

Spring Boot - Updating Seed Data

Add **AccountRepository.java** in **repository** folder.

Add **AccountService.java** in **services** folder.

AccountRepository.java

```
package org.studyeasy.SpringStarter.repositories;

import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;
import org.studyeasy.SpringStarter.models.Account;

@Repository
public interface AccountRepository extends JpaRepository<Account, Long>{

}
```

AccountService.java

```
package org.studyeasy.SpringStarter.services;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import org.studyeasy.SpringStarter.models.Account;
import org.studyeasy.SpringStarter.repositories.AccountRepository;

@Service
public class AccountService {

    @Autowired
    private AccountRepository accountRepository;

    public Account save(Account account){
        return accountRepository.save(account);
    }

}
```

Modification in SeedData.java

```
package org.studyeasy.SpringStarter.config;

import java.util.List;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.CommandLineRunner;
import org.springframework.stereotype.Component;
import org.studyeasy.SpringStarter.models.Account;
import org.studyeasy.SpringStarter.models.Post;
import org.studyeasy.SpringStarter.services.AccountService;
import org.studyeasy.SpringStarter.services.PostService;

@Component
public class SeedData implements CommandLineRunner{

    @Autowired
    private PostService postService;

    @Autowired
    private AccountService accountService;

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

        //Account
        Account account01 = new Account();
        Account account02 = new Account();
```

```

account01.setEmail("account01@studyeasy.org");
account01.setPassword("password");
account01.setFirstname("user01");

account02.setEmail("account02@studyeasy.org");
account02.setPassword("password");
account02.setFirstname("user02");

accountService.save(account01);
accountService.save(account02);

//Post
List<Post> posts = postService.getAll();
if(posts.size() == 0){
    Post post01 = new Post();
    post01.setTitle("Post01");
    post01.setBody("Post 01 Body .....");

    //added
    post01.setAccount(account01);

    postService.save(post01);

    Post post02 = new Post();
    post02.setTitle("Post02");
    post02.setBody("Post 02 Body .....");

    //added
    post02.setAccount(account02);

    postService.save(post02);
}
}

```

The screenshot shows the H2 Console interface. The SQL statement entered is `SELECT * FROM ACCOUNT;`. The result is displayed as a table with 2 rows and 4 columns: ID, EMAIL, FIRSTNAME, and PASSWORD.

ID	EMAIL	FIRSTNAME	PASSWORD
1	account01@studyeasy.org	user01	password
2	account02@studyeasy.org	user02	password

(2 rows, 1 ms)

The screenshot shows the H2 Console interface. The SQL statement entered is `SELECT * FROM POST;`. The result is displayed as a table with 4 columns: ACCOUNT_ID, CREATED_AT, ID, BODY, and TITLE. There are 2 rows of data.

ACCOUNT_ID	CREATED_AT	ID	BODY	TITLE
1	2024-06-16 17:05:28.330544	1	Post 01 Body	Post01
2	2024-06-16 17:05:28.342998	2	Post 02 Body	Post02

(2 rows, 1 ms)