

Demonstrate writing Hibernate Query Language and Native Query

1. Hibernate Query Language (HQL)

A. HQL to Get All Permanent Employees (with Department & Skill)

Repository

```
@Query("SELECT e FROM Employee e LEFT JOIN FETCH e.department d LEFT JOIN FETCH e.skillList WHERE e.permanent = 1")
```

```
List<Employee> getAllPermanentEmployees();
```

Service

```
public List<Employee> getAllPermanentEmployees() {  
    return employeeRepository.getAllPermanentEmployees();  
}
```

Application Test Method

```
private static 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");  
}
```

B. HQL with Aggregate Function – Average Salary by Department

Repository

```
@Query("SELECT AVG(e.salary) FROM Employee e WHERE e.department.id = :id")  
double getAverageSalary(@Param("id") int deptId);
```

Service

```
public double getAverageSalary(int deptId) {  
    return employeeRepository.getAverageSalary(deptId);  
}
```

Application Test Method

```
private static void testGetAverageSalary() {  
    LOGGER.info("Start");  
    double avgSalary = employeeService.getAverageSalary(2);  
    LOGGER.debug("Average Salary: {}", avgSalary);  
    LOGGER.info("End");  
}
```

2. Native SQL Query

Get All Employees – Native Query

Repository

```
@Query(value = "SELECT * FROM employee", nativeQuery = true)  
List<Employee> getAllEmployeesNative();
```

Service

```
public List<Employee> getAllEmployeesNative() {  
    return employeeRepository.getAllEmployeesNative();  
}
```

Application Test Method

```
private static void testGetAllEmployeesNative() {  
    LOGGER.info("Start");  
  
    List<Employee> employees = employeeService.getAllEmployeesNative();  
  
    LOGGER.debug("Employees (Native): {}", employees);  
  
    LOGGER.info("End");  
}
```

3. HQL with JOIN & FETCH Keywords

- * Use JOIN to link tables (objects).
- * Use FETCH to load the associated entities eagerly.

Example:

```
@Query("SELECT e FROM Employee e LEFT JOIN FETCH e.department d LEFT JOIN FETCH  
e.skillList WHERE e.permanent = 1")
```

SUPERSET ID: 6407636

KANMANI MURUGHAIYAN