first example

employee.java

```
package com.cybage.model;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;


@Entity
public class Employee {
	@Id
	@GeneratedValue(strategy = GenerationType.AUTO)
	private int id;
	private String name;
	public Employee() {
		super();
	}
	public Employee( String name) {
		super();
		this.name = name;
	}
	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;
	}
	@Override
	public String toString() {
		return "Employee [id=" + id + ", name=" + name + "]";
	}
}


package com.cybage.repository;

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

import com.cybage.model.Employee;

public interface EmployeeRepository extends JpaRepository<Employee, Integer>{

}
```

application.properties

```
spring.jpa.generate-ddl=false
spring.jpa.hibernate.ddl-auto=none
spring.datasource.url=jdbc:mysql://localhost:3306/cyb
spring.datasource.username=root
spring.datasource.password=admin123
spring.jpa.properties.hibernate.dialect = org.hibernate.dialect.MySQL5Dialect
```

data.sql

```
insert into users values(101, 'dm101');
insert into users values(102, 'dm102');
insert into users values(103, 'dm103');
insert into users values(104, 'dm104');
```

schema.sql

```
DROP TABLE IF EXISTS users;
create table users(id int primary key, name varchar(20));
```

---

query from method name

```
package com.cybage.repository;

import java.util.List;

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

import com.cybage.model.User;

@Repository
public interface UserRepository extends JpaRepository <User, Long> {
	User findByEmailAddress(String emailAddress);

	List < User > findByLastname(String lastname);
}
```

```
package com.cybage;

import java.util.Date;
import java.util.List;

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.cybage.model.User;
import com.cybage.repository.UserRepository;

@SpringBootApplication
```

```
public class FirstExampleApplication implements CommandLineRunner {

	public static void main(String[] args) {
		SpringApplication.run(FirstExampleApplication.class, args);
	}

	@Autowired
   private UserRepository userRepository;
	@Override
	public void run(String...args) throws Exception {
		User user = new User();
		user.setActive(1);
		user.setAge(35);
		user.setEmailAddress("dm@gmail.com");
		user.setFirstname("dm");
		user.setLastname("jadhav");
		user.setStartDate(new Date());
		user = userRepository.save(user);

		System.out.println("-----------------------------------:: " + user.getId());

		System.out.println(" ---------------@NamedQuery ---------------------");
		System.out.println("--------------findByEmailAddress -----------------");

		User user2 = userRepository.findByEmailAddress("dm@gmail.com");
		System.out.println(user2.toString());

		System.out.println(" ---------------@NamedQuery ---------------------");
		System.out.println("--------------findByLastname -----------------");

		List < User > user3 = userRepository.findByLastname("jadhav");
		System.out.println(user3.get(0).toString());
	}

}
```

query from method name

```
package com.cybage.model;
import java.util.Date;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Table;

@Entity
@Table(name = "user")
public class User {
 private long id;
 private String firstname;
 private String lastname;
 private Date startDate;
 private int age;
```

```
 private int active;

 @Id
 @GeneratedValue(strategy = GenerationType.AUTO)
 public long getId() {
  return id;
 }

 public void setId(long id) {
  this.id = id;
 }

 public String getFirstname() {
  return firstname;
 }

 public void setFirstname(String firstname) {
  this.firstname = firstname;
 }

 public String getLastname() {
  return lastname;
 }

 public void setLastname(String lastname) {
  this.lastname = lastname;
 }

 public Date getStartDate() {
  return startDate;
 }

 public void setStartDate(Date startDate) {
  this.startDate = startDate;
 }

 public int getAge() {
  return age;
 }

 public void setAge(int age) {
  this.age = age;
 }

 public int getActive() {
  return active;
 }

 public void setActive(int active) {
  this.active = active;
 }
}
```

```
package com.cybage.repository;

import java.util.Collection;
import java.util.Date;
import java.util.List;
import java.util.Optional;

import org.hibernate.sql.Insert;
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;

import com.cybage.model.User;


@Repository
public interface UserRepository extends JpaRepository<User, Long> {

	Optional<User> findByLastnameAndFirstname(String lastname, String firstname );

	List<User> findByLastnameOrFirstname(String lastname, String firstname);

	List<User> findByStartDateBetween(Date date1, Date date2);

	List<User> findByAgeLessThan(int age);

	List<User> findByAgeLessThanEqual(int age);

	List<User> findByAgeGreaterThan(int age);

	List<User> findByAgeGreaterThanEqual(int age);

	List<User> findByStartDateAfter(Date date);

	List<User> findByStartDateBefore(Date date);

	List<User> findByAgeIsNull();

	List<User> findByFirstnameLike(String firstname);

	List<User> findByFirstnameNotLike(String firstname);

	Optional<User> findByFirstnameStartingWith(String firstname);

	List<User> findByFirstnameEndingWith(String firstname);

	List<User> findByFirstnameContaining(String firstname);

	Optional<User> findByAgeOrderByLastnameDesc(int age);

	List<User> findByLastnameNot(String lastname);

	List<User> findByAgeIn(Collection<Integer> ages);
```

```
	List<User> findByAgeNotIn(Collection<Integer> ages);

	List<User> findByActiveTrue();

	List<User> findByActiveFalse();

	List<User> findByFirstnameIgnoreCase(String firstname);
}
```

```
package com.cybage.controller;

import java.util.List;
import java.util.Optional;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

import com.cybage.model.User;
import com.cybage.repository.UserRepository;


@RestController
@RequestMapping("/user")
public class EmpCrudOperations {
	@Autowired
	UserRepository ur;
	@GetMapping
	public List<User> findAll() {
		return ur.findAll();
	}
	@GetMapping("/1")
	public Optional<User> findByLastnameAndFirstname(){
		return ur.findByLastnameAndFirstname("jadhav", "asha");
	}
	@GetMapping("/2")
	public List<User> findByLastnameOrFirstname(){
		return ur.findByLastnameOrFirstname("jadhav123", "asha");
	}
	@GetMapping("/3")
	public List<User> findByAgeLessThan(){
		return ur.findByAgeLessThan(37);
	}
}
```

named queries

```
package com.cybage.model;
import java.util.Date;
```

```
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.NamedQueries;
import javax.persistence.NamedQuery;
import javax.persistence.Table;

@Entity
@Table(name = "user")
@NamedQueries(
		value= {
				@NamedQuery(name="User.findByAge", query = "select u from User u where u.age <
?1"),
				@NamedQuery(name="User.findByActive", query = "select u from User u where
u.active < ?1")
		})
public class User {
	private long id;
	private String firstname;
	private String lastname;
	private Date startDate;
	private int age;
	private int active;

	@Id
	@GeneratedValue(strategy = GenerationType.AUTO)
	public long getId() {
		return id;
	}

	public void setId(long id) {
		this.id = id;
	}

	public String getFirstname() {
		return firstname;
	}

	public void setFirstname(String firstname) {
		this.firstname = firstname;
	}

	public String getLastname() {
		return lastname;
	}

	public void setLastname(String lastname) {
		this.lastname = lastname;
	}

	public Date getStartDate() {
		return startDate;
```

```
	}

	public void setStartDate(Date startDate) {
		this.startDate = startDate;
	}

	public int getAge() {
		return age;
	}

	public void setAge(int age) {
		this.age = age;
	}

	public int getActive() {
		return active;
	}

	public void setActive(int active) {
		this.active = active;
	}
}
```

```
package com.cybage.repository;

import java.util.Collection;
import java.util.Date;
import java.util.List;
import java.util.Optional;

import org.hibernate.sql.Insert;
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;

import com.cybage.model.User;


@Repository
public interface UserRepository extends JpaRepository<User, Long> {
	public List<User> findByAge(int age);
	public List<User> findByActive(int active);
}
```

---

named native queries

```
package com.cybage;

import java.util.Date;
import java.util.List;

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.cybage.model.User;
import com.cybage.repository.UserRepository;

@SpringBootApplication
public class FirstExampleApplication implements CommandLineRunner {

	public static void main(String[] args) {
		SpringApplication.run(FirstExampleApplication.class, args);
	}

	@Autowired
	private UserRepository userRepository;

	@Override
	public void run(String...args) throws Exception {
		User user = new User();
		user.setActive(1);
		user.setAge(28);
		user.setEmailAddress("dm@gmail.com");
		user.setFirstname("dm");
		user.setLastname("jadhav");
		user.setStartDate(new Date());
		user = userRepository.save(user);

		System.out.println("--------------findByEmailAddress ----------------");

		User user2 = userRepository.findByEmailAddress("dm@gmail.com");
		System.out.println(user2.toString());

		System.out.println(" ----------------@NamedNativeQueries ----------------------");
		System.out.println("--------------findByLastname ----------------");

		List < User > user3 = userRepository.findByLastname("jadhav");
		System.out.println(user3.get(0).toString());
	}
}


package com.cybage.model;
import java.util.Date;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.NamedNativeQueries;
import javax.persistence.NamedNativeQuery;
import javax.persistence.NamedQueries;
import javax.persistence.NamedQuery;
import javax.persistence.Table;
```

```
@Entity
@Table(name = "user")
@NamedNativeQuery(name = "User.findByEmailAddress", query = "select * from user  where email_address = ?1",
resultClass = User.class)
@NamedNativeQueries(value = {
  @NamedNativeQuery(name = "User.findByLastname", query = "select * from user  where lastname = ?1", resultClass =
User.class) })
public class User {
   private long id;
   private String firstname;
   private String lastname;
   private Date startDate;
   private String emailAddress;
   private int age;
   private int active;

   @Id
   @GeneratedValue(strategy = GenerationType.AUTO)
   public long getId() {
      return id;
   }

   public void setId(long id) {
      this.id = id;
   }

   public String getFirstname() {
      return firstname;
   }

   public void setFirstname(String firstname) {
      this.firstname = firstname;
   }

   public String getLastname() {
      return lastname;
   }

   public void setLastname(String lastname) {
      this.lastname = lastname;
   }

   public Date getStartDate() {
      return startDate;
   }

   public void setStartDate(Date startDate) {
      this.startDate = startDate;
   }

   public int getAge() {
      return age;
   }
```

```
   public void setAge(int age) {
      this.age = age;
   }

   public int getActive() {
      return active;
   }

   public void setActive(int active) {
      this.active = active;
   }

   public String getEmailAddress() {
      return emailAddress;
   }

   public void setEmailAddress(String emailAddress) {
      this.emailAddress = emailAddress;
   }

   @Override
   public String toString() {
      return "User [id=" + id + ", firstname=" + firstname + ", lastname=" + lastname + ", startDate=" + startDate +
         ", emailAddress=" + emailAddress + ", age=" + age + ", active=" + active + "]";
   }
}


package com.cybage.repository;
import java.util.List;

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

import com.cybage.model.User;

@Repository
public interface UserRepository extends JpaRepository < User, Long > {

          User findByEmailAddress(String emailAddress);
          List <User> findByLastname(String lastname);
}
```

@query

```
package com.cybage;

import java.util.Date;
import java.util.List;

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.cybage.model.User;
import com.cybage.repository.UserRepository;

@SpringBootApplication
public class FirstExampleApplication implements CommandLineRunner {

	public static void main(String[] args) {
		SpringApplication.run(FirstExampleApplication.class, args);
	}

	@Autowired
	private UserRepository userRepository;

	@Override
	public void run(String...args) throws Exception {
		User user = new User();
		user.setActive(1);
		user.setAge(28);
		user.setEmailAddress("dm@gmail.com");
		user.setFirstname("dm");
		user.setLastname("jadhav");
		user.setStartDate(new Date());
		user = userRepository.save(user);

		System.out.println("-----------------------------------:: " + user.getId());

		System.out.println(" ----------@Query ----------");
		System.out.println("----------findByEmailAddress ----------");

		User user2 = userRepository.findByEmailAddress("dm@gmail.com");
		System.out.println(user2.toString());

		System.out.println(" ----------@Query ----------");
		System.out.println("----------findByLastname ----------");

		List < User > user3 = userRepository.findByFirstnameEndsWith("dm");
		System.out.println(user3.get(0).toString());
	}

}


package com.cybage.model;
import java.util.Date;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.NamedQueries;
import javax.persistence.NamedQuery;
import javax.persistence.Table;
```

```
@Entity
@Table(name = "users")
public class User {
   private long id;
   private String firstname;
   private String lastname;
   private Date startDate;
   private String emailAddress;
   private int age;
   private int active;

   @Id
   @GeneratedValue(strategy = GenerationType.AUTO)
   public long getId() {
      return id;
   }

   public void setId(long id) {
      this.id = id;
   }

   public String getFirstname() {
      return firstname;
   }

   public void setFirstname(String firstname) {
      this.firstname = firstname;
   }

   public String getLastname() {
      return lastname;
   }

   public void setLastname(String lastname) {
      this.lastname = lastname;
   }

   public Date getStartDate() {
      return startDate;
   }

   public void setStartDate(Date startDate) {
      this.startDate = startDate;
   }

   public int getAge() {
      return age;
   }

   public void setAge(int age) {
      this.age = age;
   }
```

```
   public int getActive() {
      return active;
   }

   public void setActive(int active) {
      this.active = active;
   }

   public String getEmailAddress() {
      return emailAddress;
   }

   public void setEmailAddress(String emailAddress) {
      this.emailAddress = emailAddress;
   }

   @Override
   public String toString() {
      return "User [id=" + id + ", firstname=" + firstname + ", lastname=" + lastname + ", startDate=" + startDate +
         ", emailAddress=" + emailAddress + ", age=" + age + ", active=" + active + "]";
   }
}


package com.cybage.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.cybage.model.User;

@Repository
public interface UserRepository extends JpaRepository < User, Long > {
          @Query("select u from User u where u.emailAddress = ?1")
          User findByEmailAddress(String emailAddress);

          @Query("select u from User u where u.firstname like %?1")
          List < User > findByFirstnameEndsWith(String firstname);

          //native query with @Query
//          @Query(value = "select * from users where first_name like %?1", nativeQuery = true)
//          List < User > findByFirstnameEndsWith(String firstname);
//
//          @Query(value = "SELECT * FROM USERS WHERE EMAIL_ADDRESS = ?1", nativeQuery = true)
//          User findByEmailAddress(String emailAddress);
}
```

sorting

```
package com.cybage;

import java.util.List;
```

```
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.CommandLineRunner;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.data.domain.Page;
import org.springframework.data.domain.PageRequest;
import org.springframework.data.domain.Sort;
import org.springframework.data.web.config.EnableSpringDataWebSupport;

import com.cybage.model.User;
import com.cybage.repository.UserRepository;

@SpringBootApplication
@EnableSpringDataWebSupport
public class FirstExampleApplication implements CommandLineRunner{

	public static void main(String[] args) {
		SpringApplication.run(FirstExampleApplication.class, args);
	}
	@Autowired
	UserRepository ur;

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

		System.out.println("------inserting records-----");
		User user1 = new User("adm101", "swimming", 41);
		User user2 = new User("bdm101", "cycling", 40);
		User user3 = new User("cdm101", "running", 39);
		User user4 = new User("ddm101", "dancing", 38);
		User user5 = new User("edm101", "swimming", 37);
		User user6 = new User("fdm101", "swimming", 42);
		User user7 = new User("fdm101", "swimming", 41);
		User user8 = new User("fdm101", "swimming", 40);
		ur.save(user1);
		ur.save(user2);
		ur.save(user3);
		ur.save(user4);
		ur.save(user5);
		ur.save(user6);
		ur.save(user7);
		ur.save(user8);

		System.out.println("------printing all records-----");
		System.out.println(ur.findAll());

		System.out.println("------printing certain hobbies-----");
		System.out.println(ur.findByHobbyOrderByNameAsc("swimming"));

		System.out.println("------JPQL order by clause (asc)-----");
		System.out.println(ur.findUsersAsc());
		System.out.println("------JPQL order by clause (desc)-----");
		System.out.println(ur.findUsersDesc());
```

```
		System.out.println("-------Sorting with a Sort Parameter-----------");
		System.out.println(ur.findAll(Sort.by(Sort.Direction.ASC, "name")));

		System.out.println("-------Sorting with a Sort Parameter-----------");
		System.out.println(ur.findAll(Sort.by("age")));


		System.out.println("-------Sorting with a Sort Parameter-----------");
		Page<User> page = ur.findAll(PageRequest.of(0, 3, Sort.Direction.ASC, "name"));
		System.out.println(page.getContent());

		//query using query generation and sorting
		System.out.println("query using query generation and sorting");
		List<User> result = ur.findByName("fdm101", Sort.by(Sort.Direction.DESC, "age"));
		System.out.println(result);


		//with sort inside method of @Query
		System.out.println("with sort inside method of @Query");
		System.out.println(ur.myMethodofSorting(Sort.by(Sort.Direction.DESC, "name")));
	}

}
```

```
package com.cybage.model;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Table;



@Entity
@Table(name = "user")
public class User {

	@Id
	@GeneratedValue(strategy = GenerationType.AUTO)
	private long id;

	private String name;

	private String hobby;

	private int age;

	public User() {
		super();
		// TODO Auto-generated constructor stub
	}
```

```
	public User(String name, String hobby, int age) {
		super();
		this.name = name;
		this.hobby = hobby;
		this.age = age;
	}

	public long getId() {
		return id;
	}

	public void setId(long id) {
		this.id = id;
	}

	public String getName() {
		return name;
	}

	public void setName(String name) {
		this.name = name;
	}

	public String getHobby() {
		return hobby;
	}

	public void setHobby(String hobby) {
		this.hobby = hobby;
	}

	public int getAge() {
		return age;
	}

	public void setAge(int age) {
		this.age = age;
	}

	@Override
	public String toString() {
		return "\nUser [id=" + id + ", name=" + name + ", hobby=" + hobby + ", age=" + age + "]";
	}



}


package com.cybage.repository;
import java.util.List;

import org.springframework.data.domain.Sort;
import org.springframework.data.jpa.repository.JpaRepository;
```

```
import org.springframework.data.jpa.repository.Query;
import org.springframework.stereotype.Repository;

import com.cybage.model.User;

@Repository
public interface UserRepository extends JpaRepository<User, Long>{

	public List<User> findByHobby(String Hobby, Sort sort);

	List<User> findByHobbyOrderByNameAsc(String hobby);

	@Query("select u from User u order by u.name asc")
	public List<User> findUsersAsc();

	@Query("select u from User u order by u.name desc")
	public List<User> findUsersDesc();

	//with sort inside method of @Query
	@Query("select u from User u")
	public List<User> myMethodofSorting(Sort sort);

	//query using query generation and sorting
	public List<User> findByName(String name, Sort sort);
}
```

---

sorting and pagination

```
package com.cybage;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.CommandLineRunner;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.data.domain.Page;
import org.springframework.data.domain.PageRequest;
import org.springframework.data.domain.Sort;
import org.springframework.data.web.config.EnableSpringDataWebSupport;

import com.cybage.model.User;
import com.cybage.repository.UserRepository;

@SpringBootApplication
@EnableSpringDataWebSupport
public class FirstExampleApplication implements CommandLineRunner{

	public static void main(String[] args) {
		SpringApplication.run(FirstExampleApplication.class, args);
	}
	@Autowired
	UserRepository ur;
```

```
	@Override
	public void run(String... args) throws Exception {

		System.out.println("------inserting records-----");
		User user1 = new User("adm101", "swimming", 41);
		User user2 = new User("bdm101", "cycling", 40);
		User user3 = new User("cdm101", "running", 39);
		User user4 = new User("ddm101", "dancing", 38);
		User user5 = new User("edm101", "swimming", 37);
		User user6 = new User("fdm101", "swimming", 42);
		User user7 = new User("fdm101", "swimming", 41);
		User user8 = new User("fdm101", "swimming", 40);
		ur.save(user1);
		ur.save(user2);
		ur.save(user3);
		ur.save(user4);
		ur.save(user5);
		ur.save(user6);
		ur.save(user7);
		ur.save(user8);

		System.out.println("------printing all records-----");
		System.out.println(ur.findAll());

		System.out.println("------printing all records-----");
		System.out.println(ur.findAll(Sort.by(Sort.Direction.ASC, "age")));
	}

}
```

```
package com.cybage.model;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Table;



@Entity
@Table(name = "user")
public class User {

	@Id
	@GeneratedValue(strategy = GenerationType.AUTO)
	private long id;

	private String name;

	private String hobby;

	private int age;
```

```
	public User() {
		super();
		// TODO Auto-generated constructor stub
	}

	public User(String name, String hobby, int age) {
		super();
		this.name = name;
		this.hobby = hobby;
		this.age = age;
	}

	public long getId() {
		return id;
	}

	public void setId(long id) {
		this.id = id;
	}

	public String getName() {
		return name;
	}

	public void setName(String name) {
		this.name = name;
	}

	public String getHobby() {
		return hobby;
	}

	public void setHobby(String hobby) {
		this.hobby = hobby;
	}

	public int getAge() {
		return age;
	}

	public void setAge(int age) {
		this.age = age;
	}

	@Override
	public String toString() {
		return "\nUser [id=" + id + ", name=" + name + ", hobby=" + hobby + ", age=" + age + "]";
	}
}


package com.cybage.repository;
import java.util.List;
```

```
import org.springframework.data.domain.Sort;
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.data.jpa.repository.Query;
import org.springframework.data.repository.PagingAndSortingRepository;
import org.springframework.stereotype.Repository;

import com.cybage.model.User;

@Repository
public interface UserRepository extends PagingAndSortingRepository<User, Long>{

}
```

pagination

```
package com.cybage;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.CommandLineRunner;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.data.domain.Page;
import org.springframework.data.domain.PageRequest;
import org.springframework.data.domain.Pageable;
import org.springframework.data.domain.Sort;
import org.springframework.data.web.config.EnableSpringDataWebSupport;

import com.cybage.model.User;
import com.cybage.repository.UserRepository;

@SpringBootApplication
@EnableSpringDataWebSupport
public class FirstExampleApplication implements CommandLineRunner{

	public static void main(String[] args) {
		SpringApplication.run(FirstExampleApplication.class, args);
	}
	@Autowired
	UserRepository ur;

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

		System.out.println("------inserting records-----");
		User user1 = new User("adm101", "swimming", 41);
		User user2 = new User("bdm102", "cycling", 40);
		User user3 = new User("cdm103", "running", 39);
		User user4 = new User("ddm104", "dancing", 38);
		User user5 = new User("edm105", "swimming", 37);
		User user6 = new User("fdm106", "swimming", 42);
		User user7 = new User("fdm107", "swimming", 41);
		User user8 = new User("fdm108", "swimming", 40);
```

```
		ur.save(user1);
		ur.save(user2);
		ur.save(user3);
		ur.save(user4);
		ur.save(user5);
		ur.save(user6);
		ur.save(user7);
		ur.save(user8);

		//without sorting
		Page<User> result = ur.findAll(PageRequest.of(0, 4));
		for(User r : result) {
			System.out.println(r);
		}

		//with sorting
		System.out.println("with sorting");
		result = ur.findAll(PageRequest.of(0, 4, Sort.by("age")));
		for(User r : result) {
			System.out.println(r);
		}
	}
}


package com.cybage.model;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Table;

@Entity
@Table(name = "user")
public class User {

	@Id
	@GeneratedValue(strategy = GenerationType.AUTO)
	private long id;

	private String name;

	private String hobby;

	private int age;

	public User() {
		super();
		// TODO Auto-generated constructor stub
	}

	public User(String name, String hobby, int age) {
		super();
		this.name = name;
```

```
		this.hobby = hobby;
		this.age = age;
	}

	public long getId() {
		return id;
	}

	public void setId(long id) {
		this.id = id;
	}

	public String getName() {
		return name;
	}

	public void setName(String name) {
		this.name = name;
	}

	public String getHobby() {
		return hobby;
	}

	public void setHobby(String hobby) {
		this.hobby = hobby;
	}

	public int getAge() {
		return age;
	}

	public void setAge(int age) {
		this.age = age;
	}

	@Override
	public String toString() {
		return "\nUser [id=" + id + ", name=" + name + ", hobby=" + hobby + ", age=" + age + "]";
	}
}


package com.cybage.repository;
import java.util.List;

import org.springframework.data.domain.PageRequest;
import org.springframework.data.repository.PagingAndSortingRepository;
import org.springframework.stereotype.Repository;

import com.cybage.model.User;

@Repository
public interface UserRepository extends PagingAndSortingRepository<User, Long>{
```

```
}
```

pagination repository

```
package com.cybage;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.data.domain.Page;
import org.springframework.data.domain.PageRequest;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

import com.cybage.model.User;
import com.cybage.repository.UserRepository;

@RestController
public class UserController {
	@Autowired
	UserRepository ur;

	@GetMapping("/user")
	public void getUsers() {
		System.out.println("finding all");
		Page<User> result = ur.findAll(PageRequest.of(0, 3));
		for(User u: result) {
			System.out.println(u);
		}
		System.out.println("find using hobby");
		System.out.println(ur.findByHobby("swimming", PageRequest.of(0, 2)));


		System.out.println("find using age");
		System.out.println(ur.findByAge(35, PageRequest.of(0, 2)));

		System.out.println("finding by name");
		System.out.println(ur.findByName("adm", PageRequest.of(0, 2)));
	}
}


package com.cybage.repository;
import java.util.List;

import org.springframework.data.domain.Page;
import org.springframework.data.domain.Pageable;
import org.springframework.data.domain.Sort;
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.cybage.model.User;
```

```
@Repository
public interface UserRepository extends org.springframework.data.repository.Repository<User, Long>{
	Page<User> findAll(Pageable p);

	public   List<User> findByHobby(String hobby, Pageable pr);
	public List<User> findByAge(int age, Pageable pr);

	//with query annotation
	@Query(value = "select * from user where name = :name", nativeQuery = true)
	public List<User> findByName(@Param("name")String name, Pageable pr);


}
```

---

inheritance single table

```
package com.cybage;
import java.util.ArrayList;
import java.util.List;

import javax.persistence.*;

@Entity
@Inheritance(strategy = InheritanceType.SINGLE_TABLE)
@DiscriminatorColumn(name="EmployeeType", discriminatorType = DiscriminatorType.STRING)
public abstract class Employee {
	@Id
	@GeneratedValue(strategy = GenerationType.AUTO)
	private int id;
	private String name;
	public Employee() {
		super();
	}
	public Employee(String name) {
		super();
		this.name = name;
	}
	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;
	}
	@Override
	public String toString() {
		return "Employee [id=" + id + ", name=" + name + "]";
	}

}
```

```
package com.cybage;

import javax.persistence.DiscriminatorValue;
import javax.persistence.Entity;

@Entity
@DiscriminatorValue(value="CURRENT")
public class Current extends Employee{
	private int salary;

	public Current() {
		super();
	}

	public Current(String name, int salary) {
		super(name);
		this.salary = salary;
	}

	public int getSalary() {
		return salary;
	}

	public void setSalary(int salary) {
		this.salary = salary;
	}

	@Override
	public String toString() {
		return "Current [salary=" + salary + ", getId()=" + getId() + ", getName()=" + getName() + "]";
	}


}


package com.cybage;

import javax.persistence.DiscriminatorValue;
import javax.persistence.Entity;

@Entity
@DiscriminatorValue("RETIRED")
public class Retired extends Employee{
	private int pension;

	public Retired() {
		super();
	}

	public Retired(String name, int pension) {
		super(name);
		this.pension = pension;
```

```
	}

	public int getPension() {
		return pension;
	}

	public void setPension(int pension) {
		this.pension = pension;
	}

	@Override
	public String toString() {
		return "Retired [pension=" + pension + ", getId()=" + getId() + ", getName()=" + getName() + "]";
	}
}
```

```
package com.cybage.repository;

import org.springframework.stereotype.Repository;

@Repository
public interface CurrentRepository extends EmployeeRepository{

}
```

```
package com.cybage.repository;

import org.springframework.data.repository.CrudRepository;
import org.springframework.data.repository.NoRepositoryBean;

import com.cybage.Employee;

@NoRepositoryBean
public interface EmployeeRepository extends CrudRepository<Employee , Integer> {

}
```

```
package com.cybage.repository;

import org.springframework.stereotype.Repository;

@Repository
public interface RetiredRepository extends EmployeeRepository{

}
```

```
package com.cybage;

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.cybage.repository.CurrentRepository;
import com.cybage.repository.RetiredRepository;


@SpringBootApplication(scanBasePackages = "com.cybage")
public class InheritanceSingleTableApplication implements CommandLineRunner{

	@Autowired
	CurrentRepository cr;

	@Autowired
	RetiredRepository rr;

	public static void main(String[] args) {
		SpringApplication.run(InheritanceSingleTableApplication.class, args);
	}

	@Override
	public void run(String... args) throws Exception {
		Employee ce = new Current("dm", 123);
		Employee re = new Retired("dm1", 12);
		cr.save(ce);
		rr.save(re);
	}
}
```

inheritance table per class

```
package com.cybage;
import java.util.ArrayList;
import java.util.List;

import javax.persistence.*;

@Entity
@Inheritance(strategy = InheritanceType.TABLE_PER_CLASS)
public class Employee {
	@Id
	@GeneratedValue(strategy = GenerationType.AUTO)
	private int id;
	private String name;
	public Employee() {
		super();
	}
	public Employee(String name) {
		super();
		this.name = name;
	}
	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;
	}
	@Override
	public String toString() {
		return "Employee [id=" + id + ", name=" + name + "]";
	}

}


package com.cybage;

import javax.persistence.DiscriminatorValue;
import javax.persistence.Entity;

@Entity
@DiscriminatorValue(value="CURRENT")
public class Current extends Employee{
	private int salary;

	public Current() {
		super();
	}

	public Current(String name, int salary) {
		super(name);
		this.salary = salary;
	}

	public int getSalary() {
		return salary;
	}

	public void setSalary(int salary) {
		this.salary = salary;
	}

	@Override
	public String toString() {
		return "Current [salary=" + salary + ", getId()=" + getId() + ", getName()=" + getName() + "]";
	}


}
package com.cybage;

import javax.persistence.DiscriminatorValue;
import javax.persistence.Entity;
```

```
@Entity
@DiscriminatorValue("RETIRED")
public class Retired extends Employee{
	private int pension;

	public Retired() {
		super();
	}

	public Retired(String name, int pension) {
		super(name);
		this.pension = pension;
	}

	public int getPension() {
		return pension;
	}

	public void setPension(int pension) {
		this.pension = pension;
	}

	@Override
	public String toString() {
		return "Retired [pension=" + pension + ", getId()=" + getId() + ", getName()=" + getName() + "]";
	}
}
```

```
package com.cybage.repository;

import org.springframework.stereotype.Repository;

@Repository
public interface CurrentRepository extends EmployeeRepository{

}
```

```
package com.cybage.repository;

import org.springframework.context.annotation.Primary;
import org.springframework.data.repository.CrudRepository;
import org.springframework.data.repository.NoRepositoryBean;
import org.springframework.stereotype.Repository;

import com.cybage.Employee;

@Repository
@Primary
public interface EmployeeRepository extends CrudRepository<Employee , Integer> {
```

```
}

package com.cybage.repository;

import org.springframework.stereotype.Repository;

@Repository
public interface RetiredRepository extends EmployeeRepository{

}


package com.cybage;

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.cybage.repository.CurrentRepository;
import com.cybage.repository.EmployeeRepository;
import com.cybage.repository.RetiredRepository;


@SpringBootApplication(scanBasePackages = "com.cybage")
public class InheritanceTPC implements CommandLineRunner{

	@Autowired
	EmployeeRepository er;

	@Autowired
	CurrentRepository cr;

	@Autowired
	RetiredRepository rr;

	public static void main(String[] args) {
		SpringApplication.run(InheritanceTPC.class, args);
	}

	@Override
	public void run(String... args) throws Exception {
		Employee emp = new Employee("dm3");
		Employee ce = new Current("dm1", 123);
		Employee re = new Retired("dm2", 12);
		er.save(emp);
		cr.save(ce);
		rr.save(re);

		System.out.println(er.findById(3));
	}
}
```

inheritance joined table

```
package com.cybage;
import java.util.ArrayList;
import java.util.List;

import javax.persistence.*;

@Entity
@Inheritance(strategy = InheritanceType.JOINED)
public class Employee {
	@Id
	@GeneratedValue(strategy = GenerationType.AUTO)
	private int id;
	private String name;
	public Employee() {
		super();
	}
	public Employee(String name) {
		super();
		this.name = name;
	}
	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;
	}
	@Override
	public String toString() {
		return "Employee [id=" + id + ", name=" + name + "]";
	}

}


package com.cybage;

import javax.persistence.DiscriminatorValue;
import javax.persistence.Entity;

@Entity
public class Current extends Employee{
	private int salary;

	public Current() {
		super();
```

```
	}

	public Current(String name, int salary) {
		super(name);
		this.salary = salary;
	}

	public int getSalary() {
		return salary;
	}

	public void setSalary(int salary) {
		this.salary = salary;
	}

	@Override
	public String toString() {
		return "Current [salary=" + salary + ", getId()=" + getId() + ", getName()=" + getName() + "]";
	}


}


package com.cybage;

import javax.persistence.DiscriminatorValue;
import javax.persistence.Entity;

@Entity
public class Retired extends Employee{
	private int pension;

	public Retired() {
		super();
	}

	public Retired(String name, int pension) {
		super(name);
		this.pension = pension;
	}

	public int getPension() {
		return pension;
	}

	public void setPension(int pension) {
		this.pension = pension;
	}

	@Override
	public String toString() {
		return "Retired [pension=" + pension + ", getId()=" + getId() + ", getName()=" + getName() + "]";
	}
```

```
}


package com.cybage.repository;

import org.springframework.stereotype.Repository;

@Repository
public interface CurrentRepository extends EmployeeRepository{

}

package com.cybage.repository;

import org.springframework.context.annotation.Primary;
import org.springframework.data.repository.CrudRepository;
import org.springframework.data.repository.NoRepositoryBean;
import org.springframework.stereotype.Repository;

import com.cybage.Employee;

@NoRepositoryBean
public interface EmployeeRepository extends CrudRepository<Employee , Integer> {

}


package com.cybage.repository;

import org.springframework.stereotype.Repository;

@Repository
public interface RetiredRepository extends EmployeeRepository{

}


package com.cybage;

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.cybage.repository.CurrentRepository;
import com.cybage.repository.EmployeeRepository;
import com.cybage.repository.RetiredRepository;


@SpringBootApplication(scanBasePackages = "com.cybage")
public class InheritanceJT implements CommandLineRunner{

	@Autowired
	CurrentRepository cr;
```

```
	@Autowired
	RetiredRepository rr;

	public static void main(String[] args) {
		SpringApplication.run(InheritanceJT.class, args);
	}

	@Override
	public void run(String... args) throws Exception {
		Employee emp = new Employee("dm3");
		Employee ce = new Current("dm1", 123);
		Employee re = new Retired("dm2", 12);
		cr.save(ce);
		rr.save(re);

		System.out.println(cr.findById(1));
		System.out.println(cr.findById(2));
	}
}
```

---

collection mapping

```
package com.cybage;
import java.util.ArrayList;
import java.util.List;

import javax.persistence.*;

@Entity
public class Employee {
	@Id
	@GeneratedValue(strategy = GenerationType.AUTO)
	private int id;
	private String name;

	@ElementCollection(fetch = FetchType.EAGER)
	@CollectionTable(name="emp_phone",
	joinColumns = @JoinColumn(name = "id"))
	private List<String> phone = new ArrayList<String>();

	public Employee() {
		super();
	}
	public Employee(int id, String name, List<String> phone) {
		super();
		this.id = id;
		this.name = name;
		this.phone = phone;
	}
	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 List<String> getPhone() {
		return phone;
	}
	public void setPhone(List<String> phone) {
		this.phone = phone;
	}
	@Override
	public String toString() {
		return "Employee [id=" + id + ", name=" + name + ", phone=" + phone + "]";
	}


}


package com.cybage;

import java.util.ArrayList;
import java.util.List;

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.cybage.repository.EmployeeRepository;


@SpringBootApplication(scanBasePackages = "com.cybage")
public class CollectionMapping implements CommandLineRunner{

	@Autowired
	EmployeeRepository er;

	public static void main(String[] args) {
		SpringApplication.run(CollectionMapping.class, args);
	}

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

		List<String > phones = new ArrayList<String>();
		phones.add("1234");
		phones.add("2234");
		phones.add("3234");
		phones.add("4234");
```

```
		Employee emp = new Employee();
		emp.setId(101);
		emp.setName("dm101");
		emp.setPhone(phones);

		er.save(emp);

		System.out.println(er.findById(1));
	}
}
```

```
package com.cybage.repository;

import org.springframework.context.annotation.Primary;
import org.springframework.data.repository.CrudRepository;
import org.springframework.data.repository.NoRepositoryBean;
import org.springframework.stereotype.Repository;

import com.cybage.Employee;

@Repository
public interface EmployeeRepository extends CrudRepository<Employee , Integer> {

}
```

---

one to one mapping

```
package com.cybage;
import java.util.ArrayList;
import java.util.List;

import javax.persistence.*;

@Entity
public class Employee {
	@Id
	private int empId;
	private String name;

	@OneToOne
	(fetch = FetchType.LAZY,
	cascade = CascadeType.ALL,
	mappedBy = "employee")
	private Account account;

	public Employee() {
		super();
	}

	public Employee(int empId, String name, Account account) {
		super();
		this.empId = empId;
		this.name = name;
		this.account = account;
```

```
	}

	public int getEmpId() {
		return empId;
	}

	public void setEmpId(int empId) {
		this.empId = empId;
	}

	public String getName() {
		return name;
	}

	public void setName(String name) {
		this.name = name;
	}

	public Account getAccount() {
		return account;
	}

	public void setAccount(Account account) {
		this.account = account;
	}

	@Override
	public String toString() {
		return "Employee [empId=" + empId + ", name=" + name + ", account=" + account + "]";
	}
}

package com.cybage;

import javax.persistence.Entity;
import javax.persistence.FetchType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.OneToOne;

@Entity
public class Account {
	@Id
	private int accountId;
	private String accountName;

	@OneToOne(fetch = FetchType.LAZY, optional = false)
	@JoinColumn(name="empId", nullable = false)
	private Employee employee;
	public Account() {
		super();
	}
	public Account(int accountId, String accountName, Employee employee) {
		super();
```

```
		this.accountId = accountId;
		this.accountName = accountName;
		this.employee = employee;
	}
	public int getAccountId() {
		return accountId;
	}
	public void setAccountId(int accountId) {
		this.accountId = accountId;
	}
	public String getAccountName() {
		return accountName;
	}
	public void setAccountName(String accountName) {
		this.accountName = accountName;
	}
	public Employee getEmployee() {
		return employee;
	}
	public void setEmployee(Employee employee) {
		this.employee = employee;
	}
	@Override
	public String toString() {
		return "Account [accountId=" + accountId + ", accountName=" + accountName + "]";
	}


}

package com.cybage;

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.cybage.repository.AccountRepository;
import com.cybage.repository.EmployeeRepository;


@SpringBootApplication(scanBasePackages = "com.cybage")
public class OneToOneMapping implements CommandLineRunner{

	@Autowired
	EmployeeRepository er;

	@Autowired
	AccountRepository ar;

	public static void main(String[] args) {
		SpringApplication.run(OneToOneMapping.class, args);
	}
```

```
	@Override
	public void run(String... args) throws Exception {

		Account account = new Account();
		account.setAccountId(101111);
		account.setAccountName("account name");

		Employee emp = new Employee();
		emp.setEmpId(101);
		emp.setName("dm101");
		emp.setAccount(account);
		account.setEmployee(emp);

		er.save(emp);

		System.out.println(er.findAll());
	}
}
```

```
package com.cybage.repository;

import org.springframework.data.repository.CrudRepository;
import org.springframework.stereotype.Repository;

import com.cybage.Account;

@Repository
public interface AccountRepository extends CrudRepository<Account, Integer> {

}
```

```
package com.cybage.repository;

import org.springframework.context.annotation.Primary;
import org.springframework.data.repository.CrudRepository;
import org.springframework.data.repository.NoRepositoryBean;
import org.springframework.stereotype.Repository;

import com.cybage.Employee;

@Repository
public interface EmployeeRepository extends CrudRepository<Employee , Integer> {

}
```

---

one to many mapping

```
package com.cybage;

import java.util.List;

import javax.persistence.*;
```

```
@Entity
public class Employee {
	@Id
	private int empId;
	private String name;
	@OneToMany(targetEntity = Account.class, cascade = CascadeType.ALL, fetch = FetchType.EAGER)
	@JoinColumn(name = "ca_fk", referencedColumnName = "empId")
	private List<Account> accounts;
	public Employee() {
		super();
	}
	public Employee(int empId, String name, List<Account> accounts) {
		super();
		this.empId = empId;
		this.name = name;
		this.accounts = accounts;
	}
	public int getEmpId() {
		return empId;
	}
	public void setEmpId(int empId) {
		this.empId = empId;
	}
	public String getName() {
		return name;
	}
	public void setName(String name) {
		this.name = name;
	}
	public List<Account> getAccounts() {
		return accounts;
	}
	public void setAccounts(List<Account> accounts) {
		this.accounts = accounts;
	}
	@Override
	public String toString() {
		return "Employee [empId=" + empId + ", name=" + name + ", accounts=" + accounts + "]";
	}

}

package com.cybage;

import javax.persistence.Entity;
import javax.persistence.FetchType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.ManyToOne;
import javax.persistence.OneToOne;

import org.hibernate.annotations.OnDelete;
import org.hibernate.annotations.OnDeleteAction;
```

```
@Entity
public class Account {
	@Id
	private int accountId;
	private String accountName;

	public Account() {
		super();
	}

	public Account(int accountId, String accountName) {
		super();
		this.accountId = accountId;
		this.accountName = accountName;
	}

	public int getAccountId() {
		return accountId;
	}

	public void setAccountId(int accountId) {
		this.accountId = accountId;
	}

	public String getAccountName() {
		return accountName;
	}

	public void setAccountName(String accountName) {
		this.accountName = accountName;
	}

	@Override
	public String toString() {
		return "Account [accountId=" + accountId + ", accountName=" + accountName + "]";
	}

}


package com.cybage;

import java.util.ArrayList;
import java.util.List;

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.cybage.repository.AccountRepository;
import com.cybage.repository.EmployeeRepository;

@SpringBootApplication(scanBasePackages = "com.cybage")
```

```
public class OneToManyMapping implements CommandLineRunner{

	@Autowired
	EmployeeRepository er;

	@Autowired
	AccountRepository ar;

	public static void main(String[] args) {
		SpringApplication.run(OneToManyMapping.class, args);
	}

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

		List<Account> accounts = new ArrayList<Account>();
		accounts.add(new Account(11111, "account1"));
		accounts.add(new Account(22222, "account2"));

		Employee emp = new Employee();
		emp.setEmpId(101);
		emp.setName("dm101");
		emp.setAccounts(accounts);

		er.save(emp);
		System.out.println(er.findById(101));
		System.out.println(er.findNameAndAccount());
	}
}



package com.cybage.repository;

import java.util.List;

import org.springframework.context.annotation.Primary;
import org.springframework.data.jpa.repository.Query;
import org.springframework.data.repository.CrudRepository;
import org.springframework.data.repository.NoRepositoryBean;
import org.springframework.stereotype.Repository;

import com.cybage.Employee;

@Repository
public interface EmployeeRepository extends CrudRepository<Employee , Integer> {

	@Query("select new com.cybage.repository.AccountDto(e.name, a.accountId) from Employee e join e.accounts
a")
	public List<AccountDto> findNameAndAccount();
}

package com.cybage.repository;
```

```
import org.springframework.data.repository.CrudRepository;
import org.springframework.stereotype.Repository;

import com.cybage.Account;

@Repository
public interface AccountRepository extends CrudRepository<Account, Integer> {

}
```