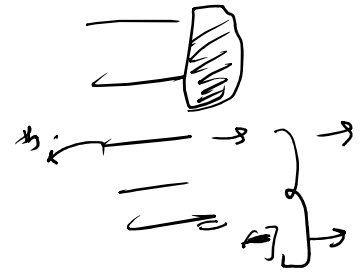


↳ super power of Spring

#

Person p = new Person();

→ Programmer was managing the life cycle of the objects



→ Spring is creating and managing the life cycle of the object.

↳ [Inversion of Control]

⊛ Any class which has @Component annotation written over it, directly/indirectly is managed by Spring.

→ Spring creates an object for all those classes which have @Component written on it.

Scope: How many copies of the object have been created.

Bean

A Java object

→ By default, scope of the bean created by Spring is 'singleton'

A Java object created and managed by Spring.

[what all beans and config Spring is managing]

⇒ IOC container : Its the context of a Spring application.  
Application Context

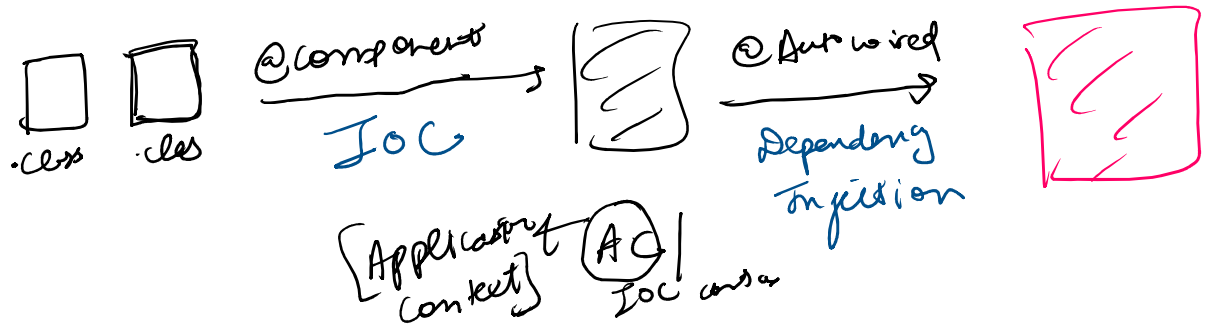
↓  
responsible for instantiating, creating and managing of Spring beans.



@Autowired

helps to use the beans/components created by Spring

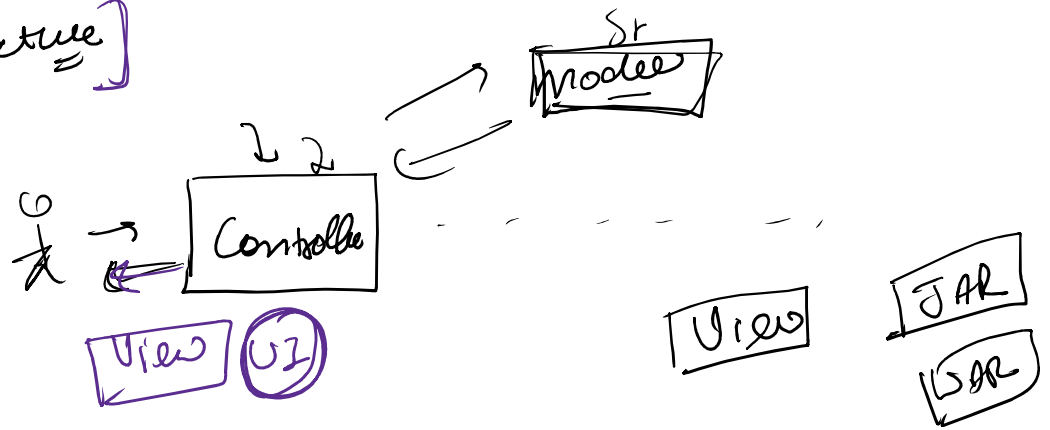
Dependency Injection : Using or injecting the beans created by Spring. @Autowired



⇒ IOC is based on Factory Design Pattern

→ What ever we can do with Lombok we can do without Lombok as well.

# MVC Architecture



# HTTP Status Codes

HTTP Response ⇒ Response + Status Code

400-499  
[4xx - Client side errors]  
500-599  
[5xx - Server side errors]

200-299 [2xx]  
200 → OK  
[201 → Created]

500 → 502  
[5xx - server side error]

# Response Entity

↳ [helps to send both response + status code]

↳ Response Entity is always returned from the controller layer