

In this lab you'll practice using Jdbc with Spring.

Note: You'll use this code when you learn transaction management in spring. So, I would suggest creating it in a completely new package or a Spring project.

create a database 'accountsdb' in MySQL.

Copy the following SQL queries in MySQL query window to create a table "accounts" and "statements".

```
create table `accountsdb`.`accounts` ( `account_number` int (4) NOT NULL  
AUTO_INCREMENT , `owner` varchar (40) NULL , `balance` int (7) NULL ,  
PRIMARY KEY ( `account_number` ) );
```

```
create table `accountsdb`.`statements` ( `id` int (4) NOT NULL  
AUTO_INCREMENT, `account_number` int (4) NULL, `amount` int (7) NULL ,  
`type` varchar (40) NULL, PRIMARY KEY ( `id` ) );
```

```
INSERT INTO accounts(owner,balance) VALUES('John',2000);
```

Implement these classes using JDBC with Spring.

```
public interface AccountDao {  
    void deposit(int accountNumber,int amount);  
    void withdraw(int accountNumber,int amount);  
}  
  
public interface StatementDao {  
    void addStatement(int accountNumber,String type,int amount);  
}  
  
public abstract class BaseDaoSupport{  
    private JdbcTemplate jdbcTemplate;  
    //ADD necessary code  
}  
public class AccountDaoImpl extends BaseDaoSupport implements AccountDao  
{  
    public void deposit(int accountNumber,int amount)  
    {  
        //update balance in accounts table  
    }  
    public void withdraw(int accountNumber,int amount)  
    {  
        //update balance in accounts table  
    }  
}
```

```
    }  
}  
  
public class StatementDaoImpl extends BaseDaoSupport implements StatementDao  
{  
    public void addStatement(int accountNumber,String type,int amount)  
    {  
        //insert into statements table  
        //type will be either “Withdraw” or “Deposit” or “Interest” depending on  
        the operation.  
    }  
}  
  
public class AccountService  
{  
    private AccountDao accountDao;  
    private StatementDao statementDao;  
    public void deposit(int accountNumber,String type, int amount) {  
        //call AccountDao’s deposit method  
        //call StatementDao’s addStatement method  
    }  
    public void withdraw(int accountNumber,String type, int amount)  
    {  
        //call AccountDao’s withdraw method  
        //call StatementDao’s addStatement method  
    }  
}
```

Have a client to call the deposit and withdraw functions in AccountService.