Spring Chitkara University Live [SCUL] 📚⚽

BRAVO !

Prerequisites of Spring Boot :

-Core Java and Collections (ArrayList , Hashmap)
-OOPS Concepts
-must have java on your machines (Java 8 version will work fine)
-we will use IntelliJ as an IDE
Install Apache Maven : https://maven.apache.org/install.html
Windows : https://www.youtube.com/watch?v=km3tLti4TCM
Mac /Unix : https://www.youtube.com/watch?v=j0OnSAP-KtU
Working with Maven : https://www.youtube.com/watch?v=pt3uB0sd5kY

IMP : Please make sure to install Postaman Client before next class .

WHAT IS SPRING AND SPRING BOOT (2.5.5) ?

The Spring Framework provides a comprehensive programming and configuration model for modern Java-based enterprise applications - on any kind of deployment platform.
It's packed with some nice features like Dependency Injection, and out of the box modules like:

1. Spring JDBC
2. Spring MVC
3. Spring Security
4. Spring AOP
5. Spring ORM
6. Spring Test

These modules can drastically reduce the development time of an application.

For example, in the early days of Java web development, we needed to write a lot of boilerplate code to insert a record into a data source. By using the JDBCTemplate of the Spring JDBC module, we can reduce it to a few lines of code with only a few configurations.

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

Spring Chitkara University Live [SCUL] 📚⚽

Spring Boot has Tomcat inbuilt . But Spring does not . **Are they different ?**
Yes ,Spring Boot is basically an extension of the Spring framework, which eliminates the boilerplate configurations required for setting up a Spring application.

Spring Boot is a faster wrapper on the Spring Framework . Read about it here
https://www.baeldung.com/spring-vs-spring-boot

Why do we need Maven ?

Building a software project typically consists of such tasks as downloading dependencies, compiling source code into binary code, running tests, packaging compiled code into deployable artifacts such as JAR, WAR, and ZIP files, and deploying these artifacts to an application server or repository.

Apache Maven automates these tasks, minimizing the risk of humans making errors while building the software manually and separating the work of compiling and packaging our code from that of code construction.Its a build automation tool for Spring Boot just like npm for Node js.

The configuration of a Maven project is done via a *Project Object Model (POM)*, represented by a *pom.xml* file. The *POM* describes the project, manages dependencies, and configures plugins for building the software.

Maven 101
https://www.geeksforgeeks.org/introduction-apache-maven-build-automation-tool-java-projects/

-> Maven helps us to resolves the dependencies we define inside the pom.xml
https://www.baeldung.com/maven [MUST READ]

Spring Chitkara University Live [SCUL] 📚⚽

Advantages of Maven for Software Developers :

https://medium.com/@leninstalinesec/benefits-of-maven-for-java-developers-8083f9d33665

Understanding Maven and POM :

https://maven.apache.org/guides/getting-started/maven-in-five-minutes.html

https://maven.apache.org/guides/introduction/introduction-to-the-pom.html

https://www.youtube.com/watch?v=793-O43F-ng

Maven Build Lifecycle and Important Commands :

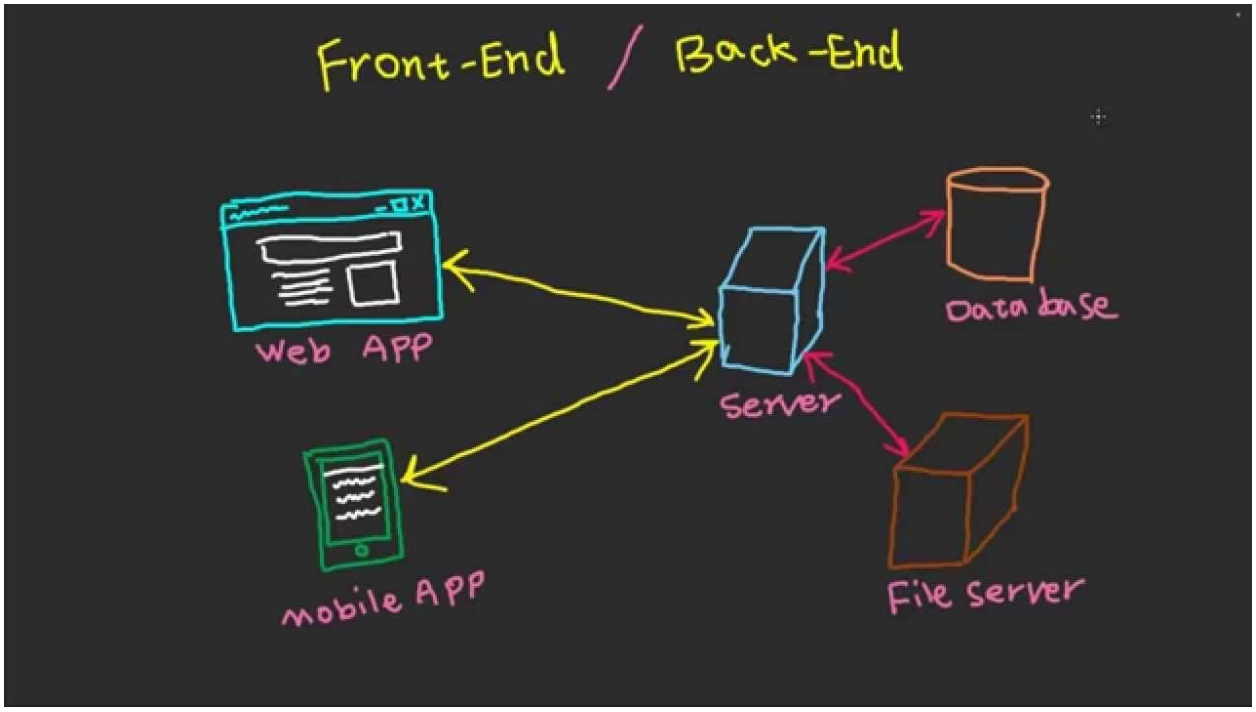
https://maven.apache.org/guides/introduction/introduction-to-the-lifecycle.html

https://www.geeksforgeeks.org/maven-lifecycle-and-basic-maven-commands/

https://medium.com/javarevisited/maven-build-lifecycle-explained-ede8494a3d48

Concept of Frontend , Backend and Database

Website vs Web Application (https://aadhario.netlify.app/ vs Facebook)
What is the **Client Server Model** ?
Why we need a Database vs storing in RAM (for persistence) [RAM vs File vs DB]



Spring Chitkara University Live [SCUL] 📚⚽

What is an API ? (eg : Search API of Google , Youtube)

Application programming interfaces, or APIs, simplify software development and innovation by enabling applications to exchange data and functionality easily and securely.

An application programming interface, or API, enables companies to open up their applications' data and functionality to external third-party developers, business partners, and internal departments within their companies.

This allows services and products to communicate with each other and leverage each other's data and functionality through a documented interface. Developers don't need to know how an API is implemented; they simply use the interface to communicate with other products and services.

Examples :

1. How Google Search Works usin search api . https://www.google.com/search?q=
2. Paytm has exposed its API for Uber , Zomatao , Swiggy , Ola to use in their application
3. Youtube API https://www.youtube.com/results?search_query=
4. Amazon API https://www.amazon.in/s?k=
5. Github API https://api.github.com/users/aadhar54

What is REST API : Representational state transfer

-stateless
-cached
-layered (example : github)

These endpoints can be hit by a web browser , mobile browser and even API Clients like Postman . https://www.postman.com/

Lets dive into Postman !

Starting your first Spring web application

Make sure you have maven installed .
Type mvn -version to check if maven is installed on your system or not .

1. Open https://start.spring.io/ in Chrome
2. Select Maven ,Java 8 , Spring version 2.5.5 and Add the dependency for Spring Web
3. Extract the created Zip folder

4. Import the folder in the IntelliJ IDE using File> New> Project from Existing sources
5. Select the pom.xml file within that folder
6. Build the project for IDE to resolve any dependency of Spring

Checkout the project structure for :

Pom.xml : here we define the dependencies
Src folder

<>Let's build an API for endpoint hi , it returns a greeting .

```
package com.example.demo;

import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class SimpleController {

    @GetMapping("/hi")
    public String sayHello(){
        return "Hello";
    }

    @GetMapping("/bye")
    public String sayBye(){
        return "Bye";
    }
}
```

Reading Logs

For successful implementation , we will be able to see the following logs . Lets understand them :

2021-10-07 07:32:55.346 INFO 10755 --- [main] com.example.demo.DemoApplication : Starting DemoApplication using Java 16.0.1 on AADHARs-MacBook-Air.local with PID 10755 (/Users/aadhar/Desktop/demo/target/classes started by aadhar in /Users/aadhar/Desktop/demo)
2021-10-07 07:32:55.349 INFO 10755 --- [main] com.example.demo.DemoApplication : No active profile set, falling back to default profiles: default

Spring Chitkara University Live [SCUL] 📚⚽

2021-10-07 07:32:57.671 INFO 10755 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8080 (http)
2021-10-07 07:32:57.694 INFO 10755 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2021-10-07 07:32:57.695 INFO 10755 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.53]
2021-10-07 07:32:57.787 INFO 10755 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2021-10-07 07:32:57.788 INFO 10755 --- [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 2275 ms
2021-10-07 07:32:58.376 INFO 10755 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path ''
2021-10-07 07:32:58.394 INFO 10755 --- [main] com.example.demo.DemoApplication : Started DemoApplication in 9.364 seconds (JVM running for 10.438)

<>Say Hello to ONLY yourself → RequestParam
Read about Spring request param : https://www.baeldung.com/spring-request-param

```
@GetMapping("sayHi2")
public String sayHiOnlyToMe(@RequestParam String name){
   if(name.equalsIgnoreCase("aadhar")){
       return "Hello Your highness "+ name;
   }else{
       return "Who are you "+ name;
   }
}
```

<> Lets take fname and name in params and return the full name

```
@GetMapping("/fullname")
public String concat(@RequestParam String p, @RequestParam String q)
{
   return "Your name is "+p+" "+q;
}
```

<>Lets now create a word search API using Hashmap as our DB

```
public class DataStore {
```

Spring Chitkara University Live [SCUL] 📚⚽

```
    private HashMap<String, String> store= new HashMap<String, String>();

    public DataStore (){
        store.put("Sachin","A great batsman");
        store.put("Ramu", " A great man");
        store.put("gfg","great wesite");
    }

    public String getFromDB(String word)
    {
        return store.get(word);
    }
}
```

Homework

IMP : Please make sure to install Postaman Client before next class .

What is difference between Java 8 vs Java 11 vs latest version of Java
https://medium.com/tribalscale/new-features-in-java-9-to-15-in-5-mins-6cc2f8f49d61

Spring vs Spring Boot https://www.javatpoint.com/spring-vs-spring-boot-vs-spring-mvc

Maven 101
https://www.geeksforgeeks.org/introduction-apache-maven-build-automation-tool-java-projects/

-> Maven helps us to resolves the dependencies we define inside the pom.xml
https://www.baeldung.com/maven [MUST READ]

Refer https://www.baeldung.com/ for reading about Spring
Official documentation : https://spring.io/

Note : All code files , notes and download links have been uploaded to Github .
Please make sure to download Java in your systems , IntelliJ as IDE and Maven in your systems as build automation tools .