**Exercise 7: Employee Management System - Enabling Entity Auditing**

**Business Scenario:**

Implement auditing to track the creation and modification of employees and departments.

**Instructions:**

1. **Entity Auditing:**
   * Enable auditing in your application by configuring auditing properties.
   * Use annotations like **@CreatedBy, @LastModifiedBy, @CreatedDate**, and **@LastModifiedDate**.

Solution

STEP 1

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

implementation 'org.springframework.boot:spring-boot-starter-data-jpa'

implementation 'org.springframework.boot:spring-boot-starter-web'

STEP 2 Enable JPA Auditing

import org.springframework.context.annotation.Configuration;

import org.springframework.data.jpa.repository.config.EnableJpaAuditing;

@Configuration

@EnableJpaAuditing

public class ApplicationConfig {

}

STEP 3 Implement Auditable Entities

import org.springframework.data.annotation.CreatedDate;

import org.springframework.data.annotation.LastModifiedDate;

import org.springframework.data.jpa.domain.support.AuditingEntityListener;

import javax.persistence.Column;

import javax.persistence.EntityListeners;

import javax.persistence.MappedSuperclass;

import java.time.Instant;

@MappedSuperclass

@EntityListeners(AuditingEntityListener.class)

public abstract class Auditable {

@CreatedDate

@Column(updatable = false)

private Instant createdDate;

@LastModifiedDate

private Instant lastModifiedDate;

// Getters and Setters

public Instant getCreatedDate() {

return createdDate;

}

public void setCreatedDate(Instant createdDate) {

this.createdDate = createdDate;

}

public Instant getLastModifiedDate() {

return lastModifiedDate;

}

public void setLastModifiedDate(Instant lastModifiedDate) {

this.lastModifiedDate = lastModifiedDate;

}

}

STEP 4

import javax.persistence.Entity;

import javax.persistence.Id;

@Entity

public class Employee extends Auditable {

@Id

private Long id;

private String name;

private String department;

// Getters and Setters

}

Document

import javax.persistence.Entity;

import javax.persistence.Id;

@Entity

public class Department extends Auditable {

@Id

private Long id;

private String name;

// Getters and Setters

}

Configure Spring Security

import org.springframework.data.domain.AuditorAware;

import org.springframework.security.core.Authentication;

import org.springframework.security.core.context.SecurityContextHolder;

import org.springframework.stereotype.Component;

import java.util.Optional;

@Component

public class AuditorAwareImpl implements AuditorAware<String> {

@Override

public Optional<String> getCurrentAuditor() {

Authentication authentication = SecurityContextHolder.getContext().getAuthentication();

if (authentication != null && authentication.isAuthenticated()) {

return Optional.of(authentication.getName());

}

return Optional.empty();

}

}

import org.springframework.data.annotation.CreatedBy;

import org.springframework.data.annotation.LastModifiedBy;

import org.springframework.data.jpa.domain.support.AuditingEntityListener;

import javax.persistence.Column;

import javax.persistence.EntityListeners;

import javax.persistence.MappedSuperclass;

import java.time.Instant;

@MappedSuperclass

@EntityListeners(AuditingEntityListener.class)

public abstract class Auditable {

@CreatedBy

@Column(updatable = false)

private String createdBy;

@LastModifiedBy

private String lastModifiedBy;

@CreatedDate

@Column(updatable = false)

private Instant createdDate;

@LastModifiedDate

private Instant lastModifiedDate;

// Getters and Setters

public String getCreatedBy() {

return createdBy;

}

public void setCreatedBy(String createdBy) {

this.createdBy = createdBy;

}

public String getLastModifiedBy() {

return lastModifiedBy;

}

public void setLastModifiedBy(String lastModifiedBy) {

this.lastModifiedBy = lastModifiedBy;

}

public Instant getCreatedDate() {

return createdDate;

}

public void setCreatedDate(Instant createdDate) {

this.createdDate = createdDate;

}

public Instant getLastModifiedDate() {

return lastModifiedDate;

}

public void setLastModifiedDate(Instant lastModifiedDate) {

this.lastModifiedDate = lastModifiedDate;

}

}