

Week-5(Microservices)

Creating Microservices for account and loan

In this hands on exercises, we will create two microservices for a bank. One microservice for handling accounts and one for handling loans. Each microservice will be a specific independent Spring RESTful Webservice maven project having it's own pom.xml. The only difference is that, instead of having both account and loan as a single application, it is split into two different applications. These webservices will be a simple service without any backend connectivity. Follow steps below to implement the two microservices: Account Microservice

- Create folder with employee id in D: drive
- Create folder named 'microservices' in the new folder created in previous step. This folder will contain all the sample projects that we will create for learning microservices.
- Open <https://start.spring.io/> in browser
- Enter form field values as specified below: o Group: com.cognizant o Artifact: account
- Select the following modules o Developer Tools > Spring Boot DevTools o Web > Spring Web
- Click generate and download the zip file
- Extract 'account' folder from the zip and place this folder in the 'microservices' folder created earlier
- Open command prompt in account folder and build using mvn clean package command
- Import this project in Eclipse and implement a controller method for getting account details based on account number. Refer specification below: o Method: GET o Endpoint: /accounts/{number} o Sample Response. Just a dummy response without any backend connectivity. { number: "00987987973432", type: "savings", balance: 234343 }
- Launch by running the application class and test the service in browser Loan Microservice
- Follow similar steps specified for Account Microservice and implement a service API to get loan account details o Method: GET o Endpoint: /loans/{number} o Sample Response. Just a dummy response without any backend connectivity. { number: "H00987987972342", type: "car", loan: 400000, emi: 3258, tenure: 18 }
- Launching this application by having account service already running
- This launch will fail with error that the bind address is already in use
- The reason is that each one of the service is launched with default port number as 8080. Account service is already using this port and it is not available for loan service.
- Include "server.port" property with value 8081 and try launching the application
- Test the service with 8081 port Now we have two microservices running on different ports. NOTE: The console window of Eclipse will have both the service console running. To switch between different consoles use the monitor icon within the console view

Account Microservice

Code:

AccountController.java

```
package com.example.account;

import org.springframework.web.bind.annotation.*;

@RestController
@RequestMapping("/accounts")
public class AccountController {

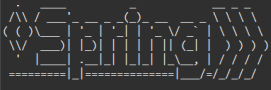
    @GetMapping("/{number}")
    public Account getAccountDetails(@PathVariable String number) {
        return new Account(number, "savings", 234343);
    }

    static class Account {
        private String number;
        private String type;
        private int balance;

        public Account(String number, String type, int balance) {
            this.number = number;
            this.type = type;
            this.balance = balance;
        }

        // Getters
        public String getNumber() { return number; }
        public String getType() { return type; }
        public int getBalance() { return balance; }
    }
}
```

Output:



```
:: Spring Boot :: (v3.5.3)

2025-07-18T12:40:55.881+05:30 INFO 10012 --- [account] [ restartedMain] com.example.account.AccountApplication : Starting AccountApplication using Java 21.0.2 with PID 10012 (E:\Cogni
2025-07-18T12:40:55.887+05:30 INFO 10012 --- [account] [ restartedMain] com.example.account.AccountApplication : No active profile set, falling back to 1 default profile: "default"
2025-07-18T12:40:55.957+05:30 INFO 10012 --- [account] [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : Devtools property defaults active! Set 'spring.devtools.add-properties:
2025-07-18T12:40:55.958+05:30 INFO 10012 --- [account] [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : For additional web related logging consider setting the 'logging.level'
2025-07-18T12:40:57.288+05:30 INFO 10012 --- [account] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8080 (http)
2025-07-18T12:40:57.314+05:30 INFO 10012 --- [account] [ restartedMain] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2025-07-18T12:40:57.314+05:30 INFO 10012 --- [account] [ restartedMain] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.42]
2025-07-18T12:40:57.380+05:30 INFO 10012 --- [account] [ restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2025-07-18T12:40:57.382+05:30 INFO 10012 --- [account] [ restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1422 ms
2025-07-18T12:40:58.095+05:30 INFO 10012 --- [account] [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : LiveReload server is running on port 35729
2025-07-18T12:40:58.192+05:30 INFO 10012 --- [account] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 8080 (http) with context path '/'
2025-07-18T12:40:58.211+05:30 INFO 10012 --- [account] [ restartedMain] com.example.account.AccountApplication : Started AccountApplication in 3.001 seconds (process running for 3.665
2025-07-18T12:41:04.634+05:30 INFO 10012 --- [account] [ nio-8080-exec-1] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring DispatcherServlet 'dispatcherServlet'
2025-07-18T12:41:04.634+05:30 INFO 10012 --- [account] [ nio-8080-exec-1] o.s.web.servlet.DispatcherServlet : Initializing Servlet 'dispatcherServlet'
2025-07-18T12:41:04.636+05:30 INFO 10012 --- [account] [ nio-8080-exec-1] o.s.web.servlet.DispatcherServlet : Completed initialization in 1 ms
```

GET	http://localhost:8080/accounts/00987987973432	Send	Status: 200 OK	Size: 61 Bytes	Time: 183 ms
Query	Headers 2	Auth	Body	Tests	Pre Run
Query Parameters			Response	Headers 4	Cookies
<input type="checkbox"/> parameter			Results		
value			Docs		
			1 { 2 "number": "00987987973432", 3 "type": "savings", 4 "balance": 234343 5 }		

Loan Microservice

Code:

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);
    }

    static class Loan {
        private String number;
        private String type;
        private int loann;
        private int emi;
        private int tenure;

        public Loan(String number, String type, int loann, int emi, int tenure) {
            this.number = number;
            this.type = type;
            this.loann = loann;
            this.emi = emi;
            this.tenure = tenure;
        }

        // Getters
        public String getNumber() { return number; }
        public String getType() { return type; }
        public int getLoann() { return loann; }
        public int getEmi() { return emi; }
        public int getTenure() { return tenure; }
    }
}
```

Output:

```

  ____  __
 / ___/  / /
/ /   /  / /
/ /___/  / /
/_____/  / /
         /_/

:: Spring Boot ::
                (v3.5.3)

2025-07-18T12:58:03.942+05:30 INFO 2372 --- [loan] [ restartedMain] com.cognizant.loan.LoanApplication : Starting LoanApplication using Java 21.0.2 with PID 2372 (E:\Cognizant-Java FSE\Week 5\Code\181
2025-07-18T12:58:03.945+05:30 INFO 2372 --- [loan] [ restartedMain] com.cognizant.loan.LoanApplication : No active profile set, falling back to 1 default profile: "default"
2025-07-18T12:58:03.997+05:30 INFO 2372 --- [loan] [ restartedMain] o.e.devtools.properties.defaults.PostProcessor : Devtools property defaults active! Set 'spring.devtools.add-properties' to 'false' to disable
2025-07-18T12:58:04.011+05:30 INFO 2372 --- [loan] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : For additional web related logging consider setting the 'logging.level.web' property to 'DEBUG'
2025-07-18T12:58:04.022+05:30 INFO 2372 --- [loan] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8081 (http)
2025-07-18T12:58:04.922+05:30 INFO 2372 --- [loan] [ restartedMain] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2025-07-18T12:58:04.922+05:30 INFO 2372 --- [loan] [ restartedMain] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.42]
2025-07-18T12:58:04.951+05:30 INFO 2372 --- [loan] [ restartedMain] o.a.c.c.c.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2025-07-18T12:58:04.952+05:30 INFO 2372 --- [loan] [ restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 954 ms
2025-07-18T12:58:05.300+05:30 INFO 2372 --- [loan] [ restartedMain] o.s.b.s.d.OptionalLiveReloadServer : LiveReload server is running on port 35729
2025-07-18T12:58:05.346+05:30 INFO 2372 --- [loan] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 8081 (http) with context path '/'
2025-07-18T12:58:05.359+05:30 INFO 2372 --- [loan] [ restartedMain] com.cognizant.loan.LoanApplication : Started LoanApplication in 1.847 seconds (process running for 2.268)

```

The screenshot displays the REST Client application interface. On the left, the 'Query' tab is active, showing a GET request to `http://localhost:8081/loans/H00987987972342`. The 'Query Parameters' section is empty. On the right, the 'Response' tab is active, showing the status '200 OK', size '79 Bytes', and time '199 ms'. The response body is a JSON object: `{ "number": "H00987987972342", "type": "car", "loann": 400000, "emi": 3258, "tenure": 18 }`.