

DEPENDENCY INJECTION

---

SPRING CORE

# SPRING FRAMEWORK

- ▶ Spring Framework is an application framework and Inversion of Control Container for Java Platform.
- ▶ Can be used by any Java Application.
- ▶ Alternative to JEE's EJB.
- ▶ Spring Framework core technology include : Dependency Injection, DAO and ORM, Spring MVC, Scheduler, Testing and Mock Object, SpEL, AOP, and many more...





# IOC – INVERSION OF CONTROL

Principle where the control flow of a program is inverted: instead of the programmer controlling the flow of a program, the external sources (framework, services, other components) take control of it.

# DI – DEPENDENCY INJECTION

Spring Framework using DI as implementation of IoC. Required Object or Dependency are passed into an object through constructors/setters/service lookups.

# DI BENEFIT

- ▶ Flexible to configure..
- ▶ Decrease coupling between class, so class more independent, easy to reuse, easy to test, easy to maintain.
- ▶ Reduce boilerplate.
- ▶ Allowing to reconfigure without recompilation.

# SPRING BEAN

In Spring, the objects that form the backbone of your application and that are managed by the Spring IoC container are called beans.

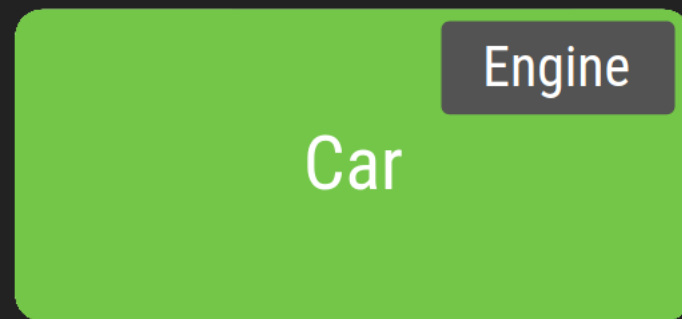
A bean is an object that is instantiated, assembled, and otherwise managed by a Spring IoC container/Spring Context by reading configuration metadata.

A Bean configuration metadata is provided using :

- ▶ XML
- ▶ Annotation
- ▶ Java Configuration

# DI IN ACTION

Create a Car which has Engine inside of it



# Stereotype - @Component

- ▶ It is Generic Bean with multipurpose



# BEAN SCOPE

- ▶ **Singleton (default)**  
This scopes the bean definition to a single instance per Spring IoC container (default).
- ▶ **Prototype**  
This scopes a single bean definition to have any number of object instances.
- ▶ **Request**  
This scopes a bean definition to an HTTP request. Only valid in the context of a web-aware Spring ApplicationContext.
- ▶ **Session**  
This scopes a bean definition to an HTTP session. Only valid in the context of a web-aware Spring ApplicationContext
- ▶ **global-session**  
This scopes a bean definition to a global HTTP session. Only valid in the context of a web-aware Spring ApplicationContext.

# STEREOTYPE

