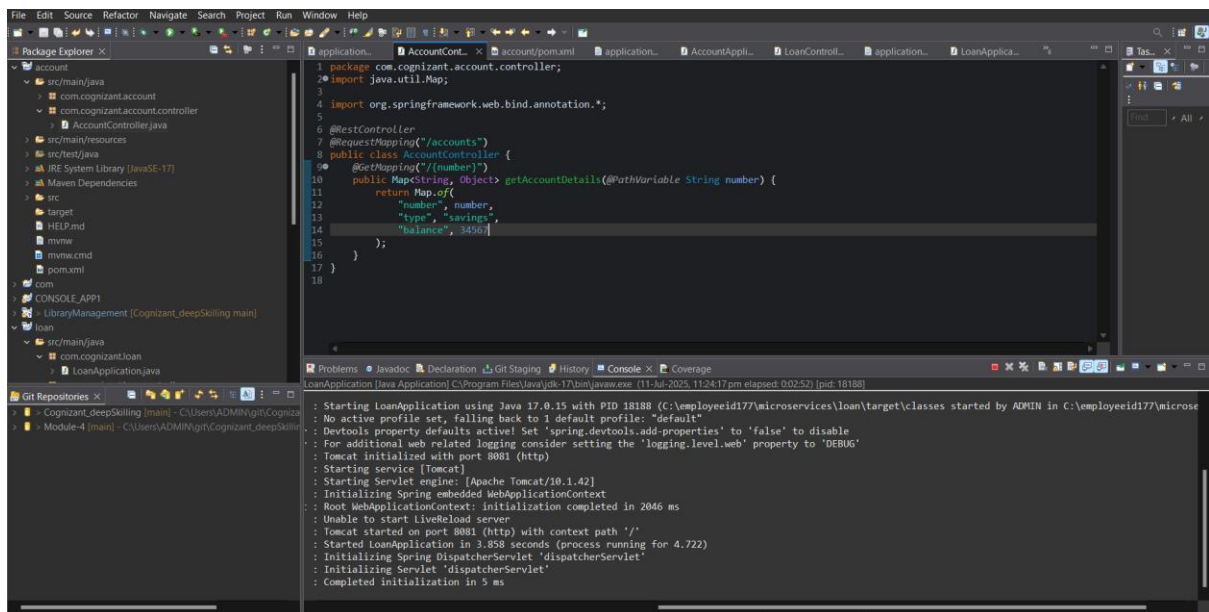


WEEK-05

Creating Microservices for account and loan

ACCOUNTS:

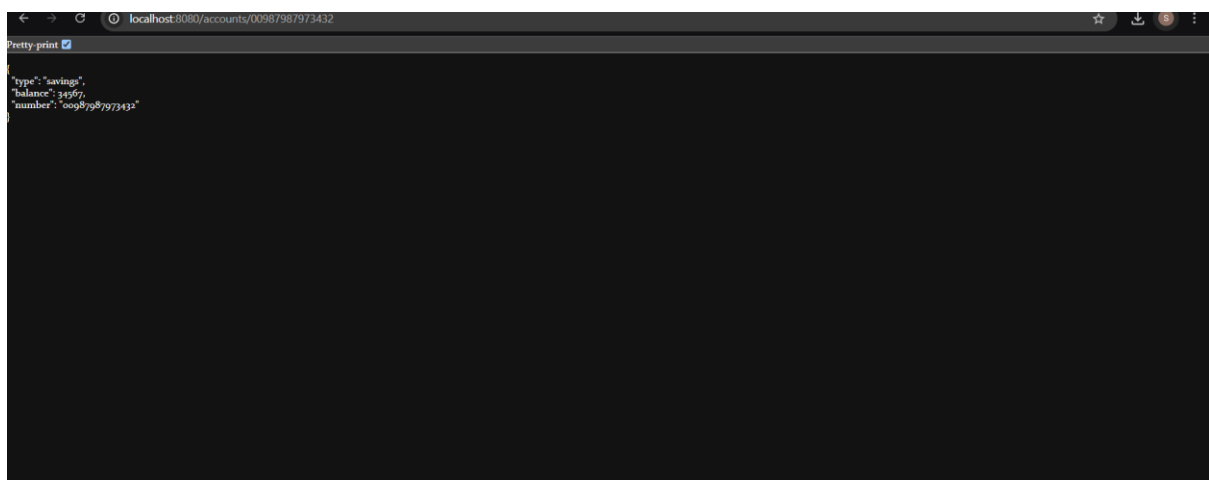


The screenshot shows an IDE with the following components:

- Package Explorer:** Shows the project structure with packages `com.cognizant.account` and `com.cognizant.loan`.
- Source Editor:** Displays the `AccountController.java` file with the following code:

```
1 package com.cognizant.account.controller;
2 import java.util.Map;
3
4 import org.springframework.web.bind.annotation.*;
5
6 @RestController
7 @RequestMapping("/accounts")
8 public class AccountController {
9     @GetMapping("/{number}")
10    public Map<String, Object> getAccountDetails(@PathVariable String number) {
11        return Map.of(
12            "number", number,
13            "type", "savings",
14            "balance", 34567
15        );
16    }
17 }
18
```
- Console:** Shows the output of the application startup:

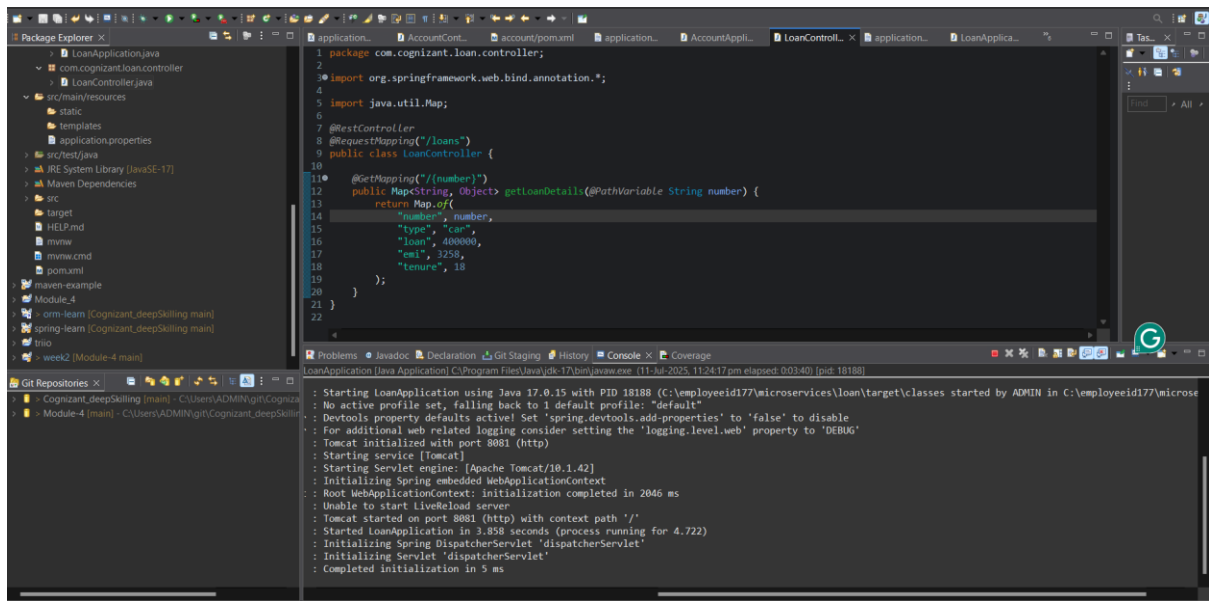
```
: Starting LoanApplication using Java 17.0.15 with PID 18188 (C:\employeeid177\microservices\loan\target\classes started by ADMIN in C:\employeeid177\microservices\loan\target\classes)
: No active profile set, falling back to 1 default profile: "default"
: Devtools property defaults active! Set 'spring.devtools.add-properties' to 'false' to disable
: For additional web related logging consider setting the 'logging.level.web' property to 'DEBUG'
: Tomcat initialized with port 8081 (http)
: Starting service [Tomcat]
: Starting Servlet engine: [Apache Tomcat/10.1.42]
: Initializing Spring embedded WebApplicationContext
: Root WebApplicationContext: initialization completed in 2046 ms
: Unable to start liveReload server
: Tomcat started on port 8081 (http) with context path '/'
: Started LoanApplication in 3.858 seconds (process running for 4.722)
: Initializing Spring DispatcherServlet 'dispatcherServlet'
: Initializing Servlet 'dispatcherServlet'
: Completed initialization in 5 ms
```



The screenshot shows a web browser at the URL `localhost:8080/accounts/009879879/3432`. The response is displayed in a JSON format:

```
{
  "type": "savings",
  "balance": 34567,
  "number": "00987987973432"
}
```

LOANS:

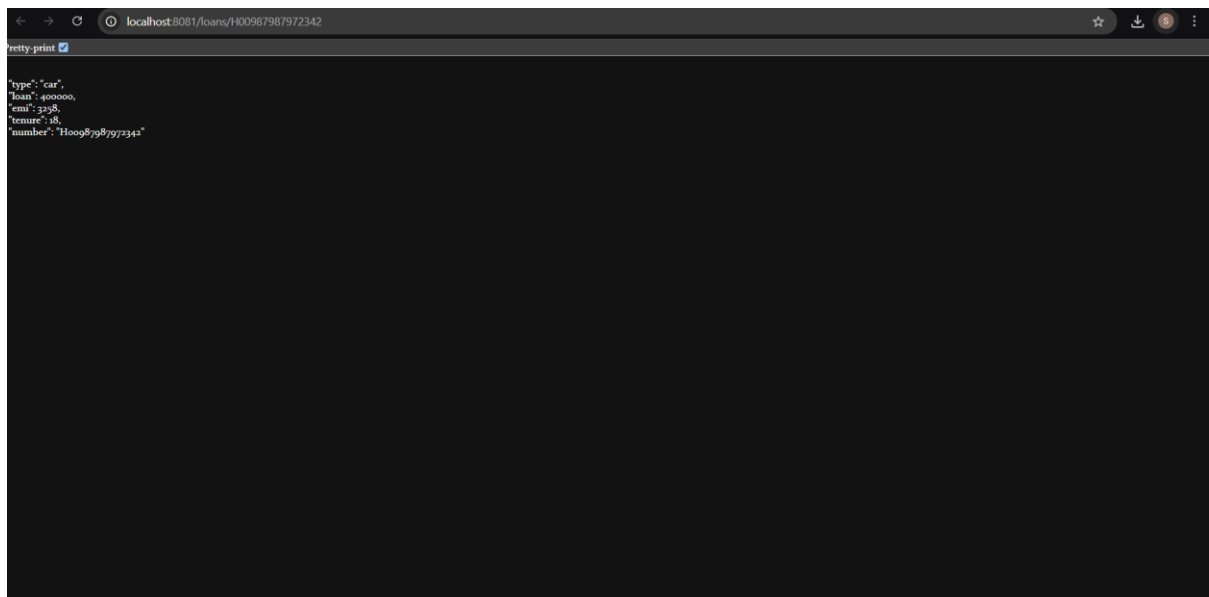


The screenshot shows an IDE with the following components:

- Package Explorer:** Shows the project structure with packages like `com.cognizant.loan.controller` and `src/main/resources`.
- Source Editor:** Displays the `LoanController.java` file with the following code:

```
1 package com.cognizant.loan.controller;
2
3 import org.springframework.web.bind.annotation.*;
4
5 import java.util.Map;
6
7 @RestController
8 @RequestMapping("/loans")
9 public class LoanController {
10
11     @GetMapping("/{number}")
12     public Map<String, Object> getLoanDetails(@PathVariable String number) {
13         return Map.of(
14             "number", number,
15             "type", "car",
16             "loan", 400000,
17             "emi", 3258,
18             "tenure", 18
19         );
20     }
21 }
22
```
- Console:** Shows the application startup logs:

```
Starting LoanApplication using Java 17.0.15 with PID 18188 (C:\employeeid177\microservices\loan\target\classes started by ADMIN in C:\employeeid177\microse
No active profile set, falling back to 1 default profile: "default"
Devtools property defaults active! Set 'spring.devtools.add-properties' to 'false' to disable
For additional web related logging consider setting the 'logging.level.web' property to 'DEBUG'
Tomcat initialized with port 8081 (http)
Starting Service [Tomcat]
Starting Servlet engine: [Apache Tomcat/10.1.42]
Initializing Spring embedded WebApplicationContext
Root WebApplicationContext: initialization completed in 2046 ms
Unable to start liveReload server
Tomcat started on port 8081 (http) with context path '/'
Started LoanApplication in 3.858 seconds (process running for 4.722)
Initializing Spring DispatcherServlet 'dispatcherServlet'
Initializing Servlet 'dispatcherServlet'
Completed initialization in 5 ms
```



The screenshot shows a web browser with the address bar set to `localhost:8081/loans/H00987987972342`. The browser is displaying the response from a REST client, which is a JSON object:

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