**Spring**

**Base Knowledge**

Maven

Maven Lifce Cycle

Multi module

Spring Container

Dependency Injection (Constructor and Setter based.)

Beans Life Cycle

Autowiring beans

Beans as Components

Custom Annotations

Qualifiers

Properties

**MVC**

Maven War Plugin

Maven Cargo and Tomcat

Server Dispatcher Servlet

Controllers

View

Model

Service

Request Params

**Spring Boot**

**Project Structure**

Dependencies 🡪 Web, DevTools

POM 🡪 Dependencies, Versions.

Packages 🡪 bootdemo: Main Class.

Tomcat 🡪 Server at port 8080.

If using Logback, paste it into Resources and rename it to Logback-Spring. Recommended so Spring can handle it.

Resources – application.properties 🡪 We can manage some properties easily.

(**ej:**

spring.output.ansi.enabled = always

spring.mvc.view.prefix = /WEB-INF/view/

spring.mvc.view.suffix = .jsp

).

**Controller**

Must be in same Package as BootDemoApplication(main file).

Maps to URLs and returns a JSON. We can call methods inside the Controller Class.

@RestController 🡪 To return a JSON

@RequestMapping(“/asd/asd”) 🡪 Maps methods to an URL, it will be auto invoked as soon as we hit the endpoint. **GET Method**.

@PostMapping 🡪 **POST**

@PutMapping 🡪 **PUT**

@DeleteMapping 🡪 **DELETE**

@PatchMappin­g 🡪 **PATCH**

Methods inside Controller:

@ResponseBody 🡪 To return inside map

@GetMapping(“api/miUrl”)