**Microservices with Spring Boot 3 and Spring Cloud**

**Exercise: Creating Microservices for account and loan**

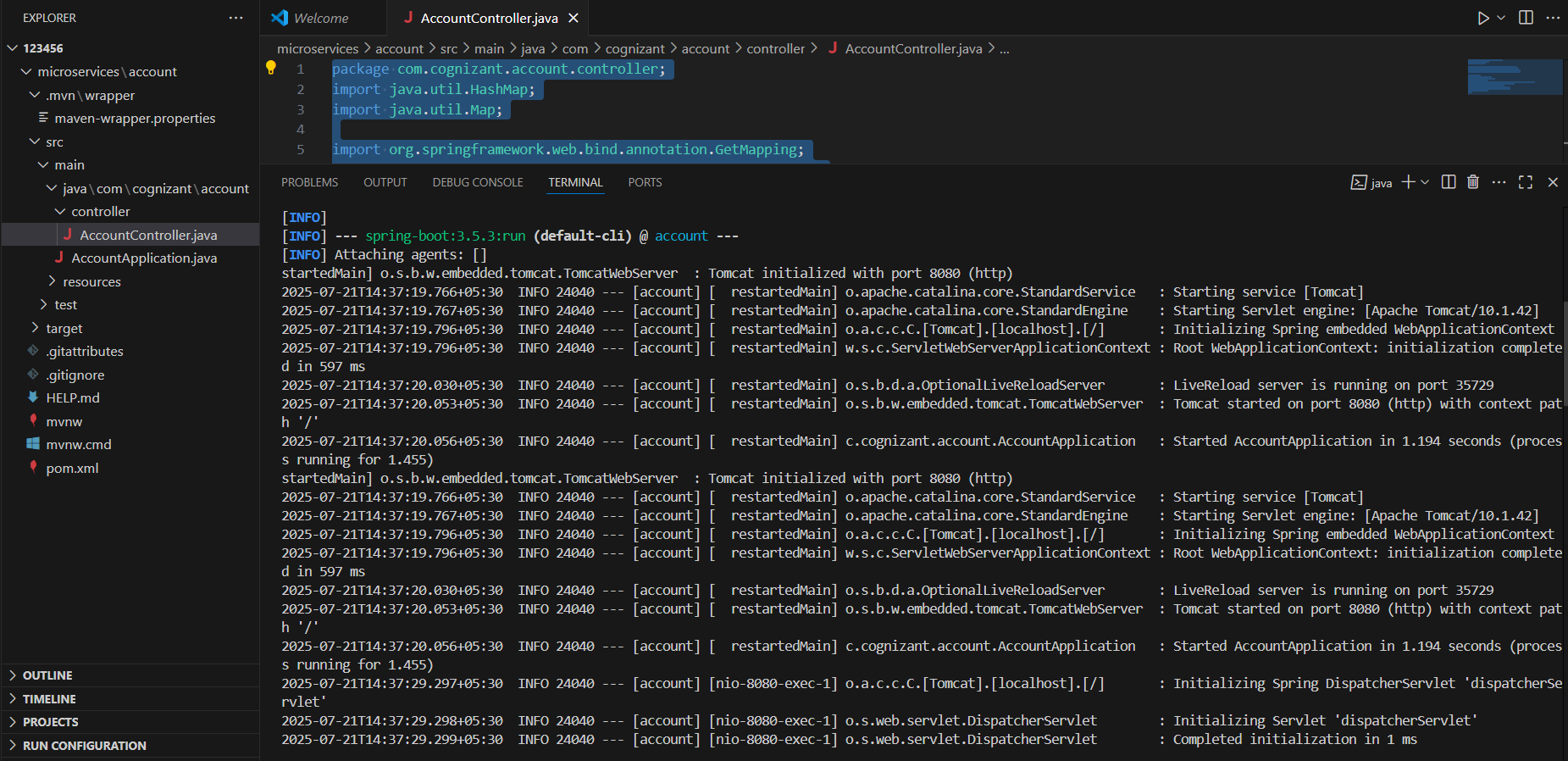
1.Generate Spring Boot project accounts.

2. Create controller file

|  |
| --- |
| AccountController.java  package com.cognizant.account.controller;  import java.util.HashMap;  import java.util.Map;  import org.springframework.web.bind.annotation.GetMapping;  import org.springframework.web.bind.annotation.PathVariable;  import org.springframework.web.bind.annotation.RequestMapping;  import org.springframework.web.bind.annotation.RestController;  @RestController  @RequestMapping("/accounts")  public class AccountController {  @GetMapping("/{number}")      public Map<String, Object> getAccountDetails(@PathVariable String number) {          Map<String, Object> response = new HashMap<>();          response.put("number", number);          response.put("type", "savings");          response.put("balance", 234343);          return response;      }  } |

3. Run code

**OUTPUT:**





1.Generate Spring Boot project loan.

2.Create controller file.

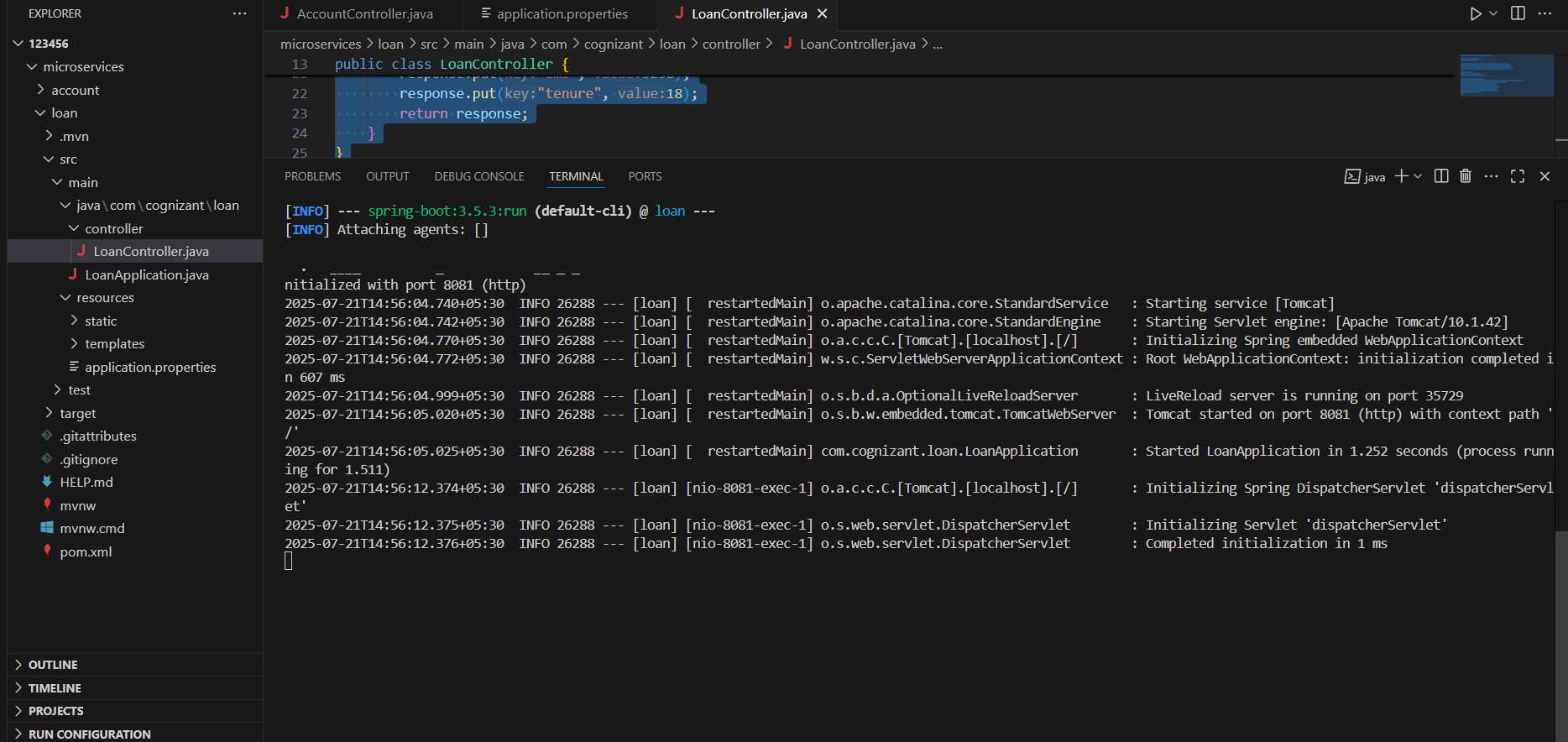
|  |
| --- |
| LoanController.java  package com.cognizant.loan.controller;  import java.util.HashMap;  import java.util.Map;  import org.springframework.web.bind.annotation.GetMapping;  import org.springframework.web.bind.annotation.PathVariable;  import org.springframework.web.bind.annotation.RequestMapping;  import org.springframework.web.bind.annotation.RestController;  @RestController  @RequestMapping("/loans")  public class LoanController {      @GetMapping("/{number}")      public Map<String, Object> getLoanDetails(@PathVariable String number) {          Map<String, Object> response = new HashMap<>();          response.put("number", number);          response.put("type", "car");          response.put("loan", 400000);          response.put("emi", 3258);          response.put("tenure", 18);          return response;      }  } |

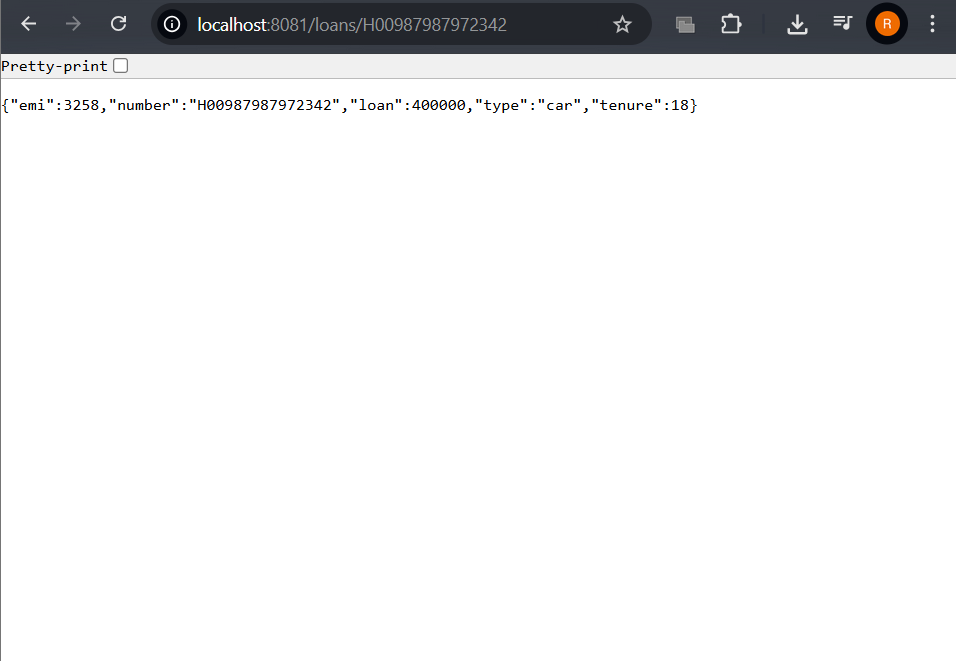
3.Change port number

|  |
| --- |
| application.properties  spring.application.name=loan  server.port=8081 |

4. Run code

**Output:**

****

****