

## **Ejercicio: Explicación sobre Inyección de dependencias y polimorfismo.**

Se muestra un pequeño programa sobre una ensambladora de teléfonos.

Donde nuestra Clase “Celular” en realidad necesita de varios “Componentes” para ser creado, en este caso solo se ilustra con las pantallas, pero también el programa puede considerar, baterías, cámaras, procesadores, memorias, conectores, antenas y gran etc.

### **Inyección de Dependencias.**

Supongamos que hoy queremos crear el primer teléfono 100% fabricado en México, ups, pero nosotros no tenemos experiencia creando teléfonos y construir nuestras propias pantallas, baterías, cámaras, sería muy complicado no? Por el momento necesitamos quien nos vendan dichos componentes para construir nuestro teléfono.

En términos técnicos.

Para evitar que en mi clase celular cree sus propias pantallas, estas pantallas son inyectadas mediante una interfaz “Pantalla”, así mi método Main se encarga de inyectar la pantalla al crear el Celular. Como ventaja, mi Celular no depende si uso una MarcaX o una MarcaY de pantalla, mediante la interfaz sabe debe tener una pantalla, incluso puedo crear un Celular con una pantalla marcaZ y funcionaria por igual.

### **Polimorfismo**

El polimorfismo va ligado con la inyección de dependencias, mi Interfaz de Pantalla, solo me dice que el Celular debe tener una pantalla y que dicha Pantalla debe tener un método “Encender Pantalla” pero en realidad no tengo que escribir un método “Encender Pantalla” para todas las marcas disponibles.

Cuando llamo al método encenderPantalla(), su comportamiento cambiará de acuerdo con el objeto que se le ha inyectado en miPantalla, es decir si se le inyecta:

PantallaSamung => Enciende la pantalla con mensaje de Samsung

PantallaLG => Enciende la pantalla con mensaje de LG

PantallaSharp => Enciende la pantalla con mensaje de Sharp