Microservices with API gateway

Creating Microservices for account and loan

Solution:

**Create Account Microservice**

**Step 1: Go to Spring Initializr**

* **Group**: com.cognizant
* **Artifact**: account
* **Dependencies**:
  + Spring Boot DevTools
  + Spring Web

**Step 2: Download and Setup**

* Click **Generate**, download the ZIP
* Extract it

**Step 3: Import to eclipse**

**Step 4: Create Controller Class**

**Path: src/main/java/com/cognizant/account/AccountController.java**

package com.cognizant.account.controller;

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;

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);

}

}

**Run the Application**

* Run LoanApplication.java
* Visit:  
  http://localhost:8080/accounts/12345

**Create REST Controller**

In src/main/java/com/cognizant/loan, create:

**LoanController.java**

package com.cognizant.loan;

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

@RestController

@RequestMapping("/loans")

public class LoanController {

@GetMapping("/{number}")

public Loan getLoan(@PathVariable String number) {

return new Loan(number, "car", 400000, 3258, 18);

}

class Loan {

private String number;

private String type;

private double loan;

private double emi;

private int tenure;

public Loan(String number, String type, double loan, double emi, int tenure) {

this.number = number;

this.type = type;

this.loan = loan;

this.emi = emi;

this.tenure = tenure;

}

public String getNumber() { return number; }

public String getType() { return type; }

public double getLoan() { return loan; }

public double getEmi() { return emi; }

public int getTenure() { return tenure; }

}

}

**Run the Application**

* Run LoanApplication.java
* Visit:  
  <http://localhost:8081/loans/H00987987972342>

Output:

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.