

Práctica 3

Luis Geovanni Méndez Ávila — 317143980

27 de Noviembre 2020

I Teoría:

1.- Características principales:

Patrón factory:

Delega la responsabilidad de crear objetos a una clase (fábrica) concreta y oculta detalles de implementación del objeto concreto. Puede crear varios tipos de objetos que implementan una interfaz común.

Patrón abstract factory:

Sigue un principio similar al de Factory pero en este caso podemos hacer más Fabricas para distintos objetos y no nada más para una clase de objeto en específico.

Patrón builder:

Crea objetos complejos a partir de objetos más simples que lo componen.

2.- Desventajas:

Patrón factory:

Unicamente se limita a crear objetos de una interfaz en común , en caso de querer crear objetos de otro tipo no se puede usar.

Patrón abstract factory:

No se me ocurre alguno.

Patrón builder:

No se me ocurre alguno.

II. Implementación:

Elegí el patrón Abstract factory debido a que para construir un auto debo usar distintos tipos de componentes (Llantas, Carroceria, etc), de los cuales derivan diferentes tipos de objetos relacionandos con esos componentes, crear esos objetos podría ser delegado a una fabrica por componente, entonces necesitaba crear más fábricas para cada componente en particular.

Notas: Se me olvidó poner lo del saldo del usuario y el javadoc.