* Spring IOC container is the core of the Spring framework
* It creates objects, configures and assembles their dependencies, and manages the entire life cycle
* Container uses Dependency injection to manage the components that make up the application.
* Spring IOC gets the information about objects from a configuration file (XML) or Java Code or Java Annotation and Java POJO class. This object is called Beans.
* Controlling of Java Objects and the life cycle is managed by Inversion of Control

Two types of IOC Containers:

1. Bean Factory

2. ApplicationContext

WE need to use BeanFactory or ApplicationContext to get benefits from IOC container

BenaFactory is the basic version of IOC container

ApplicationContext extends the features of BeanFactory

Main Features of Spring IOC

* Creating Object for us
* Managing our Object
* Helping application to be configurable
* Managing dependencies

Beans ?

* Objects maintained by Spring IOC container (backbone of application)

Dependency Injection :

* It’s a software design technique in which the creation and binding of dependencies are done outside of the dependent class.
* Dependencises are provided already instantiated and ready to be used.
* DI can be achieved in 3 ways
* Interface Injection :
* Setter Injection
* Constructor Injection