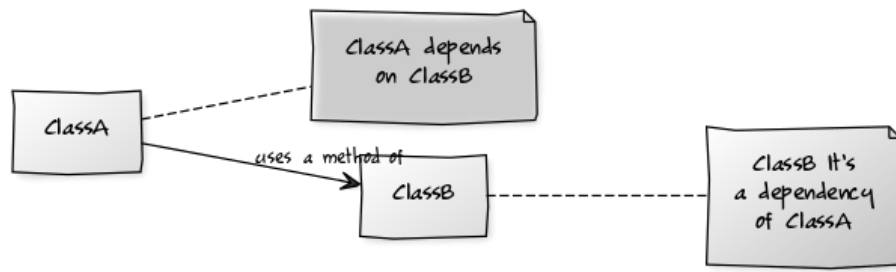


## Inyección de dependencias en Java 8

La inyección de dependencias es un patrón de diseño utilizado en Java para proveer a diferentes clases de las dependencias que estas necesitan para funcionar, y estas dependencias se proveen de manera externa en lugar de que la clase misma indique las dependencias.

Una dependencia es una clase usada en otra clase para que esta funcione correctamente, es decir si existe una clase Auto que necesita de la clase Motor para funcionar se dice que Motor es una dependencia de la clase Auto.



El patrón de diseño implementado en la inyección de dependencias consiste en definir las dependencias de manera externa en lugar de inicializar las dependencias dentro de la clase receptora.

Existen diferentes formas de inyectar las dependencias la más común es la inyección por constructor que consiste en inyectar la clase mediante el constructor.

```
class ClassA {
    ClassB classB;

    /* Constructor Injection */
    ClassA(ClassB injected) {
        classB = injected;
    }

    int tenPercent() {
        return classB.calculate() * 0.1d;
    }
}
```

También se puede inyectar usando un método set, en el que se asigna la dependencia usando un setter de la clase que necesita la dependencia.

```
class Main {
    public static void main(String... args) {
        ClassA classA = new ClassA();
        ClassB classB = new ClassB();

        classA.setClassB(classB);

        System.out.println("Ten Percent: " + classA.tenPercent());
    }
}
```

## Referencias

Chacin, C. (2019, 14 de noviembre). *Inyección de dependencias en Java*. DEV Community.  
<https://dev.to/cchacin/inyeccion-de-dependencias-en-java-44cg>

Baeldung. (2024, 4 de Abril). *Intro to Inversion of Control and Dependency Injection with Spring*  
<https://www.baeldung.com/inversion-control-and-dependency-injection-in-spring>

Bloch, J. (2018). *Effective Java* (3rd ed.). Addison-Wesley.