**Lab #10**

**Task 1: Create Micro service for bank**

using System.Collections.Generic;

using System.Web.Mvc;

Model code:

public class Account

{

public int AccountId { get; set; }

public string AccountNumber { get; set; }

public decimal Balance { get; set; }

public string AccountType { get; set; }

}

Controller Class:

using Microsoft.AspNetCore.Mvc;

using System.Collections.Generic;

using System.Linq;

namespace BankingMicroservice.Controllers

{

[Route("api/[controller]")]

[ApiController]

public class AccountController : ControllerBase

{

private static List<Account> accounts = new List<Account>

{

new Account { AccountId = 1, AccountNumber = "57987545", Balance = 1000, AccountType = "Debit" },

new Account { AccountId = 2, AccountNumber = "48487849", Balance = 5000, AccountType = "ATM" }

};

[HttpGet]

public IActionResult GetAccounts()

{

return Ok(accounts);

}

[HttpGet("{accountId}")]

public IActionResult GetAccountById(int accountId)

{

var account = accounts.FirstOrDefault(a => a.AccountId == accountId);

if (account == null)

{

return NotFound();

}

return Ok(account);

}

[HttpPost]

public IActionResult CreateAccount(Account account)

{

account.AccountId = accounts.Count + 1;

accounts.Add(account);

return CreatedAtAction(nameof(GetAccountById), new { accountId = account.AccountId }, account);

}

}

**Task 2 : Create Micro service for products**

**Model code :**

using System.Collections.Generic;

using System.Web.Mvc;

public class Product

{

public int Id { get; set; }

public string Name { get; set; }

public decimal Price { get; set; }

public string Description { get; set; }

public string Category { get; set; }

}

Controller Class:

using Microsoft.AspNetCore.Mvc;

using System.Collections.Generic;

namespace ProductMicroservice.Controllers

{

[Route("api/[controller]")]

[ApiController]

public class ProductController : ControllerBase

{

private static List<Product> products = new List<Product>

{

new Product { Id = 1, Name = "Smartphone", Price = 6559.99m, Description = "A high computable device", Category = "Electronics" },

new Product { Id = 2, Name = "Laptop", Price = 9789.99m, Description = "A portable computer", Category = "Electronics" }

};

[HttpGet]

public IActionResult GetProducts()

{

return Ok(products);

}

[HttpGet("{id}")]

public IActionResult GetProductById(int id)

{

var product = products.Find(p => p.Id == id);

if (product == null)

{

return NotFound();

}

return Ok(product);

}

[HttpPost]

public IActionResult CreateProduct(Product product)

{

products.Add(product);

return CreatedAtAction(nameof(GetProductById), new { id = product.Id }, product);

}

}