

Java EndSem Practical Exam

Name-NAITIK KUKREJA RollNO-202401100400120

<https://github.com/Naitik4141/java-praCTICAL>

UserController.java

1. Model

package com.example.usermanagement.entity;

```
import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;
import jakarta.persistence.Table;
import lombok.Data;
import lombok.NoArgsConstructor;
import lombok.AllArgsConstructor;
```

```
@Entity
@Table(name = "users")
```

```
@Data
@NoArgsConstructor
@AllArgsConstructor
public class User {
```

```
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
private Long id;  
private String username;  
private String email;  
}
```

```
2.View(FrontEnd)  
<!DOCTYPE html>  
<html lang="en">  
<head>  
<meta charset="UTF-8">  
<meta name="viewport" content="width=device-width, initial-  
scale=1.0">  
<title>User Management System</title>  
<link rel="stylesheet" href="style.css">  
<link  
href="https://fonts.googleapis.com/css2?family=Inter:wght@400;  
500;600;700&display=swap" rel="stylesheet">  
</head>  
<body>  
<div class="container">  
<header>  
<h1>User Management</h1>  
<p>Manage your users efficiently and intuitively.</p>  
</header>  
  
<main>  
<section class="card form-section">
```

```
<h2>Add New User</h2>


---


<form id="userForm">
  <input type="hidden" id="userId">
  <div class="form-group">
    <label for="username">Username</label>
    <input type="text" id="username" placeholder="e.g. johndoe" required>
  </div>
  <div class="form-group">
    <label for="email">Email Address</label>
    <input type="email" id="email" placeholder="e.g. john@example.com" required>
  </div>
  <div class="form-actions">
    <button type="submit" id="submitBtn" class="btn primary">
      <span class="icon">+</span> Add User
    </button>
    <button type="button" id="cancelBtn" class="btn secondary hidden">Cancel Edit</button>
  </div>
</form>
</section>

<section class="card list-section">
  <div class="list-header">
    <h2>Current Users</h2>
    <span class="badge" id="userCount">0 Users</span>
```

```
</div>
<div class="table-container">
  <table id="userTable">
    <thead>
      <tr>
        <th>ID</th>
        <th>Username</th>
        <th>Email</th>
        <th>Actions</th>
      </tr>
    </thead>
    <tbody>
      <!-- User rows will be injected here -->
    </tbody>
  </table>
  <div id="emptyState" class="empty-state">
    <p>No users found. Start by adding one above.</p>
  </div>
</div>
</section>
</main>
</div>

<div id="toast" class="toast hidden">Action Successful</div>
<script src="app.js"></script>
</body>
</html>
3.Controller(API)
```

```
package com.example.usermanagement.controller;

import com.example.usermanagement.entity.User;
import com.example.usermanagement.service.UserService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.ResponseEntity;
import org.springframework.web.bind.annotation.*;

import java.util.List;

@RestController
@RequestMapping("/users")
public class UserController {

        private final UserService userService;

        @Autowired
        public UserController(UserService userService) {
                this.userService = userService;
        }

        @PostMapping
        public User createUser(@RequestBody User user) {
                return userService.createUser(user);
        }

        @GetMapping
        public List<User> getAllUsers() {
```

```
    return userService.getAllUsers();  
}  
  
 @PutMapping("/{id}")  
 public ResponseEntity<User> updateUser(@PathVariable Long  
 id, @RequestBody User userDetails) {  
     try {  
         User updatedUser = userService.updateUser(id,  
 userDetails);  
         return ResponseEntity.ok(updatedUser);  
     } catch (RuntimeException e) {  
         return ResponseEntity.notFound().build();  
     }  
 }  
  
 @DeleteMapping("/{id}")  
 public ResponseEntity<Void> deleteUser(@PathVariable Long  
 id) {  
     userService.deleteUser(id);  
     return ResponseEntity.noContent().build();  
 }  
}
```

The screenshot shows a web application titled "User Management" with the subtitle "Manage your users efficiently and intuitively." Below this, there are two main sections: "Add New User" and "Current Users".

Add New User: This section contains fields for "Username" (with placeholder "e.g. john Doe") and "Email Address" (with placeholder "e.g. john@example.com"). A blue button labeled "+ Add User" is located below the input fields.

Current Users: This section displays a table header with columns "ID", "USERNAME", "EMAIL", and "ACTIONS". Below the header, a message says "No users found. Start by adding one above." and a small circular button indicates "0 Users".

The screenshot shows an IDE interface with the following components:

- Project View:** Shows the project structure under "java-practical [usermanagement]". The "src/main/java/com/example/usermanagement/UserManagementApplication.java" file is selected and displayed in the editor.
- Editor:** The code for `UserManagementApplication.java` is shown:package com.example.usermanagement;
import ...
@SpringBootApplication
public class UserManagementApplication {
 public static void main(String[] args) { SpringApplication.run(UserManagementApplication.class, args); }
}
- Run Tab:** The "UserManagementApplication" run configuration is selected.
- Terminal:** The terminal window shows the Spring Boot startup logs:2025-12-30T12:01:21.350+05:30 INFO 12796 --- [usermanagement] [main] c.e.u.UserManagementApplication : Starting UserManagementApplication using Java
2025-12-30T12:01:21.352+05:30 INFO 12796 --- [usermanagement] [main] c.e.u.UserManagementApplication : No active profile set, falling back to 1 def
2025-12-30T12:01:21.702+05:30 INFO 12796 --- [usermanagement] [main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories i
2025-12-30T12:01:21.725+05:30 INFO 12796 --- [usermanagement] [main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in
2025-12-30T12:01:21.954+05:30 INFO 12796 --- [usermanagement] [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Suggested plugin Lombok available.
2025-12-30T12:01:21.961+05:30 INFO 12796 --- [usermanagement] [main] o.apache.catalina.core.StandardService : Configure plugins... Don't suggest again
2025-12-30T12:01:21.981+05:30 INFO 12796 --- [usermanagement] [main] o.a.c.c.[Tomcat].[localhost].[] :
2025-12-30T12:01:21.987+05:30 INFO 12796 --- [usermanagement] [main] w.s.c.WebApplicationContext : Root webApplicationContext: initialization c

JP java-praCTICAL Version control

Project

```
1 package com.example.usermanagement.service;
2 > import ...
3
4 @Service 3 usages
5 public class UserService {
6
7     private final UserRepository userRepository; 7 usages
8
9     @Autowired no usages
10    public UserService(UserRepository userRepository) { this.userRepository = userRepository; }
11
12    public User createUser(User user) { return userRepository.save(user); }
13
14    public List<User> getAllUsers() { return userRepository.findAll(); }
15
16    public Optional<User> getUserById(Long id) { return userRepository.findById(id); }
17
18    public User updateUser(Long id, User userDetails) { 1 usage
19        return userRepository.findById(id).map(User user -> {
20            user.setUsername(userDetails.getUsername());
21            user.setEmail(userDetails.getEmail());
22            return userRepository.save(user);
23        }).orElseThrow(() -> new RuntimeException("User not found with id " + id));
24    }
25
26    public void deleteUser(Long id) { userRepository.deleteById(id); }
27}
```

Run UserManagementApplication

Current File

Suggested plugin Lombok available.

Configure plugins... Don't suggest again

12:14 LF UTF-8 4 spaces

JP java-praCTICAL Version control

Project

```
1 package com.example.usermanagement.repository;
2 > import ...
3
4 @Repository 3 usages
5 public interface UserRepository extends JpaRepository<User, Long> {
6}
```

Run UserManagementApplication

Current File

Suggested plugin Lombok available.

Configure plugins... Don't suggest again

8:18 LF UTF-8 4 spaces

Project

UserManagementApplication.java UserRepository.java User.java UserService.java

```
1 package com.example.usermanagement.entity;
2
3 > import ...
4
5 @Entity 16 usages
6 @Table(name = "users")
7 @Data
8 @NoArgsConstructor
9 @AllArgsConstructor
10 public class User {
11
12     @Id no usages
13     @GeneratedValue(strategy = GenerationType.IDENTITY)
14     private Long id;
15
16     private String username; no usages
17     private String email; no usages
18 }
```

Run UserManagementApplication

Suggested plugin Lombok available

Configure plugins... Don't suggest

17:14 LF UTF-8 4 spaces

Project

UserManagementApplication.java UserRepository.java User.java UserController.java UserService.java

```
1 package com.example.usermanagement.controller;
2
3 > import ...
4
5 @RestController no usages
6 @RequestMapping("/users")
7 public class UserController {
8
9     private final UserService userService; 5 usages
10
11     @Autowired no usages
12     public UserController(UserService userService) { this.userService = userService; }
13
14     @PostMapping no usages
15     public User createUser(@RequestBody User user) { return userService.createUser(user); }
16
17     @GetMapping no usages
18     public List<User> getAllUsers() { return userService.getAllUsers(); }
19
20     @PutMapping("/{id}") no usages
21     public ResponseEntity<User> updateUser(@PathVariable Long id, @RequestBody User userDetails) {
22         try {
23             User updatedUser = userService.updateUser(id, userDetails);
24             return ResponseEntity.ok(updatedUser);
25         } catch (RuntimeException e) {
26             return ResponseEntity.notFound().build();
27         }
28     }
29
30     @DeleteMapping("/{id}") no usages
31     public ResponseEntity<Void> deleteUser(@PathVariable Long id) {
32         userService.deleteUser(id);
33         return ResponseEntity.noContent().build();
34     }
35 }
```

33:14 LF UTF-8 4 spaces

JP java-praCTICAL Version control

Project

- java-praCTICAL [usermanagement] ~/Desktop
- .idea
- ...
 - .gitignore
 - compiler.xml
 - encodings.xml
 - jarRepositories.xml
 - misc.xml
 - workspace.xml
- src
 - main
 - java
 - resources
 - static
 - app.js
 - index.html
 - style.css
 - target
 - pom.xml
- External Libraries
- Scratches and Consoles

Current File

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>User Management System</title>
    <link rel="stylesheet" href="style.css">
    <link href="https://fonts.googleapis.com/css?family=Inter:wght@400;500;700&display=swap" rel="stylesheet">
</head>
<body>
    <div class="container">
        <header>
            <h1>User Management</h1>
            <p>Manage your users efficiently and intuitively.</p>
        </header>

        <main>
            <section class="card form-section">
                <h2>Add New User</h2>
                <form id="userForm">
                    <input type="hidden" id="userId">
                    <div class="form-group">
                        <label for="username">Username</label>
                        <input type="text" id="username" placeholder="e.g. johndoe" required>
                    </div>
                    <div class="form-group">
                        <label for="email">Email Address</label>
                        <input type="email" id="email" placeholder="e.g. john@example.com" required>
                    </div>
                    <div class="form-actions">
                        <button type="submit" id="submitBtn" class="btn primary">
                            <span class="icon"><></span> Add User
                        </button>
                        <button type="button" id="cancelBtn" class="btn secondary hidden">Cancel Edit</button>
                    </div>
                </form>
            </section>
        </main>
    </div>
</body>
```

1:1 LF ISO-8859-1 4 spaces

JP java-praCTICAL Version control

Project

- java-praCTICAL [usermanagement] ~/Desktop
- .idea
- ...
 - .gitignore
 - compiler.xml
 - encodings.xml
 - jarRepositories.xml
 - misc.xml
 - workspace.xml
- src
 - main
 - java
 - resources
 - static
 - app.js
 - index.html
 - style.css
 - target
 - pom.xml
- External Libraries
- Scratches and Consoles

Current File

application.properties

Spring Boot configuration files are supported by IntelliJ IDEA Ultimate Try IntelliJ IDEA Ultimate Dismiss

```
spring.application.name=usermanagement
spring.datasource.url=jdbc:h2:mem:testdb
spring.datasource.driverClassName=org.h2.Driver
spring.datasource.username=sa
spring.datasource.password=password
spring.jpa.database-platform=org.hibernate.dialect.H2Dialect
spring.h2.console.enabled=true
server.port = 8081
```

8:19 LF ISO-8859-1 4 spaces