

Desarrollo de Software

Estilo de código

Selecciona tres elementos de la guía de estilo con los que estés de acuerdo y, para cada uno, explica por qué. Si no hay suficientes elementos con los que estés de acuerdo, da tu mejor estimación sobre la razón detrás de la selección (con tus propias palabras) y los beneficios que proporciona. Nota: "No hay razón" y "No hay beneficios" no son respuestas aceptables aquí.

Utiliza Java Style Guide (Google), Python Style Guide (Google), PEP 8 (Python Enhancement Proposal).

Muestra código donde aplicas lo que estas de acuerdo de la guia que elegistes.

Selecciona ahora tres elementos de la guía de estilo con los que no esté de acuerdo y, para cada uno, explique por qué. Si no hay suficientes elementos con los que no está de acuerdo, proporciona tu mejor estimación de por qué alguien podría no estar de acuerdo y qué posibles desventajas tiene usarlo.

Muestra código donde aplicas lo que no estas de acuerdo de la guia que elegistes.

Análisis de código estático

Selecciona una herramienta de análisis estático bien establecida que tenga documentación en línea de sus capacidades. Selecciona una función de la herramienta. Explica brevemente el propósito de la función, cómo usarla y por qué es importante tener esta capacidad de análisis.

Utiliza la misma herramienta de análisis estático seleccionada para el mensaje anterior. Selecciona otra característica de la herramienta. Explica brevemente el propósito de esta característica, cómo usarla y por qué es importante tener esta capacidad de análisis.

Comentarios y documentación

Selecciona un proyecto de código abierto de tu elección que te interese. Encuentra el código fuente de este proyecto. Selecciona cualquier archivo (que se pueda encontrar y revisar mediante URL). Proporciona el nombre del proyecto, el propósito del archivo y el enlace.

Para el mismo archivo, ¿sigue un estándar de estilo de código? Si no es así, señala tres ejemplos de por qué no es así. Si tiene un estándar, ¿es un estándar formal? Si es así, nómbralo. Si el estándar no es formal, ¿qué elementos están presentes que le hacen pensar que existe un estándar?

Para el mismo archivo, ¿qué se podría mejorar? En caso contrario, explica tres elementos del estilo de código con el que estás de acuerdo.

¿El documento está bien documentado? Justifica tu respuesta.

"Autodocumentado" se refiere a la capacidad de un sistema o un componente de software para generar automáticamente documentación que describe su funcionamiento, API, parámetros, y otros aspectos relevantes. Un sistema autodocumentado puede extraer información directamente del código fuente para crear documentación actualizada y precisa.

Por ejemplo, en el contexto de la programación, un framework o una biblioteca autodocumentada puede generar automáticamente documentación en formato HTML, Markdown u otro formato legible por humanos a partir de comentarios en el código fuente, metadatos, o convenciones de nomenclatura específicas.

¿Los comentarios siguen un formato común, tanto en contenido como en ubicación?

¿El código es autodocumentado? Identifica dos lugares donde lo hace bien y explique por qué es una buena autodocumentación.

¿El código no se autodocumenta? Identifica dos lugares donde lo hace mal y explique por qué es una mala autodocumentación.