PE\_HSF302\_SPR25\_250315\_Note

SPRING 2025  
Subject: HSF302  
Duration: 90 minutes

**INSTRUCTIONS**

**Please read the instructions carefully before doing the questions.**

* You are **NOT allowed** to use any other materials. You are **NOT allowed** to use any device to share data with others.
* You must use IDE as **IntelliJ IDEA 2023 or later, MSSQL Server 2019 or** **later database** for your development tools.

**IMPORTANT – Before you start doing your solution, MUST do the following steps:**

* To do your program, you must use **Spring Boot + (JSP or Thymeleaf) + Spring Data JPA + MSSQL**, apply N-Layer architecture. Ensure that each layer interacts through an interface using Dependency Injection (DI)

*Note: you* ***are not*** *allowed to connect direct to database from Service* or UI layer*, every database connection must be used with Repository using JPA. The database connection string must get from application.properties file.* ***In the case your program connects directly to database from Service/Pages or hardcode connection string, you will get 0 point.***

1. ***If there are syntax errors or compilation errors in your PE program, you will not pass the PE requirements, the point will be 0.***
2. Create a Spring Boot project named **PE\_HSF302\_250315\_StudentCode**, with the package name prefixed as *hsf302.sp25.studentcode*, and include at least four packages: *pojo, repository, service* and *controller*.
3. Create your MS SQL database named **SonyStore2025DB** using a code-first approach with Spring Data JPA, and ensure that it includes the data provided in the **PE\_HSF302\_SPR25\_250315\_Note** file
4. Set the default user interface for your project to the Login Page with the URL: <http://localhost:8080/login>
5. ***Your work will be considered invalid (0 point) if your code inserts stuff that is unrelated to the test.***

**­­­­­**

**REFERENCES *(this session just for reference, student can use the other approach to do Practical Exam and student must change to run your program)***

1. ***Working with DB connection string from application.properties***

spring.application.name=DemoLab01  
spring.mvc.view.prefix=/WEB-INF/views/  
spring.mvc.view.suffix=.jsp  
  
spring.datasource.url=jdbc:sqlserver://localhost:1433;databaseName= SonyStore2025DB;encrypt=true;trustServerCertificate=true;  
spring.datasource.username=sa  
spring.datasource.password=1234  
spring.datasource.driverClassName=com.microsoft.sqlserver.jdbc.SQLServerDriver  
spring.jpa.show-sql=true  
spring.jpa.hibernate.ddl-auto=create

**2. Working with the *pom.xml***

<dependencies>  
 <dependency>  
 <groupId>com.microsoft.sqlserver</groupId>  
 <artifactId>mssql-jdbc</artifactId>  
 <scope>runtime</scope>  
 </dependency>  
  
 <dependency>  
 <groupId>org.hibernate</groupId>  
 <artifactId>hibernate-core</artifactId>  
 <version>6.5.2.Final</version>  
 </dependency>  
  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-web</artifactId>  
 <scope>provided</scope>  
 </dependency>  
  
 <dependency>  
 <groupId>org.apache.tomcat.embed</groupId>  
 <artifactId>tomcat-embed-jasper</artifactId>  
 <scope>provided</scope>  
 </dependency>  
  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-test</artifactId>  
 <scope>provided</scope>  
 </dependency>  
  
 <dependency>  
 <groupId>org.glassfish.web</groupId>  
 <artifactId>jakarta.servlet.jsp.jstl</artifactId>  
 <version>3.0.1</version>  
 </dependency>  
  
 <dependency>  
 <groupId>jakarta.servlet.jsp.jstl</groupId>  
 <artifactId>jakarta.servlet.jsp.jstl-api</artifactId>  
 <version>3.0.2</version>  
 <scope>provided</scope>  
 </dependency>  
  
 <dependency>  
 <groupId>jakarta.persistence</groupId>  
 <artifactId>jakarta.persistence-api</artifactId>  
 <version>3.1.0</version>  
 <scope>provided</scope>  
 </dependency>

<dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-data-jpa</artifactId>  
 <version>3.4.2</version>  
</dependency>

<dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-validation</artifactId>  
</dependency>

</dependencies>

1. **Config on SpringBootApplication project**

@SpringBootApplication  
@ComponentScan({"hsf302.se.fu.vn.controllers","hsf302.se.fu.vn.services"})  
@EnableJpaRepositories(basePackages = "hsf302.se.fu.vn.repositories")  
@EntityScan(basePackages = "hsf302.se.fu.vn")  
public class DemoLab01Application {  
 public static void main(String[] args) {  
 SpringApplication.run(DemoLab01Application.class, args);  
 }  
}

**5.Tables**

|  |  |
| --- | --- |
| **SonyAccounts** | |
| AccountID | int |
| FullName | nvarchar(50) |
| Email | varchar(30) |
| Phone | nvarchar(13) |
| Status | nvarchar(10) |
| RoleID | int |

|  |  |
| --- | --- |
| **SonyCategories** | |
| CateID | int |
| CateName | nvarchar(50) |
| CateURL | nvarchar(100) |
| Status | nvarchar(10) |

|  |  |
| --- | --- |
| **SonyProducts** | |
| ProductID | int |
| ProductName | nvarchar(50) |
| Price | int |
| Stock | int |
| CreatedAt | datetime |
| UpdatedAt | datetime |
| CateID | int |

**6.Data**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| AccountID | FullName | Email | Phone | Status | RoleID |
| 4 | Admin | admin@sony.com | 0905111111 | active | 1 |
| 5 | Staff | staff@sony.com | 0905222222 | active | 2 |
| 6 | Manager | manager@sony.com | 0905333333 | active | 3 |
| 7 | Customer | customer@sony.com | 0905444444 | active | 4 |
| 8 | Customer1 | customer1@sony.com | 0902444444 | inactive | 4 |

|  |  |  |  |
| --- | --- | --- | --- |
| CateID | CateName | CateURL | Status |
| 1 | HeadPhone | /headphon | active |
| 2 | Cameras | /cameras | active |
| 3 | TVs | /tivi | active |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ProductID | ProductName | Price | Stock | CreatedAt | UpdatedAt | CateID |
| 1 | Alpha 1 II - Full-frame Mirrorless | 6000 | 3 | 2025-03-03 | 2025-03-03 | 2 |
| 2 | Alpha 7C II – Full-frame | 2000 | 5 | 2025-04-04 | 2025-04-04 | 2 |
| 3 | BRAVIA 8 OLED 4K HDR TV | 2500 | 10 | 2025-01-01 | 2025-03-03 | 3 |
| 7 | LinkBuds Fit Truly Wireless Noise Canceling | 180 | 15 | 2025-03-03 | 2025-03-15 | 1 |

**7. JSTL**

<%@ taglib prefix="c" uri="jakarta.tags.core" %>  
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%>  
<%@ page isELIgnored="false"%>

**8. Thymeleaf**

<html xmlns:th="http://www.thymeleaf.org">

<tr th:each="user : ${users}">  
 <td th:text="${user.username}"></td>  
</tr>

<div class="error-message" th:if="${error}" th:text="${error}">

<div class="form-group">  
 <label for="username">Username:</label>  
 <input type="text" name="username" id="username" th:field="\*{username}">  
 <span class="error-message" th:if="${#fields.hasErrors('username')}"  
 th:errors="\*{username}"></span>  
</div>