

# Sportyshoes source code

## SportShoesApplication.java

```
package com.cisco.SportyShoes;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.boot.autoconfigure.domain.EntityScan;
import org.springframework.context.annotation.ComponentScan;
import org.springframework.data.jpa.repository.config.EnableJpaRepositories;

@SpringBootApplication
@ComponentScan("com.cisco")

@EntityScan("com.cisco.model")
@EnableJpaRepositories("com.cisco.dao")
public class SportyShoesApplication {

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

## OrdersDao.java

```
package com.cisco.dao;

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

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

import com.cisco.model.OrderedShoeModel;

@Repository
public interface OrdersDao extends CrudRepository<OrderedShoeModel, Integer>{

    @Query("from OrderedShoeModel where categeory=:categeoryId and date=:date")
    public List<OrderedShoeModel> getRequiredCompleteTransactionsData(int categeoryId,Date date);

}
```

## ShoesDao.java

```
package com.cisco.dao;

import org.springframework.data.jpa.repository.Modifying;
import org.springframework.data.jpa.repository.Query;
import org.springframework.data.repository.CrudRepository;
import org.springframework.data.repository.query.Param;
import org.springframework.stereotype.Repository;
import org.springframework.transaction.annotation.Transactional;

import com.cisco.model.ShoesDataModel;

import antlr.collections.List;

//@Repository
```

```
//public interface ShoesDao extends CrudRepository<ShoesDataModel, Integer> {

@Repository
public interface ShoesDao extends CrudRepository<ShoesDataModel, Integer> {

    @Query("from ShoesDataModel where categeory=:categeory")
    public Iterable<ShoesDataModel> findByCategory(int categeory);

    @Query("from ShoesDataModel where id=:seletedShoeId")
    public ShoesDataModel getshoesDataById(int seletedShoeId);

    @Modifying
    @Transactional
    @Query("delete from ShoesDataModel where id=:id")
    public int deleteProductwithId(@Param("id") int id);

    @Modifying
    @Transactional
    @Query("update ShoesDataModel set categeory=:categeory, price=:price, name=:name,
imagelink=:imagelink where id=:id")
    public int updateShoeProduct(@Param("categeory") int categeory, @Param("price") double price,
                                @Param("name") String name, @Param("imagelink") String imagelink,
                                @Param("id") int id);

}

```

## UsersDao.java

```
package com.cisco.dao;

import org.springframework.data.jpa.repository.Modifying;
import org.springframework.data.jpa.repository.Query;
import org.springframework.data.repository.CrudRepository;
import org.springframework.data.repository.query.Param;
import org.springframework.stereotype.Repository;
import org.springframework.transaction.annotation.Transactional;

import com.cisco.model.UserInfoModel;

@Repository
public interface UsersDao extends CrudRepository<UserInfoModel, Integer> {

    @Query("from UserInfoModel where firstname=:username and password=:password")
    public UserInfoModel isValidAdminUser(String username,String password );

    @Query("from UserInfoModel where rollid=:rollid")
    public Iterable<UserInfoModel> getUserDatabyRollid(int rollid);

    @Query("from UserInfoModel where password=:password")
    public UserInfoModel isCorrectPassword_or_Not(String password);

    @Modifying
    @Transactional
    @Query("update UserInfoModel set password = :password where rollid=1 and firstname=:username")
    public int updateAdminPassword(@Param("password") String password, @Param("username") String username);

}

```