Microservices Assignment

# Topic: Creating Microservices for Account and Loan

This document contains the implementation of two Spring Boot microservices developed as part of the CTS Digital Nurture Deepskilling program. Each microservice functions independently to simulate basic banking operations: one for handling account details and another for loan information.

# 1. Account Microservice

This microservice returns dummy account data based on the account number.

Sample API Endpoint:

Sample Output:

{  
 "number": "00987987973432",  
 "type": "savings",  
 "balance": 234343  
}

Java Code:

@RestController  
@RequestMapping("/accounts")  
public class AccountController {  
  
 @GetMapping("/{number}")  
 public Account getAccountDetails(@PathVariable String number) {  
 return new Account(number, "savings", 234343);  
 }  
  
 static class Account {  
 private String number;  
 private String type;  
 private double balance;  
  
 public Account(String number, String type, double balance) {  
 this.number = number;  
 this.type = type;  
 this.balance = balance;  
 }  
  
 public String getNumber() { return number; }  
 public String getType() { return type; }  
 public double getBalance() { return balance; }  
 }  
}

# 2. Loan Microservice

This microservice returns dummy loan information based on a loan account number.

Sample API Endpoint:

Sample Output:

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

Java Code:

@RestController  
@RequestMapping("/loans")  
public class LoanController {  
  
 @GetMapping("/{number}")  
 public Loan getLoanDetails(@PathVariable String number) {  
 return new Loan(number, "car", 400000, 3258, 18);  
 }  
  
 static 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; }  
 }  
}

# 3. Program Output

The following section shows the simulated output results from running each microservice in the browser.

## Account Microservice Output

URL: http://localhost:8080/accounts/00987987973432

{  
 "number": "00987987973432",  
 "type": "savings",  
 "balance": 234343  
}

## Loan Microservice Output

URL: http://localhost:8081/loans/H00987987972342

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