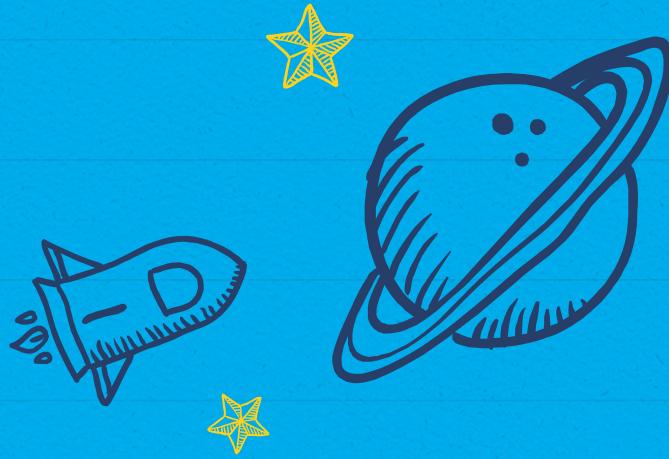


PROGRAMACIÓN II



PATRONES DE DISEÑO CREACIONALES





FACTORY (FACTORY METHOD)

FACTORY

- ✗ Lo usamos para delegar a una clase la responsabilidad de la creación (fabricación) de los objetos.
- ✗ De esta manera, se respeta el **Principio de Responsabilidad Única**



FACTORY

- ✗ Cuando no se sabe en tiempo de diseño qué clase en específico se necesitará para crear una instancia de ella.
- ✗ Con el patrón, se logra un alto nivel de **encapsulamiento**



FACTORY (IMPLEMENTACION)

- ✗ Crear una interfaz, que defina el comportamiento común de las demás clases
- ✗ Crear las clases que implementen esta interfaz



FACTORY (IMPLEMENTACION)

- ✗ Crear una fábrica. (La clase cuya función es crear objetos de las clases existentes)
- ✗ En la clase principal, usar la fábrica para crear el objeto deseado, pasandole el identificador



TOMAR EN CUENTA:

- ✗ En la fábrica, al escribir el código de creación de cada objeto, debes establecer un mecanismo para distinguir que objeto crear
- ✗ En la fábrica, el método encargado de construir los objetos, debe devolver como tipo de dato, el tipo de la interface



GRACIAS!

