

EJERCICIO 18-03-2025

YENNIFER GISELLE TORRES RINCÓN

Cod 561220175

ING. WILLIAM ALEXANDER MATA LLANA PORRAS

DOCENTE LINEA DE PROFUNDIZACION III

UNIVERSIDAD DE CUNDINAMARCA EXTENSIÓN CHÍA

PROGRAMA DE INGENIERÍA DE SISTEMAS

FACULTAD DE INGENIERÍA

2024

EJERCICIO CLASE LINEA III 18032025

1. ¿Qué es un archivo YAML y para qué se utiliza en desarrollo de software?

¿Qué es un archivo YAML?

YAML (YAML Ain't Markup Language) es un formato de serialización de datos legible por humanos. Esto significa que está diseñado para que tanto las personas como las máquinas puedan leerlo y escribirlo fácilmente. Se utiliza comúnmente para archivos de configuración y para intercambiar datos entre diferentes lenguajes de programación.

Características clave de YAML:

- **Legibilidad:** La sintaxis de YAML es simple y utiliza sangrías para estructurar los datos, lo que facilita su lectura y comprensión.
- **Independencia del lenguaje:** YAML puede ser utilizado con una amplia variedad de lenguajes de programación, lo que lo hace muy versátil.
- **Estructura jerárquica:** YAML permite representar datos estructurados de forma jerárquica, utilizando listas y diccionarios.

Usos en el desarrollo de software:

- **Archivos de configuración:** YAML se utiliza ampliamente para configurar aplicaciones y servicios. Por ejemplo, en proyectos de Python, es común usar archivos YAML para definir las dependencias y los parámetros de configuración.
- **Despliegue de aplicaciones:** Herramientas como Kubernetes utilizan archivos YAML para definir la configuración de los despliegues de aplicaciones en contenedores.
- **Automatización:** YAML se emplea en herramientas de automatización como Ansible para definir tareas y flujos de trabajo.
- **Intercambio de datos:** YAML puede ser utilizado para intercambiar datos entre diferentes aplicaciones y sistemas.

2. Diferencia entre YAML y JSON

YAML (YAML Ain't Markup Language) y JSON (JavaScript Object Notation) son formatos de serialización de datos legibles por humanos que se utilizan comúnmente para la configuración, el intercambio de datos y el almacenamiento de datos. Aunque

comparten algunas similitudes, también tienen diferencias clave, especialmente en lo que respecta a las referencias.

Diferencias clave:

Legibilidad:

- YAML está diseñado para ser más legible por humanos, utilizando la sangría para estructurar los datos y permitiendo comentarios.
- JSON es más conciso y fácil de analizar por las máquinas, pero menos legible para las personas.

Sintaxis:

- YAML utiliza una sintaxis más flexible, permitiendo varios estilos y estructuras de datos.
- JSON tiene una sintaxis estricta basada en pares clave-valor y matrices.

Comentarios:

- YAML admite comentarios, lo que puede mejorar la legibilidad y la documentación.
- JSON no admite comentarios de forma nativa.

Referencias:

- YAML admite referencias y anclas, lo que permite reutilizar partes de un documento y evitar la redundancia. Esto puede ser útil para archivos de configuración complejos.
- YAML tiene una forma de utilizar anclas (&) y alias (*) para referirse a la misma información en el documento.
- JSON no tiene un mecanismo de referencia nativo.
- Para implementar referencias en JSON, se deben utilizar soluciones personalizadas o bibliotecas externas.

Tipos de datos:

- YAML admite una gama más amplia de tipos de datos, incluidos los tipos de datos definidos por el usuario.
- JSON admite un conjunto más limitado de tipos de datos básicos (cadenas, números, booleanos, nulos, matrices y objetos).

Casos de uso:

- YAML se utiliza a menudo para archivos de configuración, archivos de orquestación (como Kubernetes) y documentación.
- JSON se utiliza ampliamente para el intercambio de datos en aplicaciones web, API y almacenamiento de datos.

3. ¿Cuándo se recomienda usar uno sobre el otro?

La elección entre YAML y JSON depende en gran medida del contexto y las prioridades del proyecto.

Cuando usar YAML:

- **Archivos de configuración:**
YAML es excelente para archivos de configuración debido a su legibilidad. La sangría y la sintaxis clara facilitan la comprensión y modificación de las configuraciones.
Su capacidad para comentarios es muy útil para documentar la configuración.
Su capacidad de referencias es muy útil para configuraciones complejas que necesitan reutilizar porciones de información.
- **Orquestación de contenedores (Kubernetes, Docker Compose):**
YAML es el estándar para definir configuraciones en herramientas de orquestación de contenedores como Kubernetes y Docker Compose.
- **Automatización y DevOps:** Herramientas de automatización como Ansible utilizan YAML para definir playbooks y tareas.
- **Documentación:** La legibilidad de YAML lo hace adecuado para la creación de documentación legible por humanos.

Cuándo usar JSON:

- **APIs web:** JSON es el formato estándar para el intercambio de datos en APIs web debido a su simplicidad y eficiencia en el análisis por máquinas.
- **Intercambio de datos entre aplicaciones:** JSON es ampliamente compatible con lenguajes de programación y plataformas, lo que facilita el intercambio de datos entre diferentes aplicaciones.
- **Almacenamiento de datos:** Bases de datos NoSQL como MongoDB utilizan JSON para almacenar y recuperar datos.
- **Configuraciones simples:** Para configuraciones simples donde la legibilidad humana no es la principal preocupación, JSON puede ser una opción más concisa.

Cuando el rendimiento es critico:

JSON al ser mas simple, requiere menos procesamiento que YAML.

4. Como se crean los archivos yml

Los archivos YAML son archivos de texto plano utilizados para la serialización de datos, comúnmente empleados en configuraciones, orquestación de contenedores y automatización. Su legibilidad y estructura jerárquica basada en sangría los hacen ideales para representar datos complejos de manera clara.

Proceso de creación:

- **Editor de texto:**
Utilice cualquier editor de texto plano, como Visual Studio Code, Sublime Text o Notepad++.
- **Estructura y sintaxis:**

Sangría: La sangría es crucial en YAML. Utilice espacios (se recomienda evitar tabulaciones) para definir la jerarquía de los datos. La consistencia en la sangría es fundamental.

Pares clave-valor: Los datos se organizan en pares "clave: valor".

Listas: Las listas se representan con un guión (-) al inicio de cada elemento.

Comentarios: Agregue comentarios usando el símbolo de almohadilla (#).

5. Referencias

- IBM. (s.f.). ¿Qué es YAML? Recuperado de: <https://www.ibm.com/es-es/topics/yaml>
- KeepCoding. (2025). Fichero YAML: ¿Qué es y para qué sirve? Recuperado de: <https://keepcoding.io/blog/que-es-y-para-que-sirve-el-fichero-yaml/>
- Pure Storage. (s.f.). ¿Qué es YAML? ¿Se pueden escribir cadenas multilínea? Recuperado de: <https://www.purestorage.com/es/knowledge/what-is-yaml-multiline-string.html>
- Wikipedia. (s.f.). YAML - Wikipedia, la enciclopedia libre. Recuperado de: <https://es.wikipedia.org/wiki/YAML>
- Online converter. (s.f.). Todo sobre el archivo YML - Online converter. Recuperado de: <https://www.online-convert.com/es/formato-de-archivo/yml>
- AWS. (s.f.). *YAML frente a JSON: diferencia entre los formatos de serialización de datos*. Amazon Web Services, Inc. Recuperado de <https://aws.amazon.com/es/compare/the-difference-between-yaml-and-json/>
- Latenode. (s.f.). *JSON vs YAML: Comparación de formatos de datos para el desarrollo moderno*. Latenode. Recuperado de <https://latenode.com/es/blog/json-vs-yaml>
- Red Hat. (s.f.). *YAML: qué es, usos, sintaxis y ejemplos*. Red Hat. Recuperado de <https://www.redhat.com/es/topics/automation/what-is-yaml>