**Assignment 19#**

**Q1) Create new Spring Boot App as Part of Microservice (Or Monolithic Application)**

**Product**

**Customercare**

**Shipping**

**Promotions-App**

**Entity**

**Id**

**Month**

**Cost**

**DurationDays**

**Mediapartner**

**Controller (CRUD Operation -GET POST DELETE)**

**Repo**

**Ans: - Application.properties**

server.port=9008

spring.datasource.username=root

spring.datasource.password=admin

spring.datasource.url=jdbc:mysql://localhost:3306/lulu02

spring.jpa.properties.hibernate.dialect = org.hibernate.dialect.MySQL8Dialect

spring.jpa.hibernate.ddl-auto = update

spring.jpa.show-sql: true

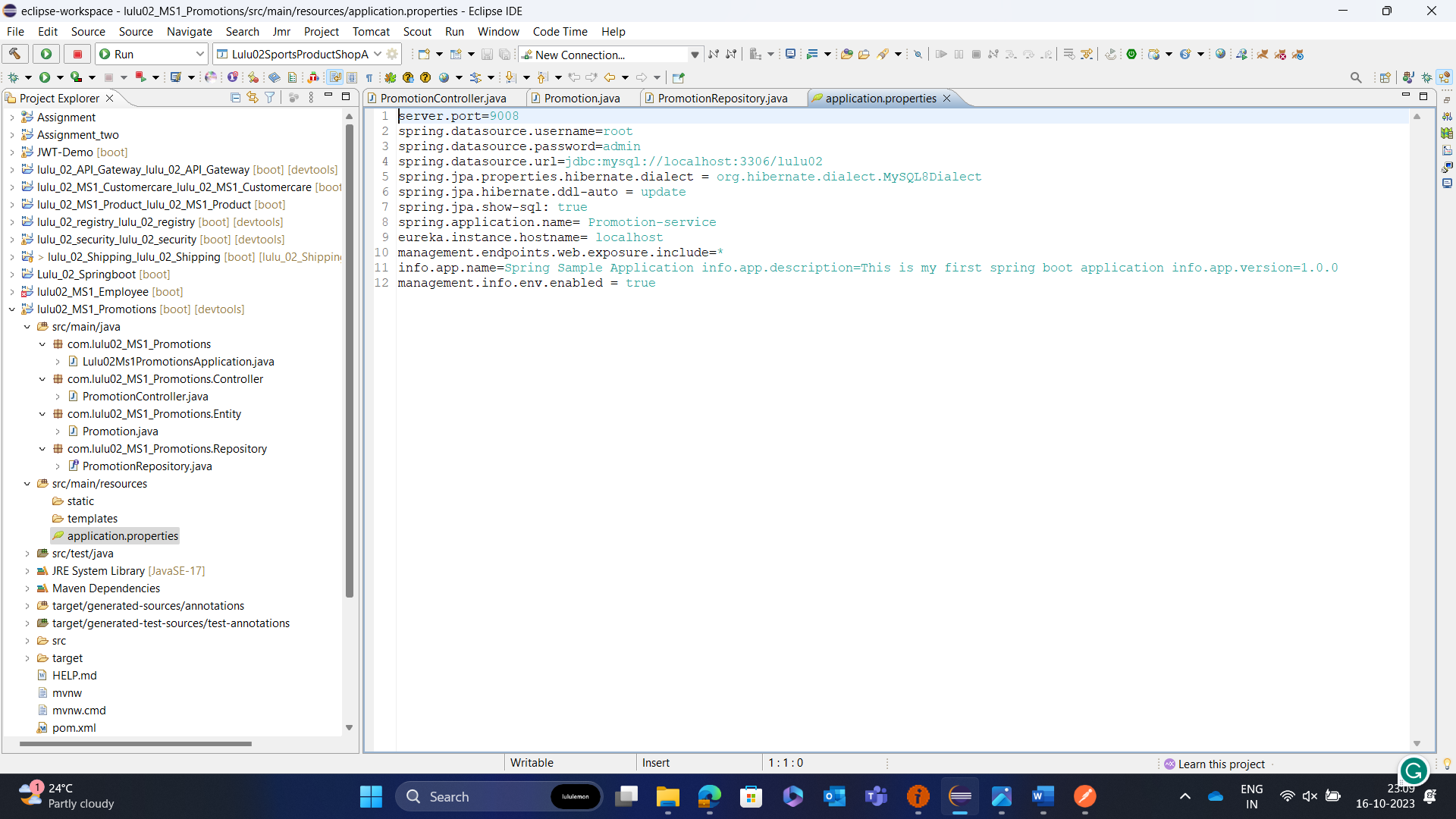
spring.application.name= Promotion-service

eureka.instance.hostname= localhost

management.endpoints.web.exposure.include=\*

info.app.name=Spring Sample Application info.app.description=This is my first spring boot application info.app.version=1.0.0

management.info.env.enabled = true

****

**Repository**

**package com.lulu02\_MS1\_Promotions.Repository;**

**import org.springframework.data.jpa.repository.JpaRepository;**

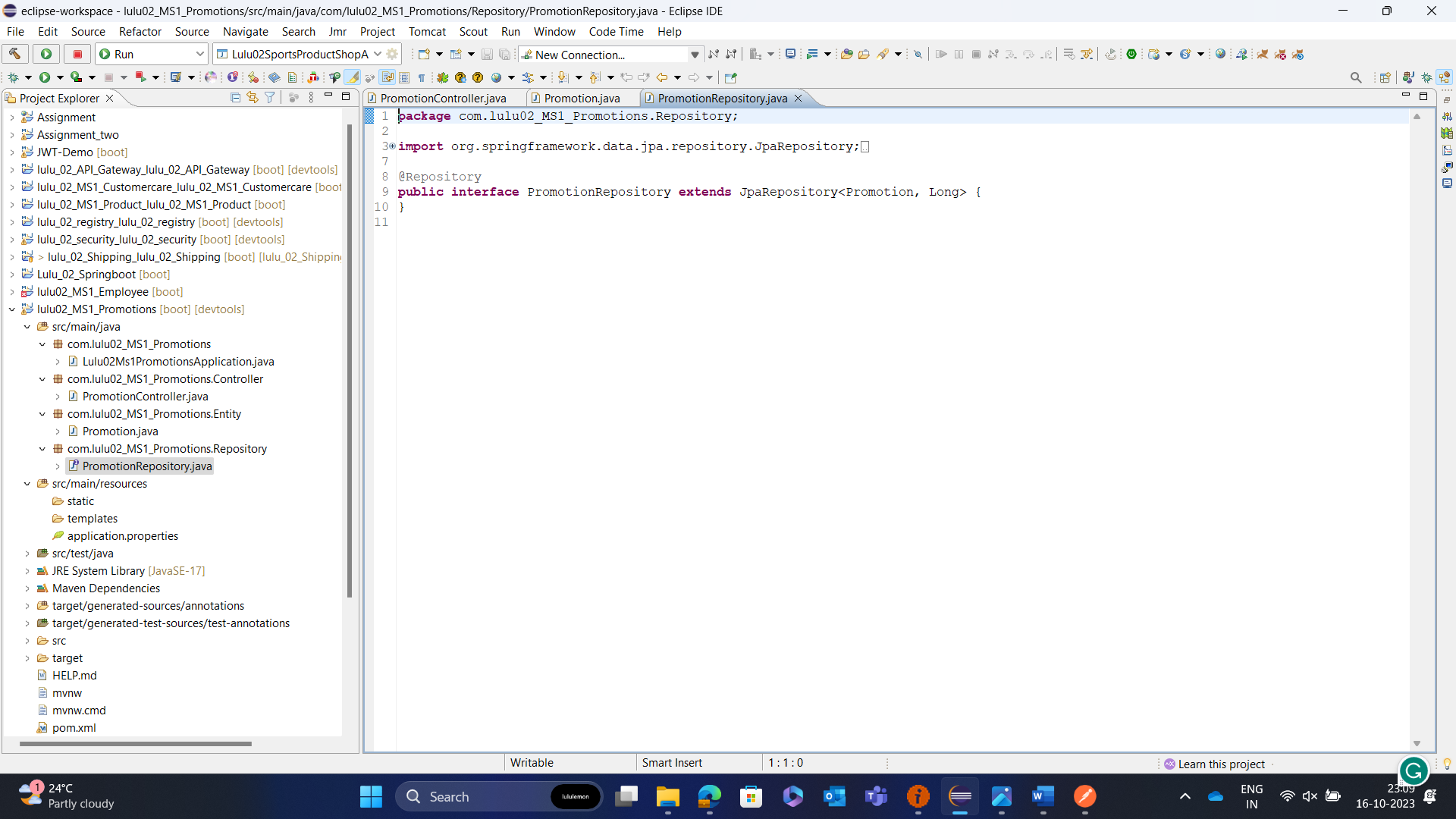
**import org.springframework.stereotype.Repository;**

**import com.lulu02\_MS1\_Promotions.Entity.Promotion;**

**@Repository**

**public interface PromotionRepository extends JpaRepository<Promotion, Long> {**

**}**

****

**Entity**

**package com.lulu02\_MS1\_Promotions.Entity;**

**import jakarta.persistence.Entity;**

**import jakarta.persistence.GeneratedValue;**

**import jakarta.persistence.GenerationType;**

**import jakarta.persistence.Id;**

**import jakarta.persistence.Table;**

**@Entity**

**@Table(name = "promotion")**

**public class Promotion {**

**@Id**

**@GeneratedValue(strategy = GenerationType.IDENTITY)**

**int id;**

**String month;**

**double cost;**

**int duration\_Day;**

**String media\_Partner;**

**public int getId() {**

**return id;**

**}**

**public void setId(int id) {**

**this.id = id;**

**}**

**public String getMonth() {**

**return month;**

**}**

**public void setMonth(String month) {**

**this.month = month;**

**}**

**public double getCost() {**

**return cost;**

**}**

**public void setCost(double cost) {**

**this.cost = cost;**

**}**

**public int getDuration\_Day() {**

**return duration\_Day;**

**}**

**public void setDuration\_Day(int duration\_Day) {**

**this.duration\_Day = duration\_Day;**

**}**

**public String getMedia\_Partner() {**

**return media\_Partner;**

**}**

**public void setMedia\_Partner(String media\_Partner) {**

**this.media\_Partner = media\_Partner;**

**}**

**}**

**A screenshot of a computer

Description automatically generated**

**Controller**

**package com.lulu02\_MS1\_Promotions.Controller;**

**import java.util.List;**

**import org.springframework.beans.factory.annotation.Autowired;**

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

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

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

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

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

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

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

**import com.lulu02\_MS1\_Promotions.Entity.Promotion;**

**import com.lulu02\_MS1\_Promotions.Repository.PromotionRepository;**

**@RestController**

**@RequestMapping("/promotion")**

**public class PromotionController {**

**@Autowired**

**private PromotionRepository promotionRepository;**

**@GetMapping**

**public List<Promotion> getAllPromotions() {**

**return promotionRepository.findAll();**

**}**

**@GetMapping("/{id}")**

**public Promotion getPromotionById(@PathVariable Long id) {**

**return promotionRepository.findById(id)**

**.orElseThrow(() -> new RuntimeException("Promotion not found with id: " + id));**

**}**

**@PostMapping**

**public Promotion createPromotion(@RequestBody Promotion promotion) {**

**return promotionRepository.save(promotion);**

**}**

**@DeleteMapping("/{id}")**

**public void deletePromotion(@PathVariable Long id) {**

**promotionRepository.deleteById(id);**

**}**

**}**

**A computer screen shot of a computer screen

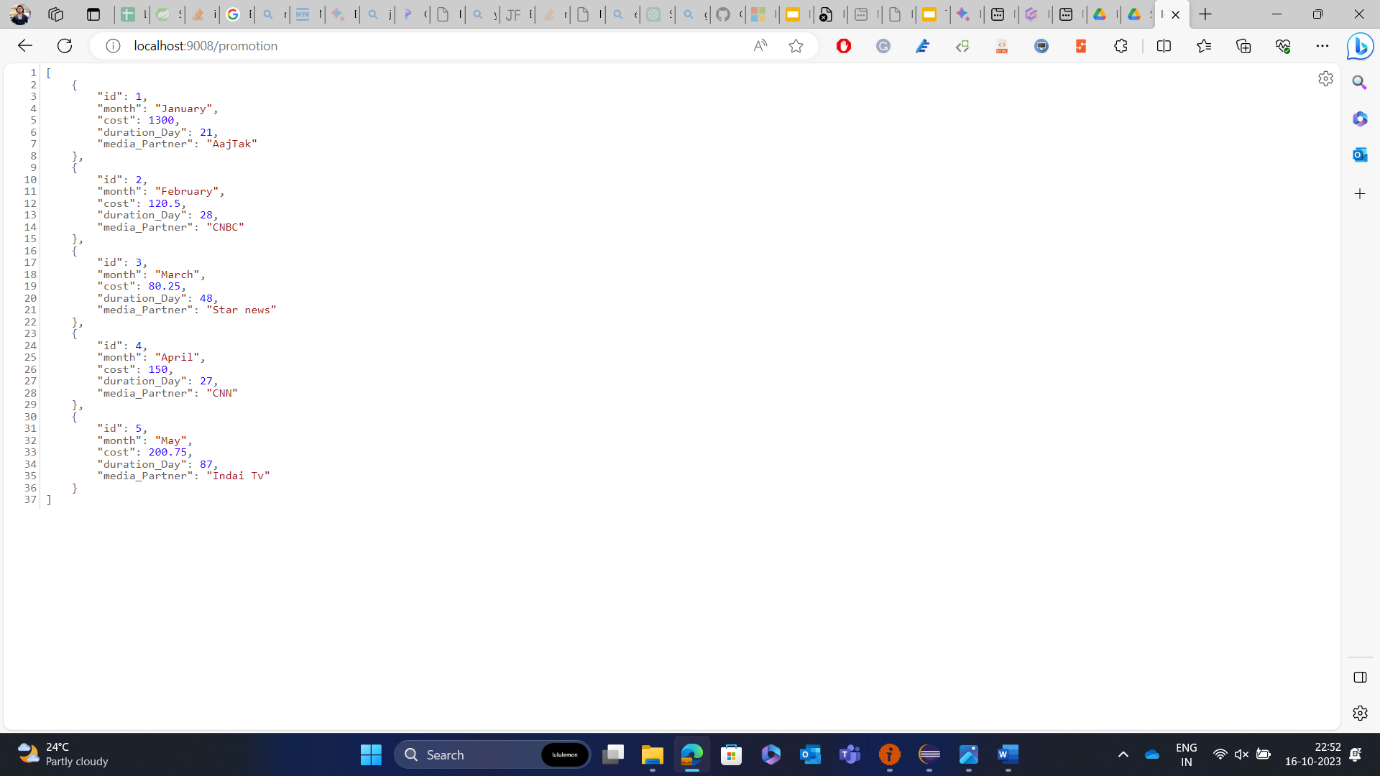
Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated**