**JAVA Spring Boot Note**

1. **Spring vs Spring Boot Framework**

**Spring** is a framework that allow Java EE 7 developer to build **simple, reliable and scalable** enterprise application.  
Main concept:  
- **Dependency Injection** (DI) (Inversion of Control) – In this principle, rather than the application taking control of the flow sequentially, it gives control to an external controller who drives the flow. The external controller is the **events**. When some event happens, the application flow continues. This gives flexibility to the application. In Spring, IoC is done by DI which are of three types – **setter injection, method injection,** and **constructor injection**.

- **Beans & Spring Context** – **objects** are called **beans**, and there is a **BeanFactory** that manages and configures (beans.xml) these beans.

Spring **Context**: For Dependency Injection (DI)

Spring **DAO**: For database operation using DAO pattern

Spring **JDBC**: For JDBC and Datasource support

Spring **ORM**: For ORM tool support such as Hibernate

Spring **AOP**: For Aspect Oriented Programming

Spring **Web** module: For creating web application

**Sprint Boot** is a framework that is built on top of the Spring Framework, widely used to develop **REST API**.

**RAD** (Rapid Application Developmen) -> to create a stand-alone Spring-based application with minimal effort (No XML configuration).



1. Restful API
2. JAVA Basic
3. Basic Syntax

Check version

java -version

|  |
| --- |
| public class Main {  public static void main(String[] args) {  System.out.println("Hello World");  }  } |

1. D
2. E
3. F
4. G
5. H