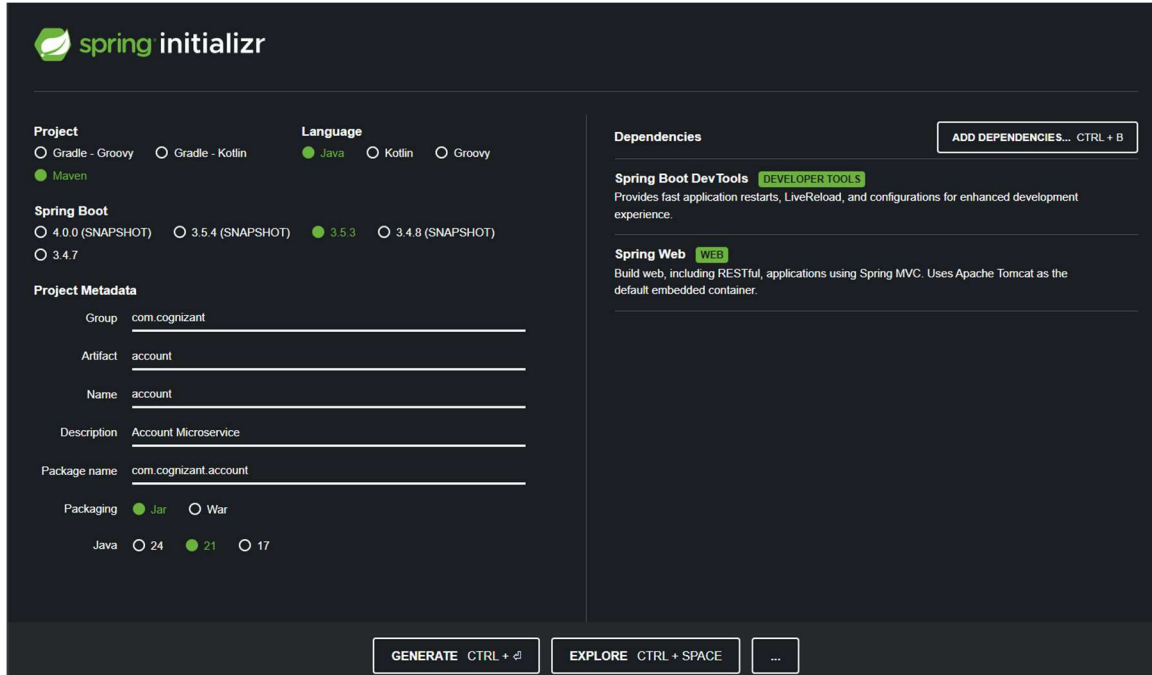


Creating Microservices for account and loan

Account:

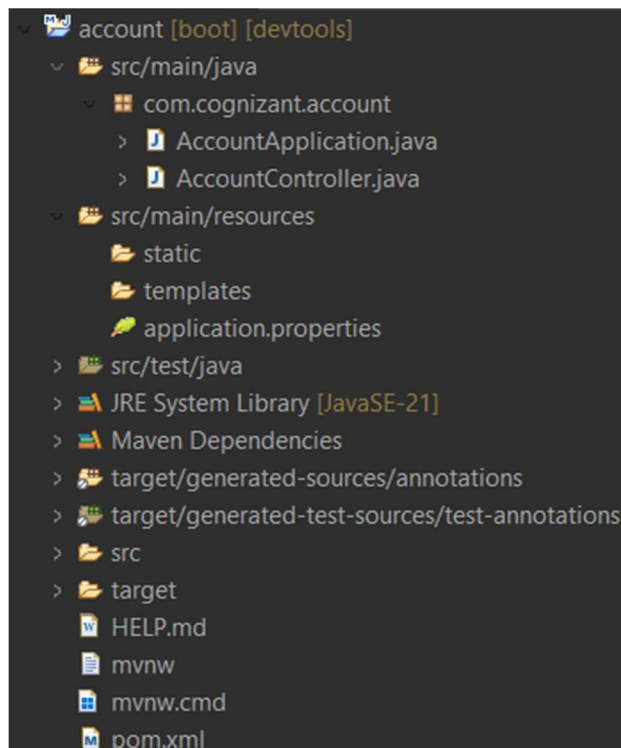
Open <https://start.spring.io/> in browser

And added the following requirements



The screenshot shows the Spring Initializr web application interface. The 'Project' section has 'Maven' selected. The 'Language' section has 'Java' selected. The 'Spring Boot' section has '3.5.3' selected. The 'Project Metadata' section shows the following fields: Group (com.cognizant), Artifact (account), Name (account), Description (Account Microservice), Package name (com.cognizant.account), Packaging (Jar), and Java version (21). The 'Dependencies' section shows 'Spring Boot Dev Tools' and 'Spring Web' selected. At the bottom, there are buttons for 'GENERATE', 'EXPLORE', and a menu icon.

And then generated it and extracted it into the required folder which was created in the D drive and then imported it into eclipse



Created AccountController.java class in the package

```
1 package com.cognizant.account;
2
3 import org.springframework.web.bind.annotation.*;
4
5 @RestController
6 @RequestMapping("/accounts")
7 public class AccountController {
8
9     @GetMapping("/{number}")
10    public Account getAccountDetails(@PathVariable String number) {
11        return new Account(number, "savings", 234343);
12    }
13 }
14
15 class Account {
16     private String number;
17     private String type;
18     private double balance;
19
20     public Account(String number, String type, double balance) {
21         this.number = number;
22         this.type = type;
23         this.balance = balance;
24     }
25
26     // Getters
27     public String getNumber() { return number; }
28     public String getType() { return type; }
29     public double getBalance() { return balance; }
30 }
31
```

Updated the server port number to 8081 in application.properties

```
1 spring.application.name=account
2 server.port=8081
3
```

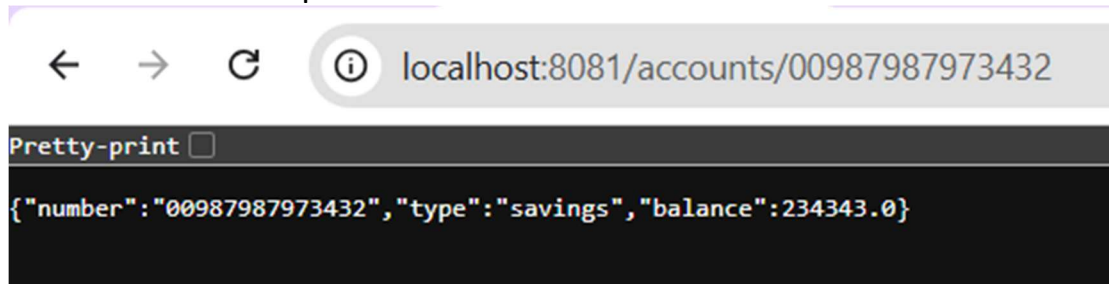
Then run the AccountApplication.java class which consists of main method

```
1 package com.cognizant.account;
2
3 import org.springframework.boot.SpringApplication;
4
5
6 @SpringBootApplication
7 public class AccountApplication {
8
9     public static void main(String[] args) {
10         SpringApplication.run(AccountApplication.class, args);
11     }
12
13 }
14 |
```

Then it showed that tomcat running at port 8081

And then checked at <http://localhost:8081/accounts/00987987973432>

Then it showed output



The screenshot shows a web browser window with the address bar displaying `localhost:8081/accounts/00987987973432`. Below the address bar, there is a section labeled "Pretty-print" with a checkbox. The output of the API call is displayed as a JSON object: `{"number": "00987987973432", "type": "savings", "balance": 234343.0}`.

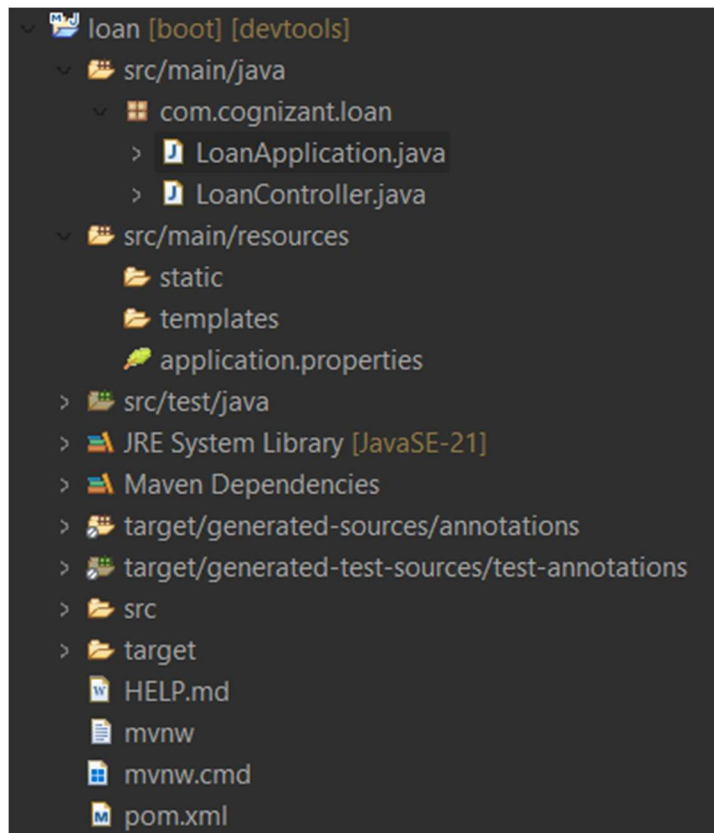
Loan:

Open <https://start.spring.io/> in browser

And added the following requirements

The screenshot shows the Spring Initializr web application interface. The 'Project' section has 'Maven' selected. The 'Language' section has 'Java' selected. The 'Spring Boot' section has '3.5.3' selected. The 'Project Metadata' section shows 'Group' as 'com.cognizant', 'Artifact' as 'loan', 'Name' as 'loan', 'Description' as 'Account Microservice', and 'Package name' as 'com.cognizant.loan'. The 'Packaging' section has 'Jar' selected. The 'Dependencies' section has 'Spring Boot DevTools' and 'Spring Web' selected. The 'GENERATE' button is visible at the bottom.

And then generated it and extracted it into the required folder which was created in the D drive and then imported it into eclipse



Created LoanController.java class in the package

```
1 package com.cognizant.loan;
2
3 import org.springframework.web.bind.annotation.*;
4
5 @RestController
6 @RequestMapping("/loans")
7 public class LoanController {
8
9     @GetMapping("/{number}")
10    public Loan getLoanDetails(@PathVariable String number) {
11        return new Loan(number, "car", 400000, 3258, 18);
12    }
13 }
14
15 class Loan {
16     private String number;
17     private String type;
18     private double loan;
19     private int emi;
20     private int tenure;
21
22     public Loan(String number, String type, double loan, int emi, int tenure) {
23         this.number = number;
24         this.type = type;
25         this.loan = loan;
26         this.emi = emi;
27         this.tenure = tenure;
28     }
29
30     // Getters
31     public String getNumber() { return number; }
32     public String getType() { return type; }
33     public double getLoan() { return loan; }
34     public int getEmi() { return emi; }
35     public int getTenure() { return tenure; }
36 }
```

Updated the server port number to 8082 in application.properties

```
1 spring.application.name=loan
2 server.port=8082
```

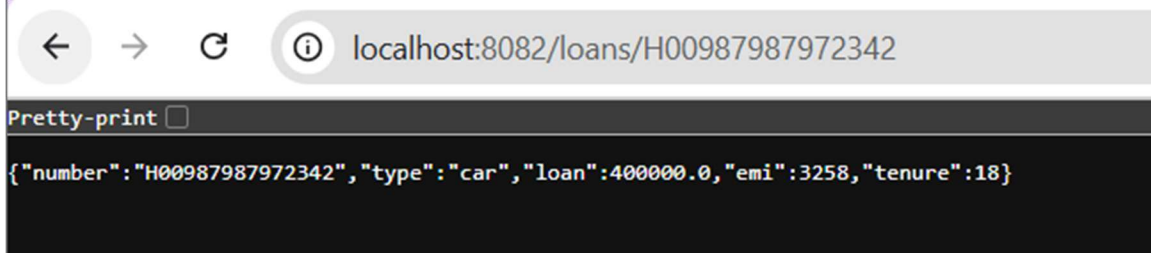
Then run the LoanApplication.java class which consists of main method

```
1 package com.cognizant.loan;
2
3 import org.springframework.boot.SpringApplication;
4
5
6 @SpringBootApplication
7 public class LoanApplication {
8
9     public static void main(String[] args) {
10         SpringApplication.run(LoanApplication.class, args);
11     }
12
13 }
14
```

Then it showed that tomcat running at port 8082

And then checked at <http://localhost:8082/loans/H00987987972342>

Then it showed output



The screenshot shows a web browser window with the address bar displaying `localhost:8082/loans/H00987987972342`. Below the address bar, there is a "Pretty-print" checkbox. The main content area of the browser displays a JSON object: `{"number": "H00987987972342", "type": "car", "loan": 400000.0, "emi": 3258, "tenure": 18}`.