

## Objetivos

- Aplicar y entender el uso del patrón factory
- Comprender la manipulación de los objetos y la data expuesta
- Aplicar técnicas para mejorar la estabilidad del código en el tiempo

## Material de trabajo

- ¿Qué ventajas tiene el enum al programar?
- ¿Cuándo usar un ENUM?
- ¿Puedo usar un ENUM en la DB?
- ¿Puede el ENUM usarse en el JSON?
- ¿Es posible tener un ENUM que se convierta en elementos diferentes cuando se trabaje con JSON, DB o el código?
- ¿Por qué debo usar el patrón factory?
- ¿Cuál es la implementación adecuada del patrón factory?
- ¿Por qué se usan clases separadas para los request y response?
- ¿Qué otras ventajas tiene el uso de clases independientes para los request y response?

## Tareas

- ¿Qué sucede cuando los LOGs en una aplicación tienen mal diseño?
- ¿Qué data debo hacer log en la aplicación?
- ¿Por qué no debo usar los logs para debuggear las aplicaciones?

## Otros materiales

- <https://www.baeldung.com/java-abstract-factory-pattern>
- <https://www.geeksforgeeks.org/factory-method-design-pattern-in-java/>
- <https://www.baeldung.com/jackson-annotations>
- <https://www.baeldung.com/jackson-serialize-enums>
-