

## WEEK-5 HANDS ON

### Exercise 1 - Creating Microservices for account and loan

- Created a Spring Boot application named account for account microservice
- Added A new file named “AccountController” that handles requests with url endpoint /account/{accountId}
- Set its port to 8080
- Started the application and retrieved the result in the browser
- Created a Spring Boot application named loanfor loanmicroservice
- Added A new file named “LoanController” that handles requests with url endpoint /loan/{loanId}
- Set its port to 8081
- Started the application and retrieved the result in the browser
- As a result, the two microservices are running in different ports.

### Spring Initializr for loan application

The screenshot shows the Spring Initializr web application interface. The top navigation bar includes the Spring logo and the text "spring initializr". A button labeled "To exit full screen, press and hold Esc" is visible. The main content area is divided into several sections:

- Project:** Includes radio buttons for "Gradle - Groovy", "Gradle - Kotlin", and "Maven" (selected).
- Language:** Includes radio buttons for "Java" (selected), "Kotlin", and "Groovy".
- Spring Boot:** Includes radio buttons for "4.0.0 (SNAPSHOT)", "4.0.0 (M1)", "3.5.5 (SNAPSHOT)", "3.5.4" (selected), "3.4.9 (SNAPSHOT)", and "3.4.8".
- Project Metadata:** Includes input fields for "Group" (com.cognizant), "Artifact" (account), "Name" (account), "Description" (Demo project for Spring Boot), and "Package name" (com.cognizant.account).
- Packaging:** Includes radio buttons for "Jar" (selected) and "War".
- Java:** Includes radio buttons for "24", "21", and "17" (selected).
- Dependencies:** Includes a button "ADD DEPENDENCIES... CTRL + B" and two sections: "Spring Boot DevTools" (DEVELOPER TOOLS) and "Spring Web" (WEB).

At the bottom, there are buttons for "GENERATE CTRL + G", "EXPLORE CTRL + SPACE", and a menu button "...".

### pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
      https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
```

```
<parent>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-parent</artifactId>
  <version>3.5.4</version>
  <relativePath/> <!-- lookup parent from repository -->
</parent>
<groupId>com.cognizant</groupId>
<artifactId>account</artifactId>
<version>0.0.1-SNAPSHOT</version>
<name>account</name>
<description>Demo project for Spring Boot</description>
<url/>
<licenses>
  <license/>
</licenses>
<developers>
  <developer/>
</developers>
<scm>
  <connection/>
  <developerConnection/>
  <tag/>
  <url/>
</scm>
<properties>
  <java.version>17</java.version>
</properties>
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-devtools</artifactId>
    <scope>runtime</scope>
    <optional>true</optional>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
```

```

        <artifactId>spring-boot-starter-test</artifactId>
        <scope>test</scope>
    </dependency>
</dependencies>
<build>
    <plugins>
        <plugin>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-maven-plugin</artifactId>
        </plugin>
    </plugins>
</build>
</project>

```

### AccountApplication.java

```

package com.cognizant.account;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class AccountApplication {
    public static void main(String[] args) {
        SpringApplication.run(AccountApplication.class, args);
    }
}

```

### AccountController.java

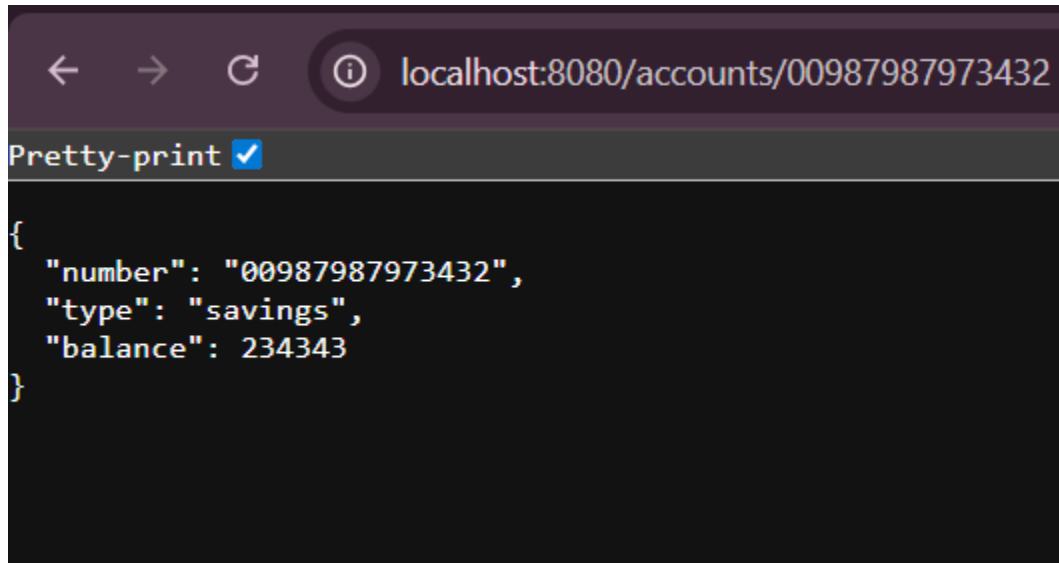
```

package com.cognizant.account;
import org.springframework.web.bind.annotation.*;

@RestController
@RequestMapping("/accounts")
public class AccountController {
    @GetMapping("/{number}")
    public Account getAccountDetails(@PathVariable String number) {
        return new Account(number, "savings", 234343);
    }
    record Account(String number, String type, int balance) {}
}

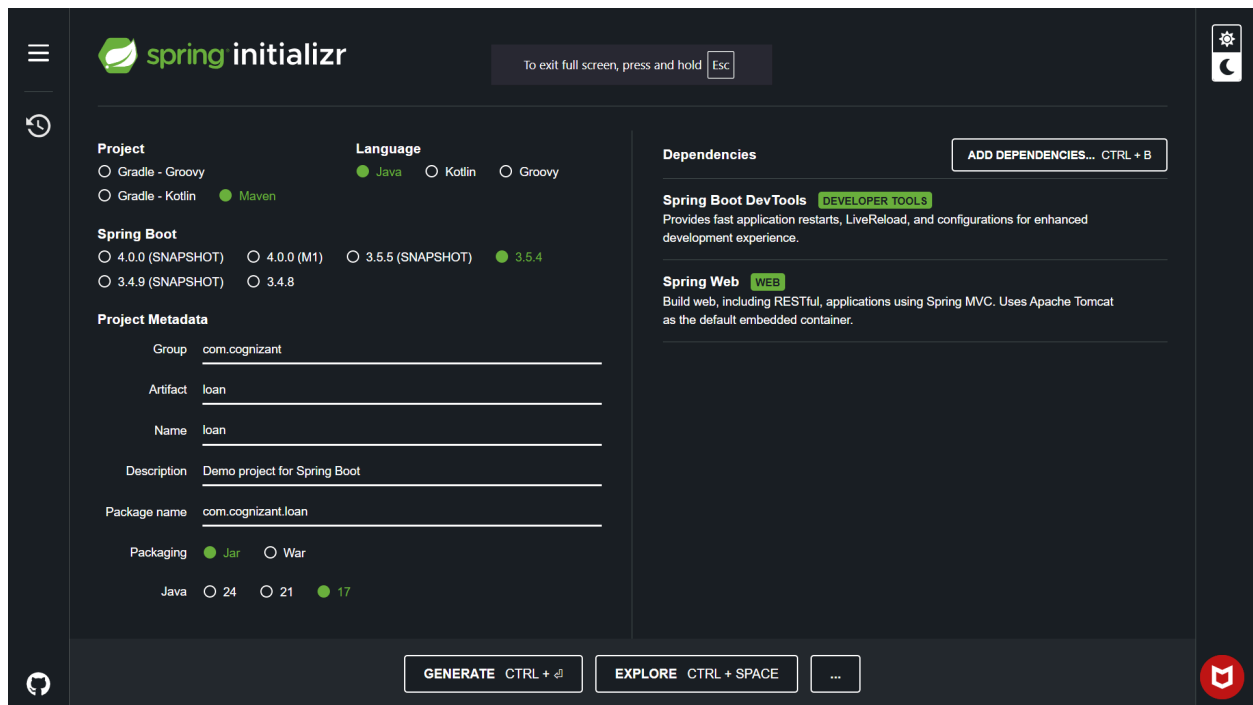
```

## Output:



```
{
  "number": "00987987973432",
  "type": "savings",
  "balance": 234343
}
```

## Spring Initializr for Loan Application



## LoanApplication.java

```
package com.cognizant.loan;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
```

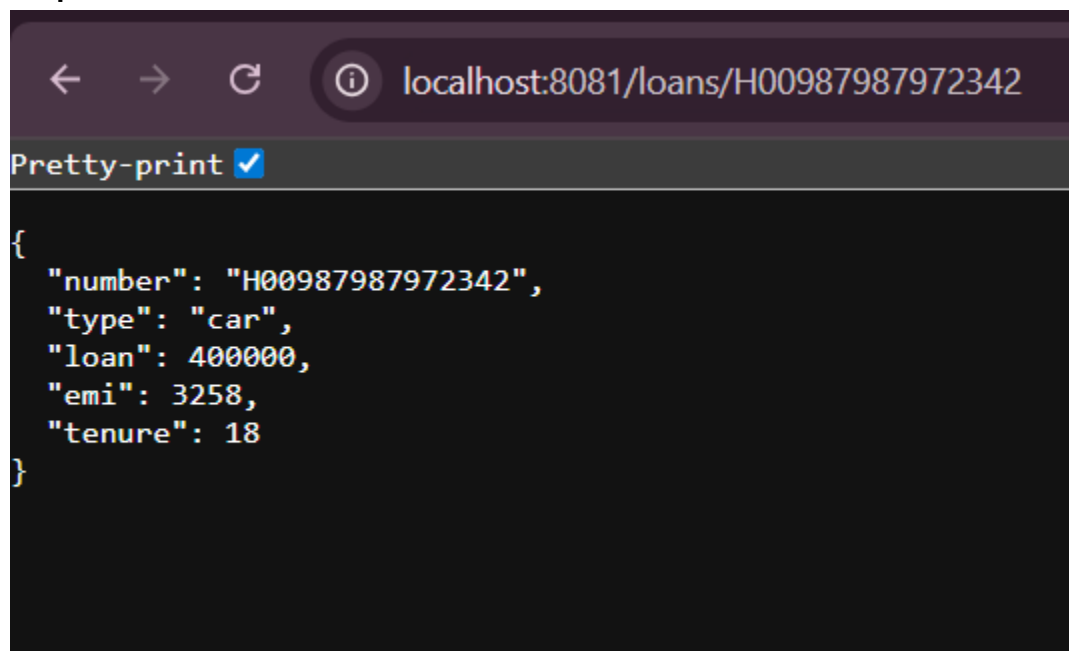
```
@SpringBootApplication
public class LoanApplication {
    public static void main(String[] args) {
        SpringApplication.run(LoanApplication.class, args);
    }
}
```

#### LoanController.java

```
package com.cognizant.loan;
import org.springframework.web.bind.annotation.*;

@RestController
@RequestMapping("/loans")
public class LoanController {
    @GetMapping("/{number}")
    public Loan getLoanDetails(@PathVariable String number) {
        return new Loan(number, "car", 400000, 3258, 18);
    }
    record Loan(String number, String type, int loan, int emi, int tenure) {}
}
```

#### Output:



localhost:8081/loans/H00987987972342

Pretty-print ☒

```
{
  "number": "H00987987972342",
  "type": "car",
  "loan": 400000,
  "emi": 3258,
  "tenure": 18
}
```