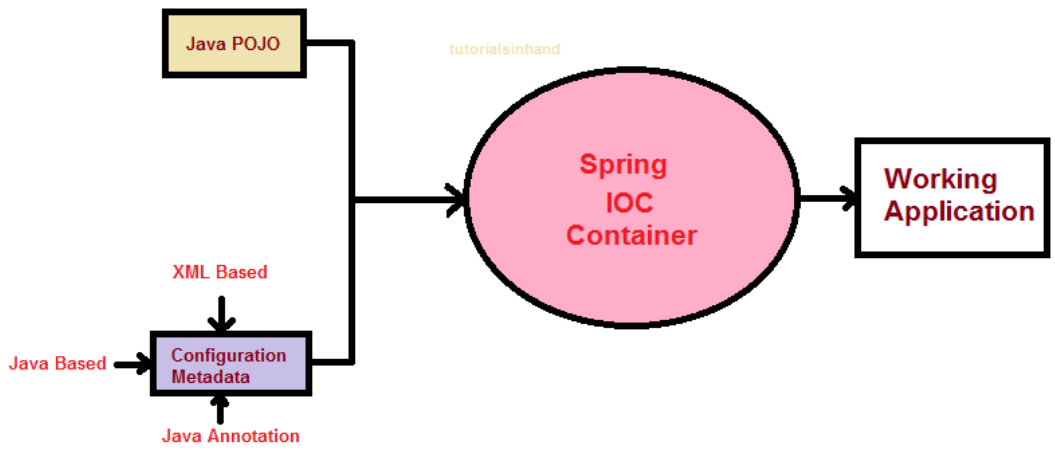
EXPLORING SPRING IOC CONTAINER, APPLICATION CONTEXT & BEAN FACTORY

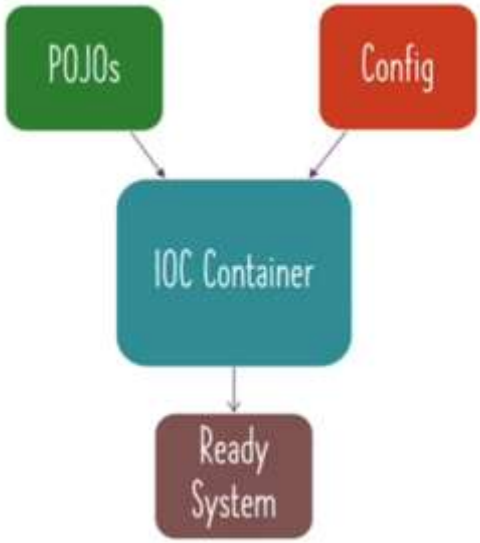
What is Spring Container?

- The Spring container is **responsible for creating and managing objects**, which are also known as **beans**, that **form the backbone of a Spring application**. The container is also responsible for **injecting dependencies** between beans, managing their configuration, and **handling the lifecycle** events of beans.
- The Spring container is typically implemented as an **inversion of control (IoC)** container, which means that it **manages the lifecycle** of objects **based on a set of configuration rules defined by the developer**, rather than the objects themselves controlling their own lifecycle. This allows for more flexible and modular applications, as well as easier testing and maintenance.
- There are **two types of Spring containers**:
    - The **BeanFactory** and the **ApplicationContext**.
        - The **BeanFactory** is the basic container that **provides the basic features** of the container.
        - The **ApplicationContext** builds on top of the BeanFactory and adds **more enterprise-level features** like **internationalization**, **event propagation, Easy integrations with spring AOP (Aspect Oriented Programming) and web-specific contexts**. This is most frequently used. It is recommended for web applications, web services, REST API and microservices.
- Overall, the Spring container is a critical component of any Spring-based application, as it provides the necessary infrastructure for **managing the lifecycle of objects** and promoting modularity and flexibility in the application architecture.
- **Terms** used to refer Spring Container:
    - Spring Context, Spring container, Spring IOC container.



Working of Spring Container

In our example, we have created **java classes** (record person, Address) and we have created **configuration file** "**HelloWorldConfiguration**" containing all the definition of the beans. These are the inputs for the Spring Container. The output of the spring container is a ready system. The system like, inside the **JVM** we have **spring context** which is managing all these beans that we have configured. This is how it would look like at run time.



Once we have created java classes and configuration, the IOC container creates the run time system for us.