**WEEK-5**

(Implemented in Eclipse)

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

1. **2. Microservices with API gateway**

**Creating Microservices for account and loan**

**Account**

AccountController.java

**package** com.cognizant.account.controller;

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

**import** java.util.Map;

@RestController

@RequestMapping("/accounts")

**public** **class** AccountController {

@GetMapping("/{number}")

**public** Map<String, Object> getAccount(@PathVariable String number) {

**return** Map.*of*(

"number", number,

"type", "savings",

"balance", 234343

);

}

}

Application.properties

spring.application.name=account

server.port=8081

AccountApplication.java

**package** com.cognizant.account;

**import** org.springframework.boot.SpringApplication;

**import** org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

**public** **class** AccountApplication {

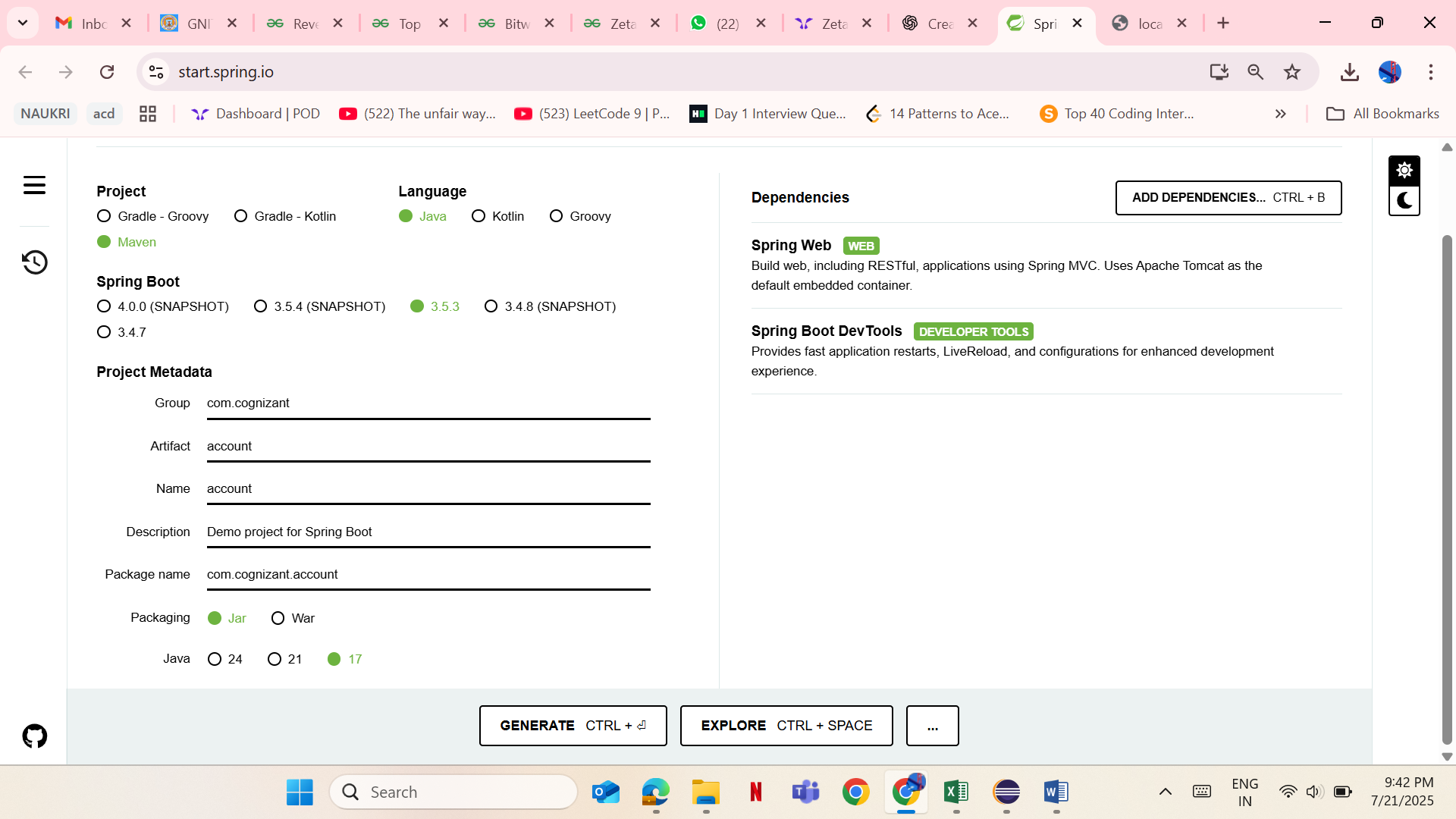
**public** **static** **void** main(String[] args) {

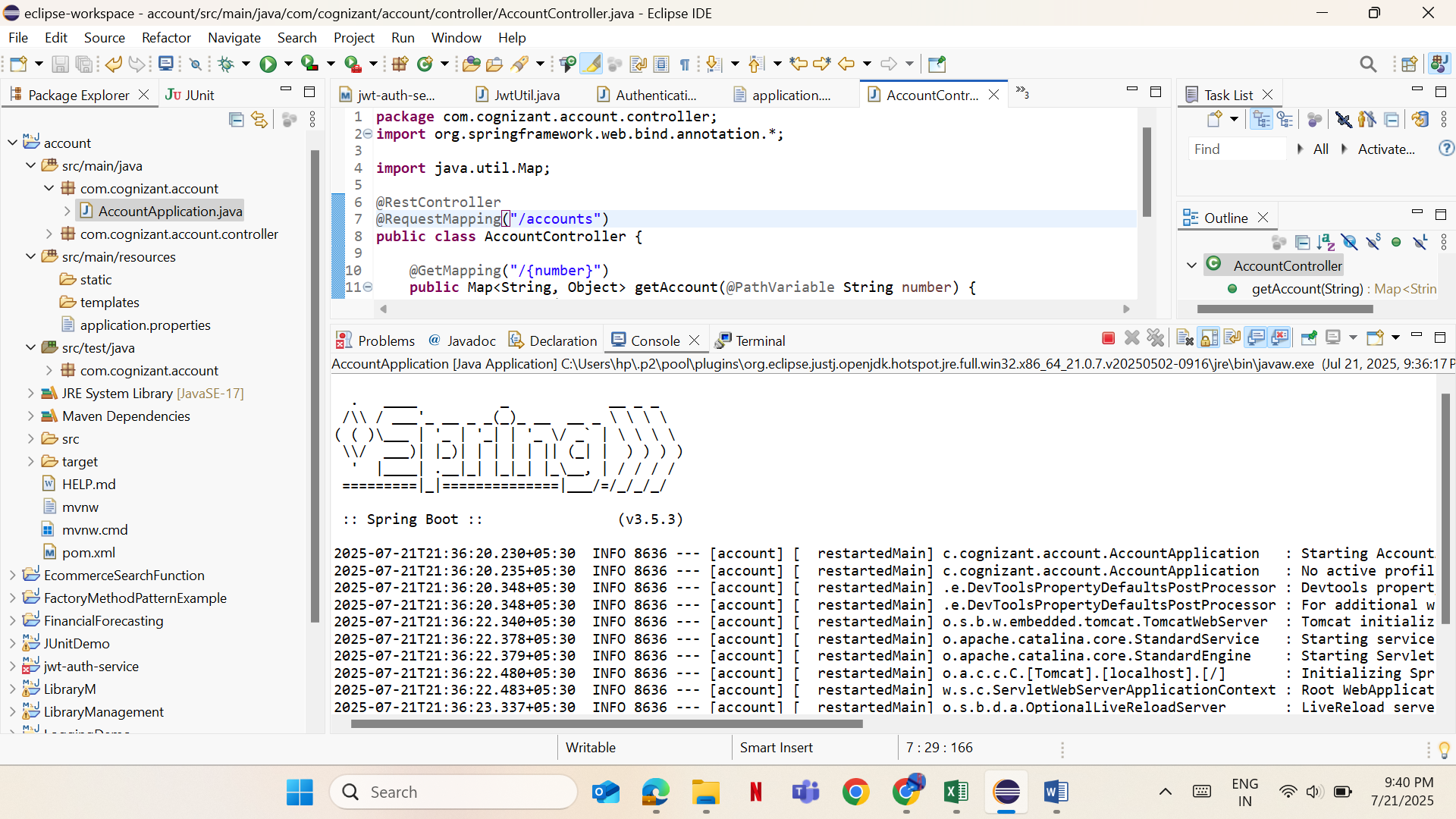
SpringApplication.*run*(AccountApplication.**class**, args);

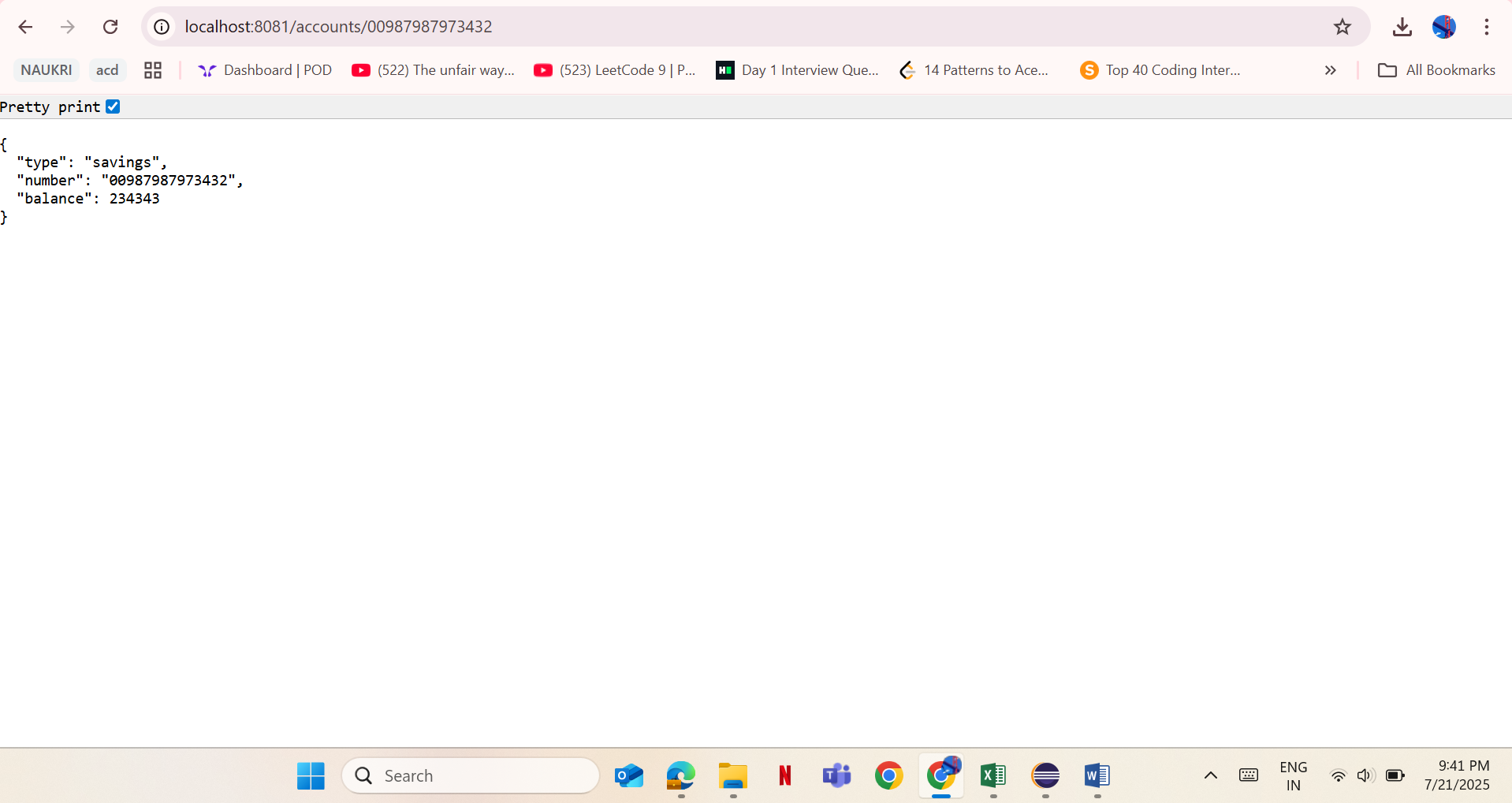
}

}

**Output:**

****





**Loan**

LoanController.java

**package** com.cognizant.loan.controller;

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

**import** java.util.Map;

@RestController

@RequestMapping("/loans")

**public** **class** LoanController {

@GetMapping("/{number}")

**public** Map<String, Object> getLoan(@PathVariable String number) {

**return** Map.*of*(

"number", number,

"type", "car",

"loan", 400000,

"emi", 3258,

"tenure", 18

);

}

}

Application.properties

spring.application.name=account

server.port=8080

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);

}

}

Output:

