

Alumno: Esteban D. Caballero

1. ¿Qué propósito tiene el prólogo en el libro "Código limpio"?
2. ¿Quién es el autor del prólogo y qué relación tiene con Robert Cecil Martin?
3. ¿Cuál es la importancia de escribir código limpio según el prólogo?

Introducción:

4. ¿Cuál es el objetivo principal del libro "Código limpio"?
5. Mencionar al menos tres beneficios de tener código limpio en un proyecto de desarrollo de software.
6. ¿Por qué es importante el aspecto económico al considerar el mantenimiento de código limpio?

Sobre la imagen de cubierta:

7. ¿Qué representa la imagen de cubierta del libro?
8. ¿Cuál es el mensaje que se intenta transmitir a través de la imagen de cubierta?

Capítulo 1: Código limpio:

9. ¿Cuál es la metáfora utilizada para describir el código limpio?
10. Mencionar al menos tres características de un código limpio según el autor.
11. ¿Cuál es la importancia de que el código sea fácil de leer?

Capítulo 2: Nombres con sentido:

12. ¿Por qué los nombres bien elegidos son importantes en la escritura de código?
13. Mencionar tres pautas sugeridas por el autor para nombrar variables, funciones y clases.
14. ¿Cuál es el efecto de utilizar nombres genéricos o ambiguos en el código?

Resolución:

Prólogo:

1. El propósito del prólogo en el libro "Código limpio" es dar una introducción en la que se vislumbre la importancia del código limpio en el desarrollo de software.
2. El autor del prólogo es James O. Coplien, quien es un reconocido experto en desarrollo de software y ha trabajado en colaboración con Robert Cecil Martin en diferentes proyectos.
3. Según el prólogo, escribir código limpio es importante porque el mismo es leído por humanos y debe poder transmitir su funcionamiento sin dedicar mucho tiempo a su revisión. El código limpio es

una gran ayuda a la hora de trabajar con grandes equipos de desarrollo dado que de no existir se complicaría el desarrollo de los mismos puesto que con el pasar del tiempo los miembros del grupo que inicio con el desarrollo empezarían a desaparecer y se volvería una tarea tediosa y sumamente molesta el ponerse a día con el código sin mencionar las posibles consecuencias de tomar tal decisión como pueden ser los largos periodos de revisión e incluso retrasos tan grandes que comprometen la estabilidad futura del producto.

Introducción:

4. El objetivo principal del libro "Código limpio" es concientizar a los programadores sobre la importancia de las buenas prácticas dada las ventajas del código de calidad.

5. Algunos beneficios de tener código limpio en un proyecto de desarrollo de software son:

- Disminuir la cantidad de tiempo improductivo dedicado al análisis gastado en el esfuerzo de intentar comprender el código escrito por otra persona, una mayor facilidad para realizar modificaciones, una reducción de los posibles errores por causa de mala interpretaciones.

6. El aspecto económico es importante en el mantenimiento de código limpio debido a que el costo de mantener y actualizar código mal escrito puede ser significativamente mayor a largo plazo. Lo cual a veces se encuentra compensado por sus ventajas, la más destacable es la de que es una apuesta a futuro por la escalabilidad del soft.

Sobre la imagen de cubierta:

7. La imagen de cubierta del libro es la Galaxia del Sombrero (M104). M104 se encuentra en Virgo, a unos 30 millones de años luz, y su núcleo es un súper agujero negro que pesa aproximadamente mil millones de masas solares.

8. A través de la imagen de cubierta, se intenta transmitir el mensaje de que el agujero negro que habita en su centro se tragó muchas de las cosas que se encontraban en sus proximidades; el autor menciona que la mayoría de los agujeros negros se conforman con engullir algunas estrellas lo que me da a entender que el autor hace referencia a la posibilidad de que un código que sea erróneo y demos por valido, si bien podría pasar desapercibido sin causar muchas más molestias que unos cuantos parches, el mismo podría desencadenar un desastre inmenso que se volviese incontrolable en un futuro a menos que se dispusieran de una cantidad absurda e innecesaria de recursos para salvar la situación y que, incluso de darse la situación, demandaria mucho tiempo;

Capítulo 1: Código limpio:

9. El autor utiliza la metáfora de "leer el código como si fuera un libro" para describir el código limpio. La idea es que el código debe ser fácil de leer y entender, al igual que un libro bien escrito.

10. Algunas características de un código limpio mencionadas por el autor incluyen: simplicidad, claridad, ausencia de duplicación, tener pruebas unitarias, modularidad y consistencia.

11. La importancia de que el código sea fácil de leer radica en que los programadores pasan la mayoría de su tiempo leyendo y comprendiendo el código existente. Un código legible facilita la colaboración y el mantenimiento a largo plazo.

Capítulo 2: Nombres con sentido:

12. Los nombres bien elegidos son importantes en la escritura de código porque comunican el propósito y la funcionalidad de las variables, funciones y clases. Los nombres claros y descriptivos mejoran la comprensión del código y su mantenibilidad.

13. Algunas pautas sugeridas por el autor para nombrar variables, funciones y clases son: utilizar nombres que revelen su propósito, evitar nombres genéricos o ambiguos, utilizar nombres pronunciables y buscar consistencia en la elección de nombres.

14. El uso de nombres genéricos o ambiguos en el código puede causar confusión y dificultar la comprensión del mismo. Esto puede llevar a errores y problemas de mantenimiento, ya que los desarrolladores pueden interpretar de manera incorrecta el propósito de una variable o función.