**Hands on 1**

**Introduction to HQL and JPQL** 

HQL and JPQL are object-oriented query languages that let you query databases using Java classes instead of table names. HQL is specific to Hibernate, while JPQL is part of the JPA standard.

**What is HQL?**

Hibernate Query Language used to query database using entity names, not table names.

Syntax is similar to SQL but object-oriented.

Example:

Query q = session.createQuery("FROM Country WHERE name = 'India'");

**What is JPQL?**

Java Persistence Query Language defined by JPA specification.

Works with entities and their attributes like HQL.

Example:

Query q = entityManager.createQuery("SELECT c FROM Country c WHERE c.name = 'India'");

**Difference between HQL and JPQL**

1. HQL is specific to Hibernate, while JPQL is part of the JPA specification.
2. HQL supports INSERT operations; JPQL does not support INSERT.
3. All JPQL queries are valid HQL, but not all HQL queries are valid JPQL.

**Advantages**

More readable than native SQL (uses entities).

Works across multiple databases (DB-independent).

Supports automatic joins and lazy loading.

**Disadvantages**

Slower than native SQL in complex queries.

Less control over DB-specific features.

JPQL doesn't support INSERT operations.

**Hands on 2**

**Get all permanent employees using HQL**   
  
Using HQL get all permanent employees. When retrieving the employee details it should also retrieve respective department and skill list as well.  
  
**EmployeeRepository.java**

package com.cognizant.orm\_learn.repository;

import java.util.List;

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

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

import org.springframework.stereotype.Repository;

import com.cognizant.orm\_learn.model.Employee;

*@Repository*

public interface EmployeeRepository extends JpaRepository<Employee, Integer> {

*@Query*("SELECT e FROM Employee e left join fetch e.department d left join fetch e.skillList WHERE e.permanent = true")

List<Employee> getAllPermanentEmployees();

}

EmployeeService.java

package com.cognizant.orm\_learn.service;

import java.util.List;

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

import org.springframework.stereotype.Service;

import com.cognizant.orm\_learn.model.Employee;

import com.cognizant.orm\_learn.repository.EmployeeRepository;

*@Service*

public class EmployeeService {

*@Autowired*

private EmployeeRepository employeeRepository;

public List<Employee> getAllPermanentEmployees() {

return employeeRepository.getAllPermanentEmployees();

}

}

OrmLearnApplication.java

package com.cognizant.orm\_learn;

import java.util.List;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

import org.springframework.boot.CommandLineRunner;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import com.cognizant.orm\_learn.model.Employee;

import com.cognizant.orm\_learn.service.EmployeeService;

*@SpringBootApplication*

public class OrmLearnApplication implements CommandLineRunner {

*@Autowired*

private EmployeeService employeeService;

private static final Logger ***LOGGER*** = LoggerFactory.*getLogger*(OrmLearnApplication.class);

public static void main(String[] args) {

SpringApplication.*run*(OrmLearnApplication.class, args);

}

*@Override*

public void run(String... args) throws Exception {

testGetAllPermanentEmployees();

}

public void testGetAllPermanentEmployees() {

***LOGGER***.info("Start");

List<Employee> employees = employeeService.getAllPermanentEmployees();

***LOGGER***.debug("Permanent Employees:{}", employees);

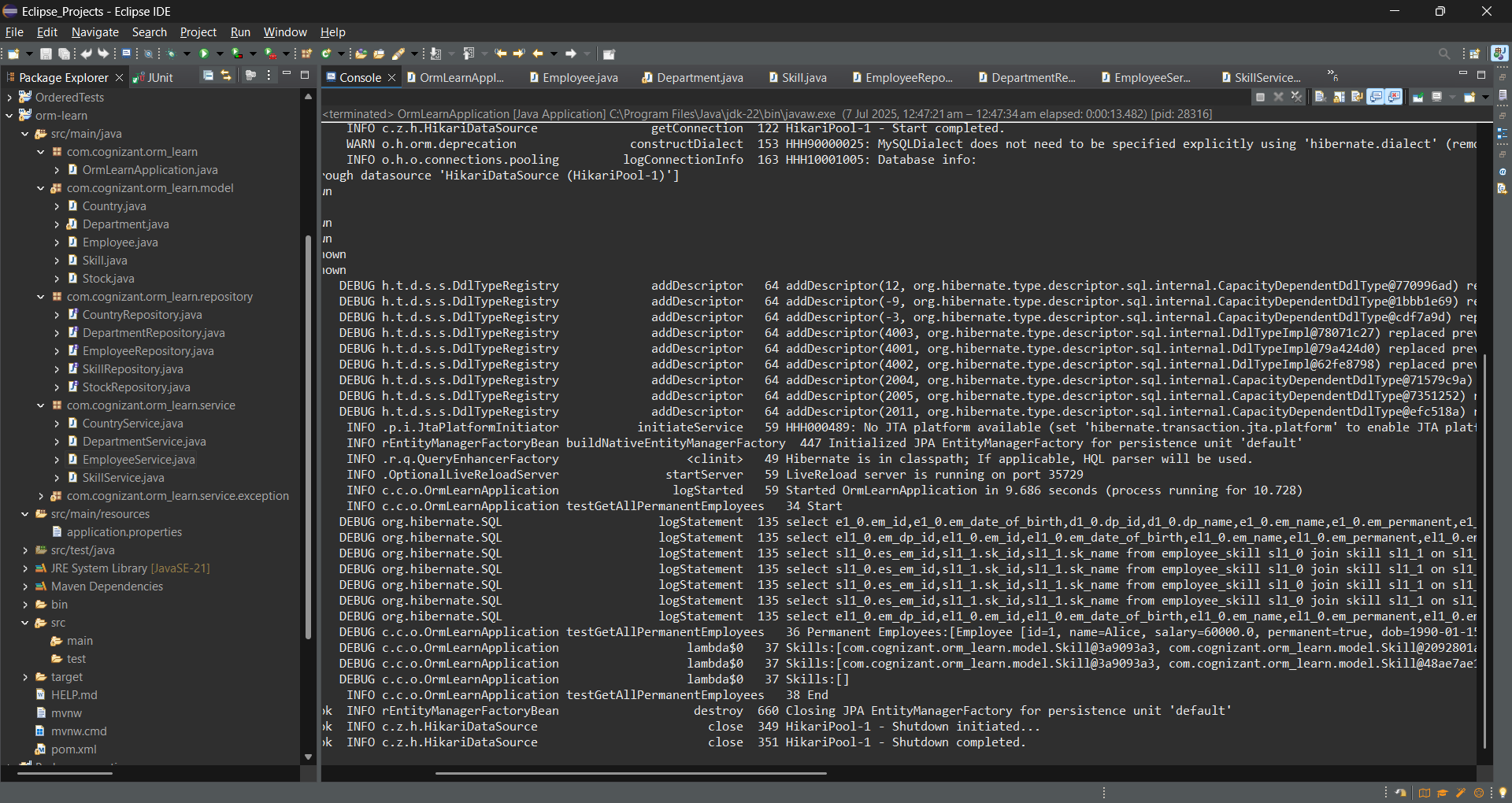
employees.forEach(e -> ***LOGGER***.debug("Skills:{}", e.getSkillList()));

***LOGGER***.info("End");

}

}

Output:



**Hands on 3**

**Fetch quiz attempt details using HQL**   
Attempt.java

package com.cognizant.orm\_learn.model;

import java.sql.Date;

import java.util.List;

import java.util.Set;

import jakarta.persistence.CascadeType;

import jakarta.persistence.Column;

import jakarta.persistence.Entity;

import jakarta.persistence.Id;

import jakarta.persistence.JoinColumn;

import jakarta.persistence.ManyToOne;

import jakarta.persistence.OneToMany;

import jakarta.persistence.Table;

*@Entity*

*@Table*(name = "attempt")

public class Attempt {

public Attempt() {

super();

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public Date getDate() {

return date;

}

public void setDate(Date date) {

this.date = date;

}

public int getUserId() {

return userId;

}

public void setUserId(int userId) {

this.userId = userId;

}

public double getScore() {

return score;

}

public void setScore(double score) {

this.score = score;

}

public User getUser() {

return user;

}

public void setUser(User user) {

this.user = user;

}

public Set<AttemptQuestion> getAttemptQuestions() {

return attemptQuestions;

}

public void setAttemptQuestions(Set<AttemptQuestion> attemptQuestions) {

this.attemptQuestions = attemptQuestions;

}

public Attempt(int id, Date date, int userId, double score, User user, Set<AttemptQuestion> attemptQuestions) {

super();

this.id = id;

this.date = date;

this.userId = userId;

this.score = score;

this.user = user;

this.attemptQuestions = attemptQuestions;

}

*@Id*

*@Column*(name = "at\_id")

private int id;

*@Column*(name = "at\_date")

private Date date;

*@Column*(name = "at\_us\_id")

private int userId;

*@Column*(name = "at\_score")

private double score;

*@ManyToOne*

*@JoinColumn*(name = "at\_us\_id", insertable = false, updatable = false)

private User user;

*@OneToMany*(mappedBy = "attempt", cascade = *CascadeType*.***ALL***)

private Set<AttemptQuestion> attemptQuestions;

}

AttemptOption.java

package com.cognizant.orm\_learn.model;

import jakarta.persistence.Column;

import jakarta.persistence.Entity;

import jakarta.persistence.Id;

import jakarta.persistence.JoinColumn;

import jakarta.persistence.ManyToOne;

import jakarta.persistence.Table;

*@Entity*

*@Table*(name = "attempt\_option")

public class AttemptOption {

public AttemptOption() {

super();

}

public AttemptOption(int id, int optionId, int attemptQuestionId, boolean selected, Option option,

AttemptQuestion attemptQuestion) {

super();

this.id = id;

this.optionId = optionId;

this.attemptQuestionId = attemptQuestionId;

this.selected = selected;

this.option = option;

this.attemptQuestion = attemptQuestion;

}

*@Id*

*@Column*(name = "ao\_id")

private int id;

*@Column*(name = "ao\_op\_id")

private int optionId;

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public int getOptionId() {

return optionId;

}

public void setOptionId(int optionId) {

this.optionId = optionId;

}

public int getAttemptQuestionId() {

return attemptQuestionId;

}

public void setAttemptQuestionId(int attemptQuestionId) {

this.attemptQuestionId = attemptQuestionId;

}

public boolean isSelected() {

return selected;

}

public void setSelected(boolean selected) {

this.selected = selected;

}

public Option getOption() {

return option;

}

public void setOption(Option option) {

this.option = option;

}

public AttemptQuestion getAttemptQuestion() {

return attemptQuestion;

}

public void setAttemptQuestion(AttemptQuestion attemptQuestion) {

this.attemptQuestion = attemptQuestion;

}

*@Column*(name = "ao\_aq\_id")

private int attemptQuestionId;

*@Column*(name = "ao\_selected")

private boolean selected;

*@ManyToOne*

*@JoinColumn*(name = "ao\_op\_id", insertable = false, updatable = false)

private Option option;

*@ManyToOne*

*@JoinColumn*(name = "ao\_aq\_id", insertable = false, updatable = false)

private AttemptQuestion attemptQuestion;

}

AttemptQuestion.java

package com.cognizant.orm\_learn.model;

import java.util.List;

import java.util.Set;

import jakarta.persistence.CascadeType;

import jakarta.persistence.Column;

import jakarta.persistence.Entity;

import jakarta.persistence.Id;

import jakarta.persistence.JoinColumn;

import jakarta.persistence.ManyToOne;

import jakarta.persistence.OneToMany;

import jakarta.persistence.Table;

*@Entity*

*@Table*(name = "attempt\_question")

public class AttemptQuestion {

public AttemptQuestion() {

super();

}

public AttemptQuestion(int id, int attemptId, int questionId, Attempt attempt, Question question,

Set<AttemptOption> attemptOptions) {

super();

this.id = id;

this.attemptId = attemptId;

this.questionId = questionId;

this.attempt = attempt;

this.question = question;

this.attemptOptions = attemptOptions;

}

public Set<AttemptOption> getAttemptOptions() {

return attemptOptions;

}

public void setAttemptOptions(Set<AttemptOption> attemptOptions) {

this.attemptOptions = attemptOptions;

}

*@Id*

*@Column*(name = "aq\_id")

private int id;

*@Column*(name = "aq\_at\_id")

private int attemptId;

*@Column*(name = "aq\_qt\_id")

private int questionId;

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public int getAttemptId() {

return attemptId;

}

public void setAttemptId(int attemptId) {

this.attemptId = attemptId;

}

public int getQuestionId() {

return questionId;

}

public void setQuestionId(int questionId) {

this.questionId = questionId;

}

public Attempt getAttempt() {

return attempt;

}

public void setAttempt(Attempt attempt) {

this.attempt = attempt;

}

public Question getQuestion() {

return question;

}

public void setQuestion(Question question) {

this.question = question;

}

*@OneToMany*(mappedBy = "attemptQuestion", cascade = *CascadeType*.***ALL***)

private Set<AttemptOption> attemptOptions;

*@ManyToOne*

*@JoinColumn*(name = "aq\_at\_id", insertable = false, updatable = false)

private Attempt attempt;

*@ManyToOne*

*@JoinColumn*(name = "aq\_qt\_id", insertable = false, updatable = false)

private Question question;

}

Option.java

package com.cognizant.orm\_learn.model;

import jakarta.persistence.Column;

import jakarta.persistence.Entity;

import jakarta.persistence.Id;

import jakarta.persistence.JoinColumn;

import jakarta.persistence.ManyToOne;

import jakarta.persistence.Table;

*@Entity*

*@Table*(name = "options")

public class Option {

public Option() {

super();

}

public Option(int id, int questionId, double score, String text, Question question) {

super();

this.id = id;

this.questionId = questionId;

this.score = score;

this.text = text;

this.question = question;

}

*@Id*

*@Column*(name = "op\_id")

private int id;

*@Column*(name = "op\_qt\_id")

private int questionId;

*@Column*(name = "op\_score")

private double score;

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public int getQuestionId() {

return questionId;

}

public void setQuestionId(int questionId) {

this.questionId = questionId;

}

public double getScore() {

return score;

}

public void setScore(double score) {

this.score = score;

}

public String getText() {

return text;

}

public void setText(String text) {

this.text = text;

}

public Question getQuestion() {

return question;

}

public void setQuestion(Question question) {

this.question = question;

}

*@Column*(name = "op\_text")

private String text;

*@ManyToOne*

*@JoinColumn*(name = "op\_qt\_id", insertable = false, updatable = false)

private Question question;

}

Question.java

package com.cognizant.orm\_learn.model;

import java.util.List;

import jakarta.persistence.CascadeType;

import jakarta.persistence.Column;

import jakarta.persistence.Entity;

import jakarta.persistence.Id;

import jakarta.persistence.OneToMany;

import jakarta.persistence.Table;

*@Entity*

*@Table*(name = "question")

public class Question {

public Question() {

super();

}

public Question(int id, String text, List<Option> options) {

super();

this.id = id;

this.text = text;

this.options = options;

}

*@Id*

*@Column*(name = "qt\_id")

private int id;

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getText() {

return text;

}

public void setText(String text) {

this.text = text;

}

public List<Option> getOptions() {

return options;

}

public void setOptions(List<Option> options) {

this.options = options;

}

*@Column*(name = "qt\_text")

private String text;

*@OneToMany*(mappedBy = "question", cascade = *CascadeType*.***ALL***)

private List<Option> options;

}

User.java

package com.cognizant.orm\_learn.model;

import java.util.List;

import jakarta.persistence.CascadeType;

import jakarta.persistence.Column;

import jakarta.persistence.Entity;

import jakarta.persistence.Id;

import jakarta.persistence.OneToMany;

import jakarta.persistence.Table;

*@Entity*

*@Table*(name = "user")

public class User {

public User() {

super();

}

public User(int id, String name, String email, List<Attempt> attempts) {

super();

this.id = id;

this.name = name;

this.email = email;

this.attempts = attempts;

}

*@Id*

*@Column*(name = "us\_id")

private int id;

*@Column*(name = "us\_name")

private String name;

*@Column*(name = "us\_email")

private String email;

public int getId() {

return id;

}

public void setId(int 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;

}

public List<Attempt> getAttempts() {

return attempts;

}

public void setAttempts(List<Attempt> attempts) {

this.attempts = attempts;

}

*@OneToMany*(mappedBy = "user", cascade = *CascadeType*.***ALL***)

private List<Attempt> attempts;

}

AttemptRepository.java

package com.cognizant.orm\_learn.repository;

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

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

import org.springframework.data.repository.query.Param;

import org.springframework.stereotype.Repository;

import com.cognizant.orm\_learn.model.Attempt;

*@Repository*

public interface AttemptRepository extends JpaRepository<Attempt, Integer> {

*@Query*("SELECT a FROM Attempt a " +

"JOIN FETCH a.user u " +

"JOIN FETCH a.attemptQuestions aq " +

"JOIN FETCH aq.question q " +

"JOIN FETCH aq.attemptOptions ao " +

"JOIN FETCH ao.option o " +

"WHERE u.id = :userId AND a.id = :attemptId")

Attempt getAttempt(*@Param*("userId") int userId, *@Param*("attemptId") int attemptId);

}

AttemptService.java

package com.cognizant.orm\_learn.service;

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

import org.springframework.stereotype.Service;

import com.cognizant.orm\_learn.model.Attempt;

import com.cognizant.orm\_learn.repository.AttemptRepository;

*@Service*

public class AttemptService {

*@Autowired*

private AttemptRepository attemptRepository;

public Attempt getAttempt(int userId, int attemptId) {

return attemptRepository.getAttempt(userId, attemptId);

}

}

OrmLearnApplication.java

package com.cognizant.orm\_learn;

import java.util.Set;

import com.cognizant.orm\_learn.model.\*;

import com.cognizant.orm\_learn.service.AttemptService;

import com.cognizant.orm\_learn.service.EmployeeService;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

import org.springframework.boot.CommandLineRunner;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class OrmLearnApplication implements CommandLineRunner {

*@Autowired*

private EmployeeService employeeService;

*@Autowired*

private AttemptService attemptService;

private static final Logger ***LOGGER*** = LoggerFactory.*getLogger*(OrmLearnApplication.class);

public static void main(String[] args) {

SpringApplication.*run*(OrmLearnApplication.class, args);

}

*@Override*

public void run(String... args) throws Exception {

testGetAllPermanentEmployees();

testGetAttemptDetails();

}

public void testGetAllPermanentEmployees() {

***LOGGER***.info("Start - testGetAllPermanentEmployees");

var employees = employeeService.getAllPermanentEmployees();

***LOGGER***.debug("Permanent Employees: {}", employees);

employees.forEach(e -> ***LOGGER***.debug("Skills: {}", e.getSkillList()));

***LOGGER***.info("End - testGetAllPermanentEmployees");

}

public void testGetAttemptDetails() {

***LOGGER***.info("Start - testGetAttemptDetails");

int userId = 1;

int attemptId = 1;

Attempt attempt = attemptService.getAttempt(userId, attemptId);

if (attempt == null) {

***LOGGER***.error("Attempt not found for userId={} and attemptId={}", userId, attemptId);

return;

}

***LOGGER***.debug("User: {}", attempt.getUser().getName());

***LOGGER***.debug("Date: {}", attempt.getDate());

Set<AttemptQuestion> questions = attempt.getAttemptQuestions();

for (AttemptQuestion aq : questions) {

String questionText = aq.getQuestion().getText();

***LOGGER***.info(questionText);

Set<AttemptOption> attemptOptions = aq.getAttemptOptions();

for (AttemptOption ao : attemptOptions) {

Option option = ao.getOption();

String optionText = option.getText();

double score = option.getScore();

boolean selected = ao.isSelected();

***LOGGER***.info(" {}) {} {} {}", option.getId(), optionText, score, selected);

}

***LOGGER***.info("");

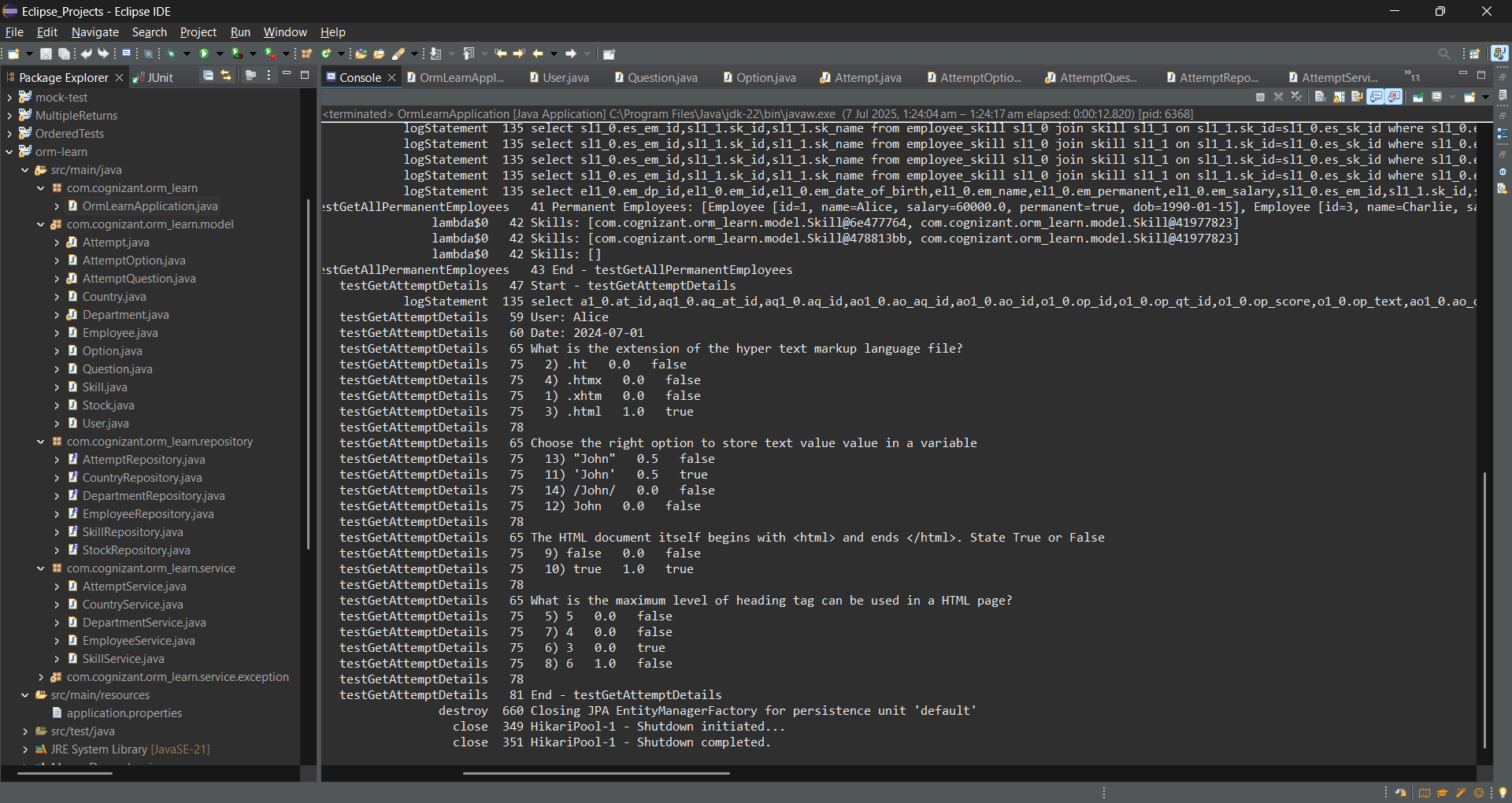
}

***LOGGER***.info("End - testGetAttemptDetails");

}

}

Output:



**Hands on 4**

**Get average salary using HQL** 

EmployeeRepository.java

package com.cognizant.orm\_learn.repository;

import java.util.List;

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

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

import org.springframework.data.repository.query.Param;

import org.springframework.stereotype.Repository;

import com.cognizant.orm\_learn.model.Employee;

*@Repository*

public interface EmployeeRepository extends JpaRepository<Employee, Integer> {

*@Query*("SELECT e FROM Employee e WHERE e.permanent = true")

List<Employee> getAllPermanentEmployees();

*@Query*("SELECT AVG(e.salary) FROM Employee e WHERE e.department.id = :id")

double getAverageSalary(*@Param*("id") int id);

}

EmployeeService.java

package com.cognizant.orm\_learn.service;

import java.util.List;

import com.cognizant.orm\_learn.model.Employee;

public interface EmployeeService {

List<Employee> getAllPermanentEmployees();

double getAverageSalary(int departmentId);

}

EmployeeServiceImpl.java

package com.cognizant.orm\_learn.service;

import java.util.List;

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

import org.springframework.stereotype.Service;

import com.cognizant.orm\_learn.model.Employee;

import com.cognizant.orm\_learn.repository.EmployeeRepository;

*@Service*

public class EmployeeServiceImpl implements EmployeeService {

*@Autowired*

private EmployeeRepository employeeRepository;

*@Override*

public List<Employee> getAllPermanentEmployees() {

return employeeRepository.getAllPermanentEmployees();

}

*@Override*

public double getAverageSalary(int departmentId) {

return employeeRepository.getAverageSalary(departmentId);

}

}

OrmLearnApplication.java

package com.cognizant.orm\_learn;

import java.util.List;

import com.cognizant.orm\_learn.model.Attempt;

import com.cognizant.orm\_learn.model.AttemptOption;

import com.cognizant.orm\_learn.model.AttemptQuestion;

import com.cognizant.orm\_learn.model.Employee;

import com.cognizant.orm\_learn.service.AttemptService;

import com.cognizant.orm\_learn.service.EmployeeService;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

import org.springframework.boot.CommandLineRunner;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class OrmLearnApplication implements CommandLineRunner {

private static final Logger ***LOGGER*** = LoggerFactory.*getLogger*(OrmLearnApplication.class);

*@Autowired*

private EmployeeService employeeService;

*@Autowired*

private AttemptService attemptService;

public static void main(String[] args) {

SpringApplication.*run*(OrmLearnApplication.class, args);

}

*@Override*

public void run(String... args) throws Exception {

testGetAllPermanentEmployees();

testGetAttemptDetails();

testGetAverageSalary();

}

public void testGetAllPermanentEmployees() {

***LOGGER***.info("Start - testGetAllPermanentEmployees");

List<Employee> employees = employeeService.getAllPermanentEmployees();

***LOGGER***.debug("Permanent Employees: {}", employees);

employees.forEach(e -> ***LOGGER***.debug("Skills: {}", e.getSkillList()));

***LOGGER***.info("End - testGetAllPermanentEmployees");

}

public void testGetAttemptDetails() {

***LOGGER***.info("Start - testGetAttemptDetails");

Attempt attempt = attemptService.getAttempt(1, 1); // Replace with valid userId and attemptId

***LOGGER***.debug("User: {}", attempt.getUser().getName());

***LOGGER***.debug("Date: {}", attempt.getDate());

for (AttemptQuestion aq : attempt.getAttemptQuestions()) {

***LOGGER***.debug("Q: {}", aq.getQuestion().getText());

for (AttemptOption ao : aq.getAttemptOptions()) {

***LOGGER***.debug(" Option: {} Score: {} Selected: {}",

ao.getOption().getText(),

ao.getOption().getScore(),

ao.isSelected());

}

}

***LOGGER***.info("End - testGetAttemptDetails");

}

public void testGetAverageSalary() {

***LOGGER***.info("Start - testGetAverageSalary");

int departmentId = 1; // replace with valid department ID in your DB

double avgSalary = employeeService.getAverageSalary(departmentId);

***LOGGER***.debug("Average Salary for Department {}: {}", departmentId, avgSalary);

***LOGGER***.info("End - testGetAverageSalary");

}

}

Output:

