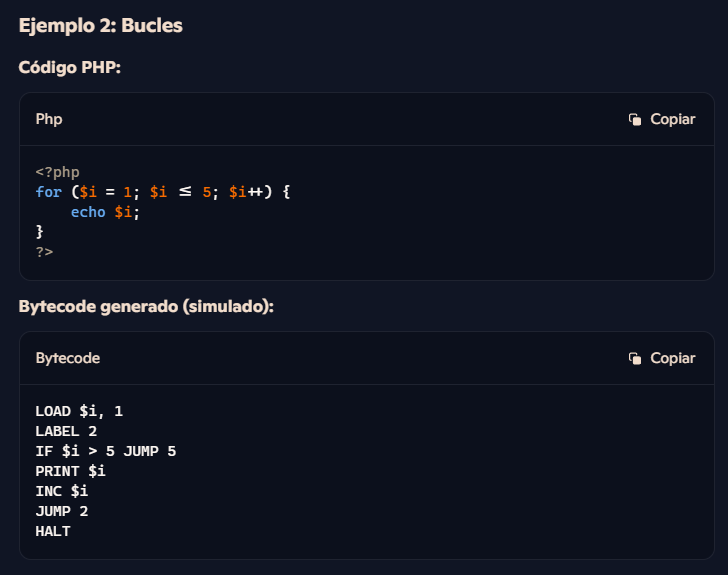
# 1. ¿Cómo convertir el código PHP en bytecode?

El bytecode es una forma intermedia del código que no es directamente ejecutable por la CPU, pero que puede ser interpretado por una máquina virtual. Para convertir el código PHP en bytecode:

* **Escribe tu código PHP** como lo harías normalmente.
* **Instala una herramienta** como **PHP-APC (Alternative PHP Cache) o PHP-OPcache.** Estas herramientas almacenan el bytecode en la memoria para mejorar la velocidad de ejecución.
* **Habilita la caché** en tu configuración de PHP (php.ini) con una línea como opcache.enable=1.

# 2. Ejemplos de PHP a bytecode







En la práctica, el bytecode generado no es tan fácilmente legible y se gestiona internamente por la máquina virtual de PHP.

# 3. ¿Cómo funciona el compilador JIT?

El **compilador JIT (Just-In-Time)** es una técnica de optimización que convierte el bytecode en código máquina nativo en tiempo real, es decir, mientras la aplicación se está ejecutando. Funciona así:

* Traducción JIT: Cuando se ejecuta un script PHP, el compilador JIT traduce partes del bytecode a código máquina.
* Ejecución rápida: El código máquina es ejecutado directamente por la CPU, lo cual es más rápido que interpretar bytecode.

Esto mejora significativamente el rendimiento de aplicaciones PHP.

# 4. ¿Qué es un servidor web? Arquitectura del servidor Web.

Un servidor web es un software que recibe solicitudes de los navegadores (clientes) y devuelve las páginas web solicitadas.

**Arquitectura de un servidor web:**

* **Cliente (Navegador web):** Envía solicitudes HTTP al servidor web.
* **Servidor web:** Procesa las solicitudes y devuelve las respuestas.
* **Aplicación Web:** Código que genera la respuesta (puede ser PHP, Python, etc.).
* **Base de Datos:** Almacena la información necesaria para generar la respuesta.

Por ejemplo, cuando visitas una página web, tu navegador (cliente) envía una solicitud al servidor web, que luego ejecuta el código (como PHP) y consulta la base de datos, finalmente devolvien do una página HTML que ves en tu navegador.