Spring Boot makes it easy to create stand-alone, production-grade Spring based Applications that you can "just run".

[Spring Initializr](https://start.spring.io/) to generate the base for our project.

@SpringBootApplication

public class Application {

    public static void main(String[] args) {

        SpringApplication.run(Application.class, args);

    }

Spring Boot Features

* Web Development

We can use the spring-boot- starter-web module to star

* SpringApplication

SpringApplication.run(className.**class**, args);

* Application events and listeners
* Admin features
* Externalized Configuration
* Properties Files
* YAML Support

It provides convenient way for specifying hierarchical configuration.

* Type-safe Configuration
* Logging

uses Common logging for all internal logging

* Security

**Beans and Dependency Injection**

The **@ComponentScan** annotation is used to find beans and the corresponding injected with **@Autowired** annotation.

**Spring Boot – Runners**

CommandLineRunner and ApplicationRunner, to run specific pieces of code when an application is fully started

public void run(String...args) throws Exception { // CommandLineRunner

public void run(ApplicationArguments args) throws Exception { // ApplicationRunner

**Spring Boot - Application Properties**

Command Line Properties:

Step 1 − After creating an executable JAR file, run it by using the command java –jar <JARFILE>.

Step 2 − Use the command given in the screenshot given below to change the port number for Spring Boot application by using command line properties.

Properties File:

application.properties – Properties files are used to keep ‘N’ number of properties in a single file to run the application in a different environment.

YAML File:

application.yml

Externalized Properties:

You can use the following command to specify the location of properties file while running the JAR −

-Dspring.config.location = C:\application.properties

Use of @Value Annotation:

read the environment or application property value in Java code.

If property not found-  Illegal Argument exception as **Could not resolve placeholder**

Spring Boot Active Profile:

From proptiers file



From yaml, single file is enough.

spring:

profiles: dev

application:

name: demoservice

server:

port: 9090