**Nombre: Carlos Alberto De La Rosa Banquez**

1. **Mejora en el diseño de tu código:**

* **Organización de clases**:
* Comienza con variables (constantes estáticas públicas primero)
* Sigue con métodos públicos
* Coloca utilidades privadas después de los métodos públicos que las usan
* **Encapsulación**:
* Haz las variables y funciones privadas cuando sea posible
* Usa protected o package-private solo si es necesario para pruebas
* **Tamaño de las clases**:
* Mantén las clases pequeñas
* Enfócate en una sola responsabilidad por clase
* **Evita "clases Dios"**:

1. Me enseña a no crear clases con demasiados métodos (como SuperDashboard con 70 métodos)
2. Comprensión de buenas prácticas: El texto introduce a estándares de la industria en programación orientada a objetos. Esto me preparará mucho mejor para mi futura carrera profesional.
3. Habilidades de refactorización: Aprender a identificar y mejorar código mal estructurado, una habilidad valiosa en el desarrollo de software.
4. Mejor documentación: Siguiendo estas pautas, aprenderás a organizar y documentar tu código de manera más efectiva, lo que facilitará la colaboración en proyectos grupales.
5. Preparación para revisiones de código: Entender estos conceptos te ayudará a dar y recibir retroalimentación más útil durante las revisiones de código en clase.
6. Resolución de problemas: Aplicar estos principios te ayudará a abordar problemas complejos dividiéndolos en partes más manejables (clases más pequeñas con responsabilidades claras).