Web API Design with Spring Boot Week 3 Coding Assignment

Points possible: 70

Category	Criteria	% of Grade
Functionality	Does the code work?	25
Organization	Is the code clean and organized? Proper use of white space, syntax, and consistency are utilized. Names and comments are concise and clear.	25
Creativity	Student solved the problems presented in the assignment using creativity and out of the box thinking.	25
Completeness	All requirements of the assignment are complete.	25

Instructions: In Eclipse, or an IDE of your choice, write the code that accomplishes the objectives listed below. Ensure that the code compiles and runs as directed. Take screenshots of the code and of the running program (make sure to get screenshots of all required functionality) and paste them in this document where instructed below. Create a new repository on GitHub for this week's assignments and push this document, with your Java project code, to the repository. Add the URL for this week's repository to this document where instructed and submit this document to your instructor when complete.

Here's a friendly tip: as you watch the videos, code along with the videos. This will help you with the homework. When a screenshot is required, look for the icon: 🖥️ You will keep adding to this project throughout this part of the course. When it comes time for the final project, use this project as a starter.

Project Resources: **https://github.com/promineotech/Spring-Boot-Course-Student-Resources**

Coding Steps:

1) In the application you've been building add a DAO layer:
    a) Add the package, com.promineotech.jeep.dao.
    b) In the new package, create an interface named `JeepSalesDao`.
    c) In the same package, create a class named `DefaultJeepSalesDao` that implements `JeepSalesDao`.
    d) Add a method in the DAO interface and implementation that returns a list of Jeep models (class `Jeep`) and takes the model and trim parameters. Here is the method signature:

```
List<Jeep> fetchJeeps(JeepModel model, String trim);
```

2) In the Jeep sales service implementation class, inject the DAO interface as an instance variable. The instance variable should be `private` and should be named `jeepSalesDao`. Call the DAO method from the service method and store the returned value in a local variable named `jeeps`. Return the value in the `jeeps` variable (we will add to this later).

3) In the DAO implementation class (`DefaultJeepSalesDao`):

   a) Add the class-level annotation: `@Service`.

   b) Add a log statement in `DefaultJeepSalesDao.fetchJeeps()` that logs the model and trim level. Run the integration test. Produce a screenshot showing the DAO implementation class and the log line in the IDE's console.

   c) \\\

   d) In `DefaultJeepSalesDao`, inject an instance variable of type `NamedParameterJdbcTemplate`.

   e) Write SQL to return a list of Jeep models based on the parameters: model and trim. Be sure to utilize the SQL Injection prevention mechanism of the NamedParameterJdbcTemplate using `:model_id` and `:trim_level` in the query.

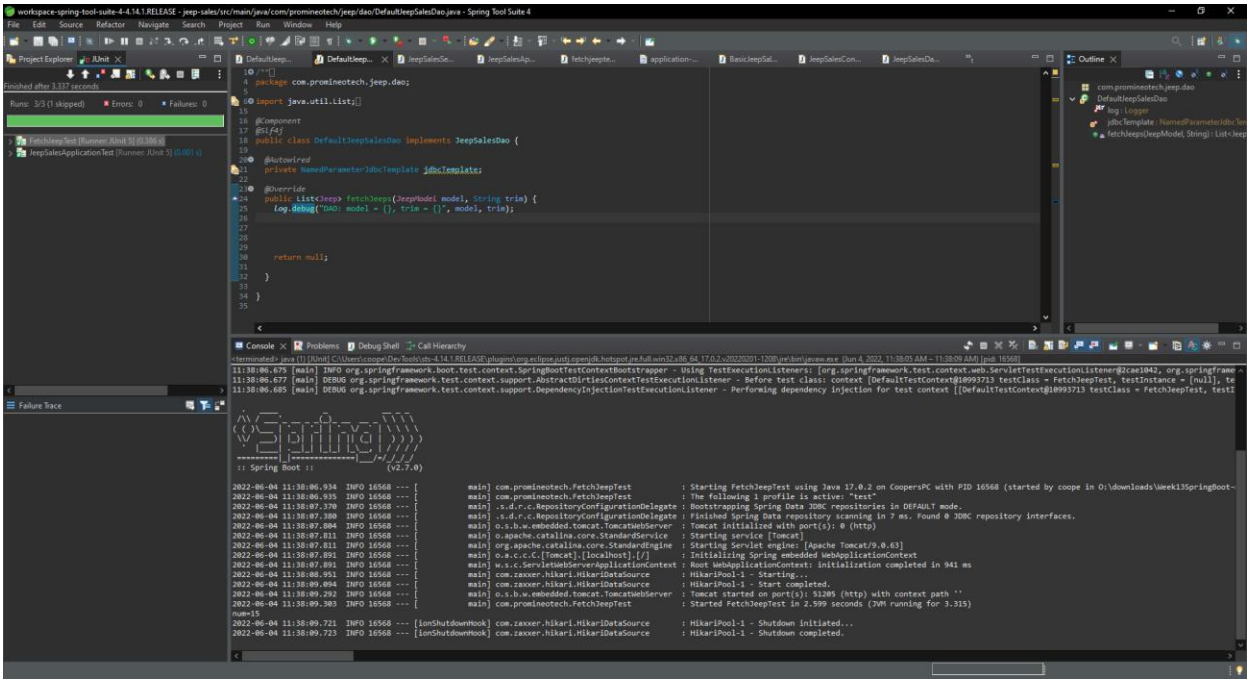
   f) Add the parameters to a parameter map as shown in the video. Don't forget to convert the JeepModel enum value to a String (i.e., `params.put("model_id", model.toString());`)

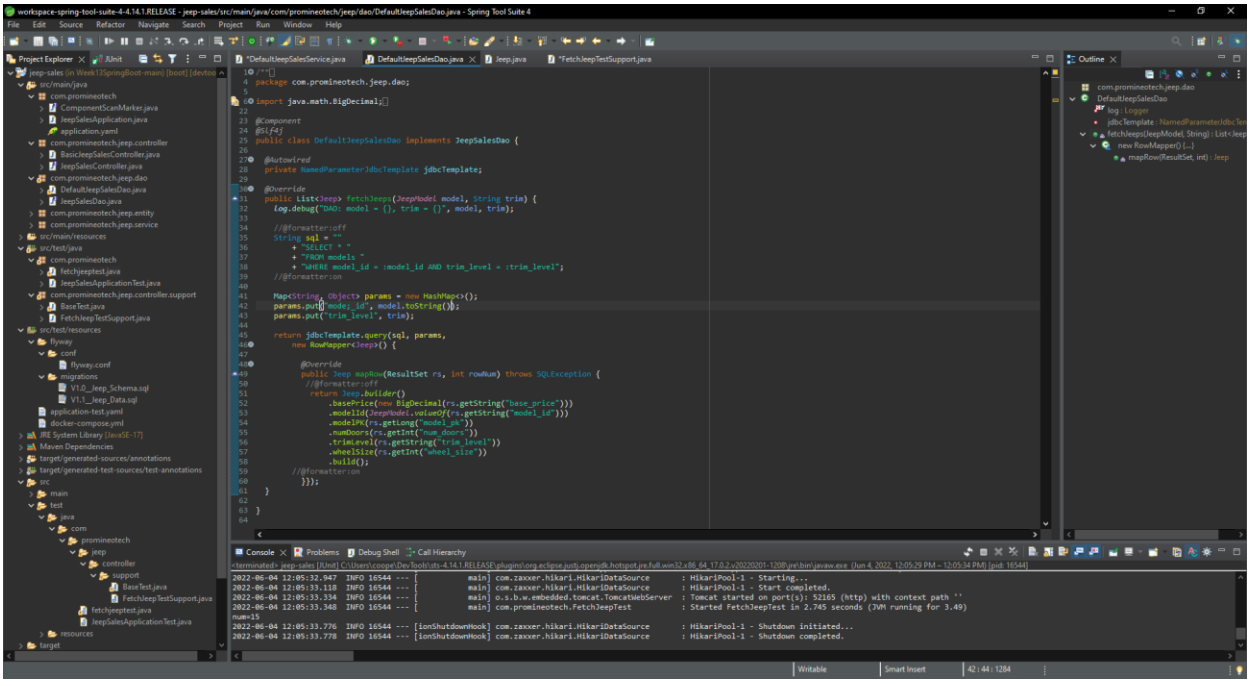
   g) Call the `query` method on the `NamedParameterJdbcTemplate` instance variable to return a list of Jeep model objects. Use a RowMapper to map each row of the result set. Remember to convert `modelId` to a `JeepModel`. See the video for details. Produce a screenshot to show the complete method in the implementation class.

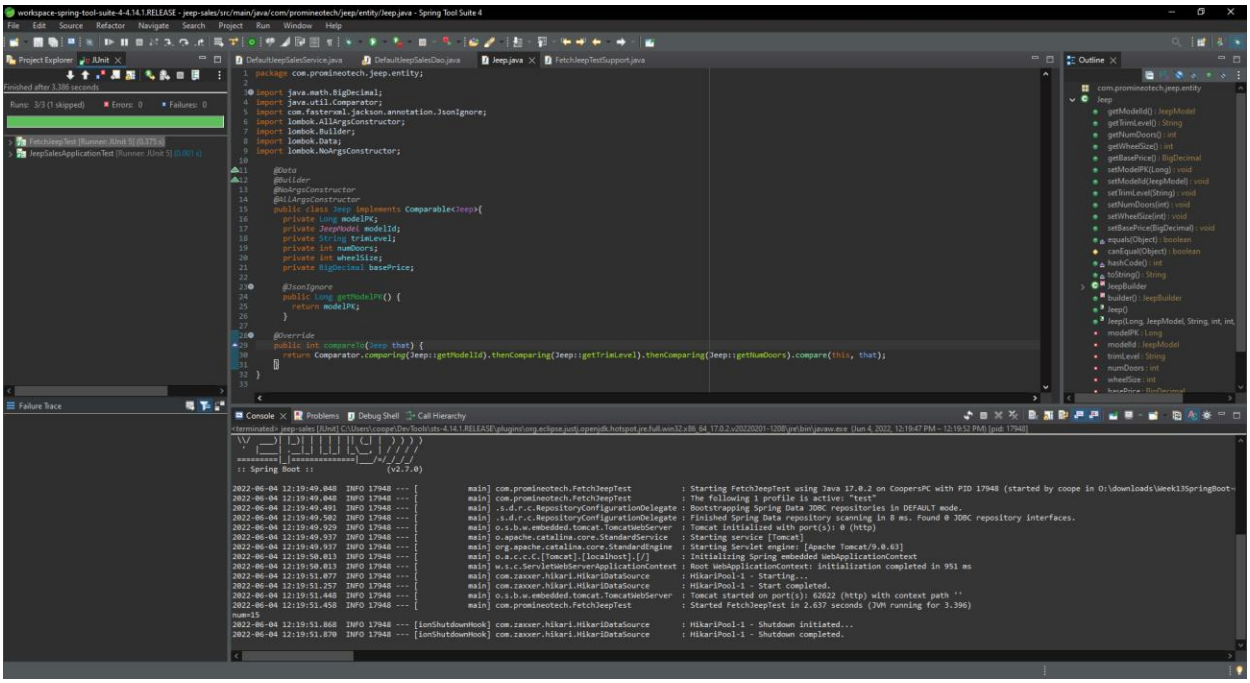
4) Add a getter in the `Jeep` class for `modelPK`. Add the `@JsonIgnore` annotation to the getter to exclude the `modelPK` value from the returned object.

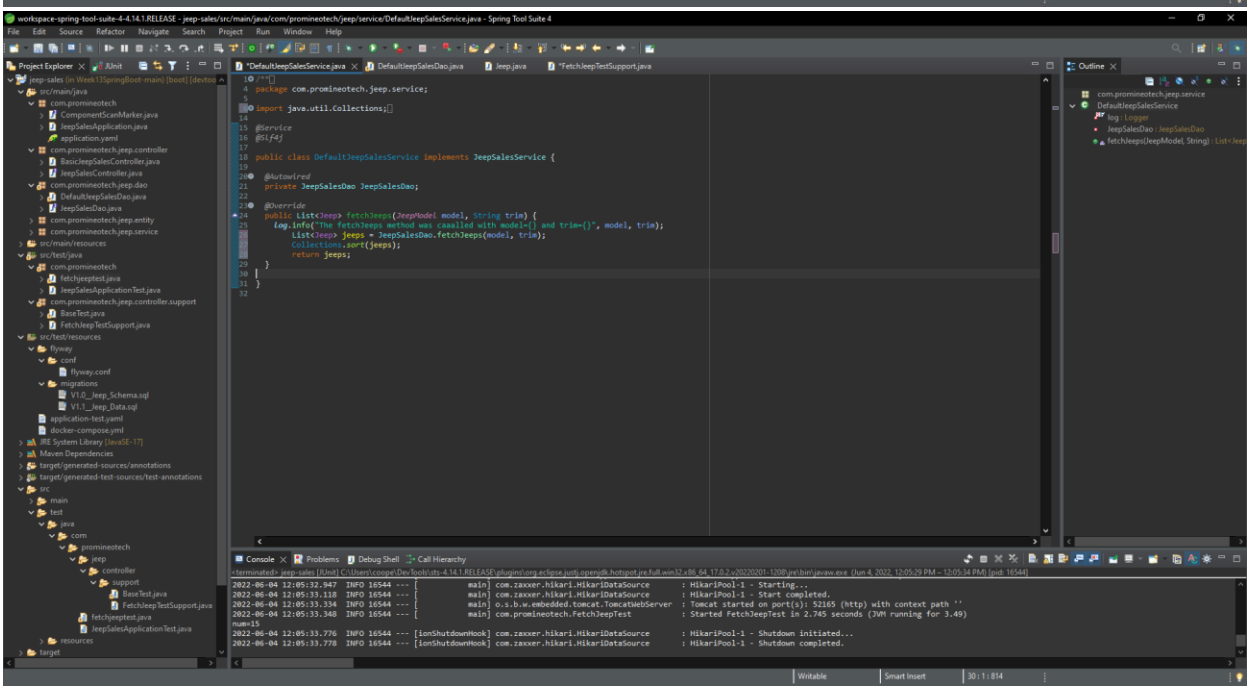
5) Run the test to produce a green status bar. Produce a screenshot showing the test and the green status bar.

Screenshots of Code:






workspace-spring-tool-suite-4-4.14.1.RELEASE - jeep-sales/src/main/java/com/promineotech/jeep/entity/Jeep.java - Spring Tool Suite 4
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer JUnit
Finished after 3.386 seconds
Runs: 3/3 (1 skipped) Errors: 0 Failures: 0
FetchJeepTest [Runner: JUnit 5] (0.375 s)
JeepSalesApplicationTest [Runner: JUnit 5] (0.001 s)
Failure Trace
DefaultJeepSalesService.java DefaultJeepSalesDao.java Jeep.java FetchJeepTestSupport.java
package com.promineotech.jeep.entity;
import java.math.BigDecimal;
import java.util.Comparator;
import com.fasterxml.jackson.annotation.JsonIgnore;
import lombok.AllArgsConstructor;
import lombok.Builder;
import lombok.Data;
import lombok.NoArgsConstructor;
@Data
@Builder
@NoArgsConstructor
@AllArgsConstructor
public class Jeep implements Comparable<Jeep>{
private Long modelPK;
private JeepModel modelId;
private String trimLevel;
private int numDoors;
private int wheelSize;
private BigDecimal basePrice;
@JsonIgnore
public Long getModelPK() {
return modelPK;
}
@Override
public int compareTo(Jeep that) {
return Comparator.comparing(Jeep::getModelId).thenComparing(Jeep::getTrimLevel).thenComparing(Jeep::getNumDoors).compare(this, that);
}
}
Outline
com.promineotech.jeep.entity
Jeep
getModelId() : JeepModel
getTrimLevel() : String
getNumDoors() : int
getWheelSize() : int
getBasePrice() : BigDecimal
setModelPK(Long) : void
setModelId(JeepModel) : void
setTrimLevel(String) : void
setNumDoors(int) : void
setWheelSize(int) : void
setBasePrice(BigDecimal) : void
equals(Object) : boolean
canEqual(Object) : boolean
hashCode() : int
toString() : String
JeepBuilder
builder() : JeepBuilder
Jeep()
Jeep(Long, JeepModel, String, int, int, BigDecimal)
modelPK : Long
modelId : JeepModel
trimLevel : String
numDoors : int
wheelSize : int
basePrice : BigDecimal
Console Problems Debug Shell Call Hierarchy
<terminated> jeep-sales [JUnit] C:\Users\coope\Dev Tools\sts-4.14.1.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v20220201-1208\jre\bin\javaw.exe (Jun 4, 2022, 12:19:47 PM – 12:19:52 PM) [pid: 17948]
:: Spring Boot :: (v2.7.0)
2022-06-04 12:19:49.048 INFO 17948 --- [main] com.promineotech.FetchJeepTest : Starting FetchJeepTest using Java 17.0.2 on CoopersPC with PID 17948 (started by coope in O:\downloads\Week13SpringBoot-
2022-06-04 12:19:49.048 INFO 17948 --- [main] com.promineotech.FetchJeepTest : The following 1 profile is active: "test"
2022-06-04 12:19:49.491 INFO 17948 --- [main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JDBC repositories in DEFAULT mode.
2022-06-04 12:19:49.502 INFO 17948 --- [main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 8 ms. Found 0 JDBC repository interfaces.
2022-06-04 12:19:49.929 INFO 17948 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 0 (http)
2022-06-04 12:19:49.937 INFO 17948 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2022-06-04 12:19:49.937 INFO 17948 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.63]
2022-06-04 12:19:50.013 INFO 17948 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2022-06-04 12:19:50.013 INFO 17948 --- [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 951 ms
2022-06-04 12:19:51.077 INFO 17948 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting...
2022-06-04 12:19:51.257 INFO 17948 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.
2022-06-04 12:19:51.448 INFO 17948 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 62622 (http) with context path ''
2022-06-04 12:19:51.458 INFO 17948 --- [main] com.promineotech.FetchJeepTest : Started FetchJeepTest in 2.637 seconds (JVM running for 3.396)
num=15
2022-06-04 12:19:51.868 INFO 17948 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown initiated...
2022-06-04 12:19:51.870 INFO 17948 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown completed.


workspace-spring-tool-suite-4-4.14.1.RELEASE - jeep-sales/src/main/java/com/promineotech/jeep/service/DefaultJeepSalesService.java - Spring Tool Suite 4
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer JUnit
jeep-sales [in Week13SpringBoot-main] [boot] [devtools]
src/main/java
com.promineotech
ComponentScanMarker.java
JeepSalesApplication.java
application.yaml
com.promineotech.jeep.controller
BasicJeepSalesController.java
JeepSalesController.java
com.promineotech.jeep.dao
DefaultJeepSalesDao.java
JeepSalesDao.java
com.promineotech.jeep.entity
com.promineotech.jeep.service
src/main/resources
src/test/java
com.promineotech
fetchjeeptest.java
JeepSalesApplicationTest.java
com.promineotech.jeep.controller.support
BaseTest.java
FetchJeepTestSupport.java
src/test/resources
flyway
conf
flyway.conf
migrations
V1.0__Jeep_Schema.sql
V1.1__Jeep_Data.sql
application-test.yaml
docker-compose.yml
JRE System Library [JavaSE-17]
Maven Dependencies
target/generated-sources/annotations
target/generated-test-sources/test-annotations
src
main
test
java
com
promineotech
jeep
controller
support
BaseTest.java
FetchJeepTestSupport.java
fetchjeeptest.java
JeepSalesApplicationTest.java
resources
target
*DefaultJeepSalesService.java DefaultJeepSalesDao.java Jeep.java *FetchJeepTestSupport.java
package com.promineotech.jeep.service;
import java.util.Collections;
@Service
@Slf4j
public class DefaultJeepSalesService implements JeepSalesService {
@Autowired
private JeepSalesDao JeepSalesDao;
@Override
public List<Jeep> fetchJeeps(JeepModel model, String trim) {
log.info("The fetchJeeps method was caaalled with model={} and trim={}", model, trim);
List<Jeep> jeeps = JeepSalesDao.fetchJeeps(model, trim);
Collections.sort(jeeps);
return jeeps;
}
}
Outline
com.promineotech.jeep.service
DefaultJeepSalesService
log : Logger
JeepSalesDao : JeepSalesDao
fetchJeeps(JeepModel, String) : List<Jeep>
Console Problems Debug Shell Call Hierarchy
<terminated> jeep-sales [JUnit] C:\Users\coope\Dev Tools\sts-4.14.1.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v20220201-1208\jre\bin\javaw.exe (Jun 4, 2022, 12:05:29 PM – 12:05:34 PM) [pid: 16544]
2022-06-04 12:05:32.947 INFO 16544 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting...
2022-06-04 12:05:33.118 INFO 16544 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.
2022-06-04 12:05:33.334 INFO 16544 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 52165 (http) with context path ''
2022-06-04 12:05:33.348 INFO 16544 --- [main] com.promineotech.FetchJeepTest : Started FetchJeepTest in 2.745 seconds (JVM running for 3.49)
num=15
2022-06-04 12:05:33.776 INFO 16544 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown initiated...
2022-06-04 12:05:33.778 INFO 16544 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown completed.
Writable
Smart Insert
30 : 1 : 814