**Spring Data JPA - Quick Example**

**Code:**

**User.java**

package com.example.demo;

import jakarta.persistence.Entity;

import jakarta.persistence.GeneratedValue;

import jakarta.persistence.GenerationType;

import jakarta.persistence.Id;

@Entity

public class User {

    @Id

    @GeneratedValue(strategy = GenerationType.IDENTITY)

    private Long id;

    private String name;

    private String email;

    // Constructors

    public User() {}

    public User(String name, String email) {

        this.name = name;

        this.email = email;

    }

    // Getters and setters

    public Long getId() { return id; }

    public void setId(Long id) { this.id = id; }

    public String getName() { return name; }

    public void setName(String name) { this.name = name; }

    public String getEmail() { return email; }

    public void setEmail(String email) { this.email = email; }

}

**UserRepository.java**

package com.example.demo;

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

// Spring auto-implements this interface at runtime

public interface UserRepository extends JpaRepository<User, Long> {

}

**HelloController.java**

package com.example.demo;

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

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

import java.util.List;

@RestController

public class HelloController {

    private final GreetingService greetingService;

    private final UserRepository userRepository;

    @Autowired

    public HelloController(GreetingService greetingService, UserRepository userRepository) {

        this.greetingService = greetingService;

        this.userRepository = userRepository;

    }

    @GetMapping("/")

    public String hello() {

        return greetingService.getGreetingMessage();

    }

    @GetMapping("/welcome")

    public String welcome() {

        return greetingService.getWelcomeMessage();

    }

    // List all users

    @GetMapping("/users")

    public List<User> getUsers() {

        return userRepository.findAll();

    }

    // Create a new user

    @PostMapping("/users")

    public User createUser(@RequestBody User user) {

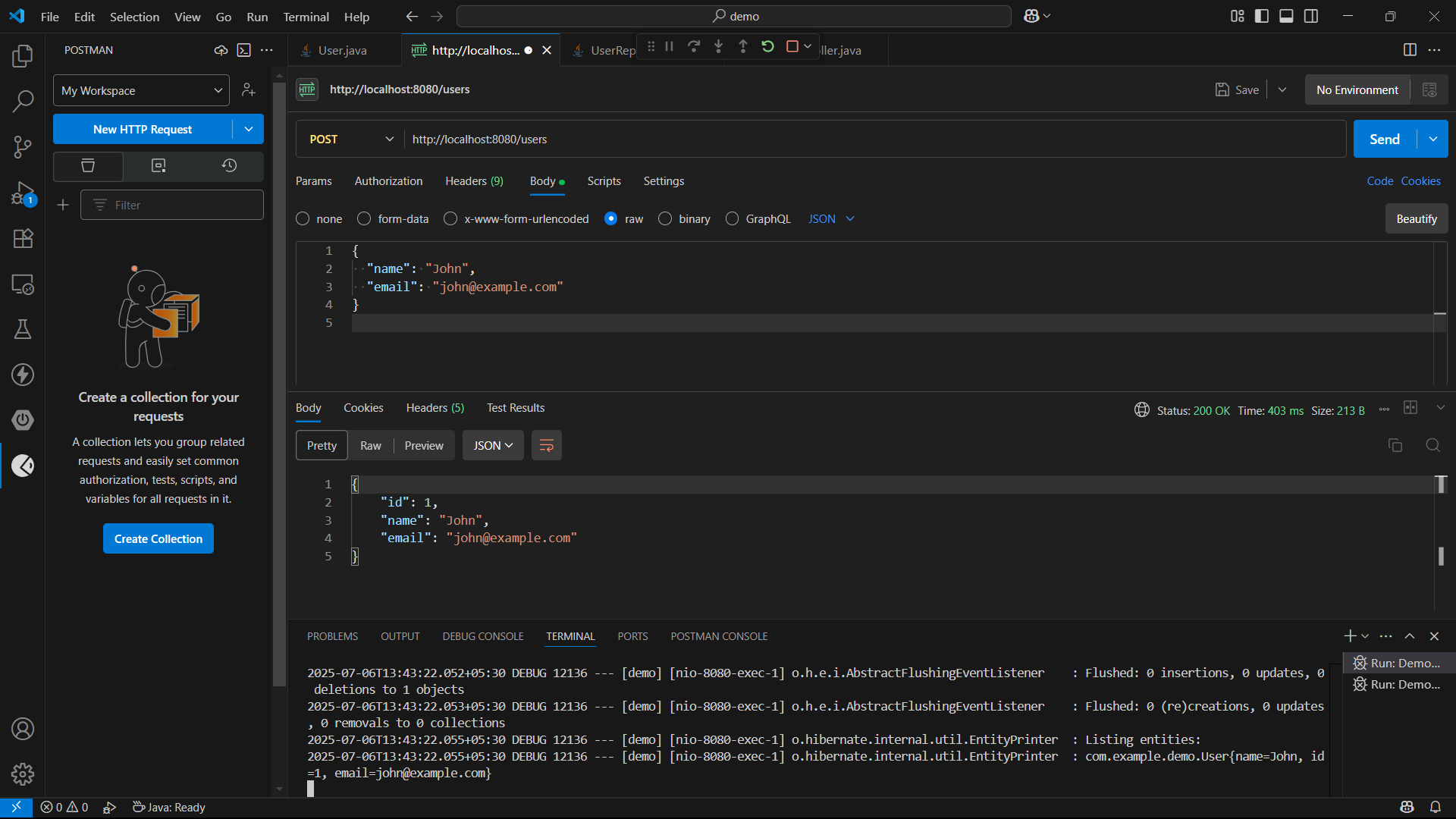
        return userRepository.save(user);

    }

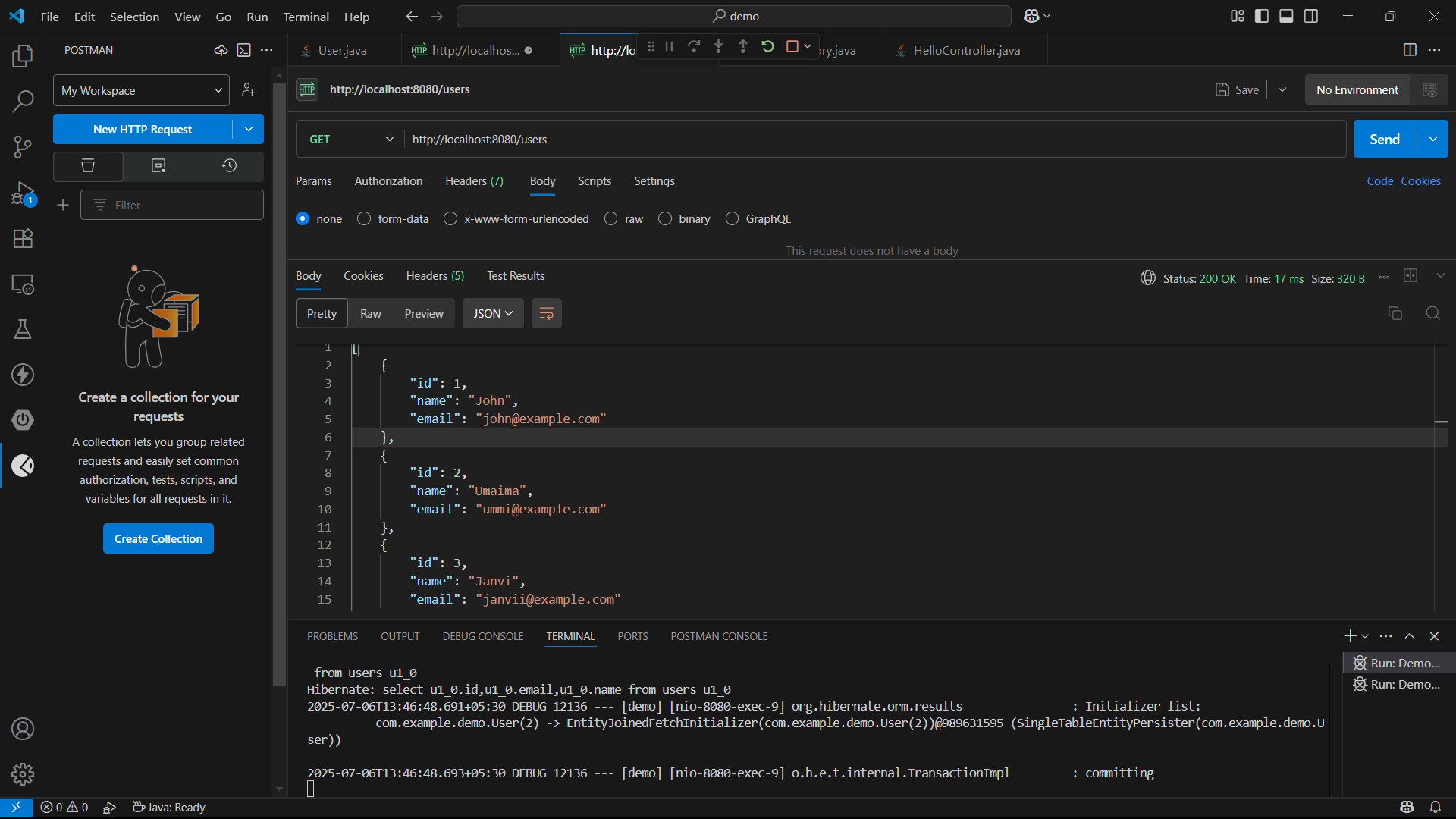
}

**Output:**

**POST:**



**GET:**



**Check Users in H2 Console:**

