Sorting with method queries

OrderBy

```java
@Repository
public interface EmployeeRepository extends JpaRepository<Employee,
Long> {
    List<Employee> findAllByOrderByNameAsc();
    List<Employee> findAllByOrderByNameDesc();
}
```

Sorting with the Sort class

Sort Parameter In Query Methods

```
@Repository
public interface EmployeeRepository extends JpaRepository<Employee,
Long> {
    List<Employee> findByDepartment(String department, Sort sort);
}
```

Using the Sort class

```
Sort sort = Sort.by(Sort.Direction.ASC, sortField);
Sort sort = Sort.by(Sort.Order.asc("name"), Sort.Order.desc("salary"));
```

Key Concepts of Pagination

- Page: A single chunk of data that contains a subset of the total dataset. It is an interface representing a page of data, including information about the total number of pages, total number of elements, and the current page's data.

- Pageable: An interface that provides pagination information such as page number, page size, and sorting options.

- PageRequest: A concrete implementation of Pageable that provides methods to create pagination and sorting information.

Using Pageable

```
public interface UserRepository extends JpaRepository<User, Long> {
    Page<User> findAll(Pageable pageable);
    Page<User> findByLastName(String lastName, Pageable pageable);
}
```

Creating Pageable instance:

```
Pageable pageable = PageRequest.of(pageNumber, size,
                                    Sort.by("lastName").ascending());
```