1. Process Flow:

extends JpaRepository<OuMasterEntity, Long>

*interface*

*implements*

Controller

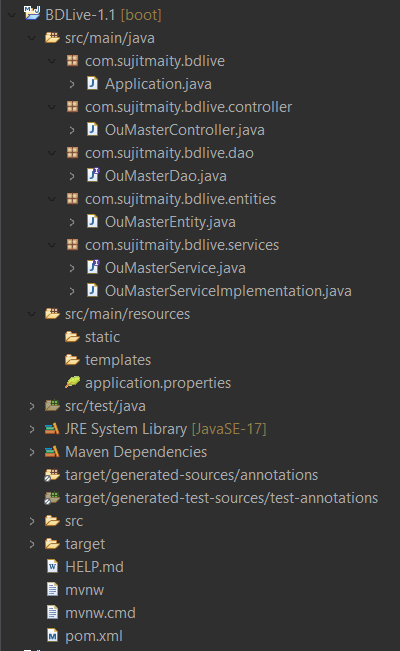
Services and Service Imp

DAO or Repository

*@Repository*

Entity or Model

1. Package:



1. Database Configuration:

Application.properties:

##connect to database

spring.datasource.driver-class-name=org.postgresql.Driver

spring.datasource.url=jdbc:postgresql://localhost:5432/BharatBillPay

spring.datasource.username=postgres

spring.datasource.password=Sujit\*\*\*\*\*\*

server.port=9000

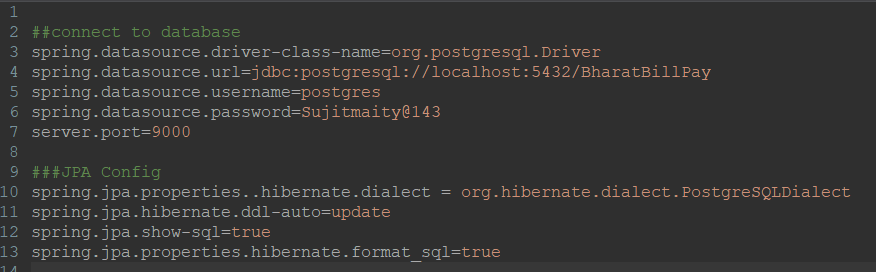
###JPA Config

spring.jpa.properties..hibernate.dialect = org.hibernate.dialect.PostgreSQLDialect

spring.jpa.hibernate.ddl-auto=update

spring.jpa.show-sql=true

spring.jpa.properties.hibernate.format\_sql=true





1. Package and Class Implementation:

Entity or Model Class

package com.sujitmaity.bdlive.entity;

import java.sql.Timestamp;

import jakarta.persistence.Column;

import jakarta.persistence.Entity;

import jakarta.persistence.GeneratedValue;

import jakarta.persistence.GenerationType;

import jakarta.persistence.Id;

import jakarta.persistence.Table;

import lombok.AllArgsConstructor;

import lombok.Getter;

import lombok.NoArgsConstructor;

import lombok.Setter;

import lombok.ToString;

*@*Getter

*@*Setter

*@*ToString

*@*NoArgsConstructor

*@*AllArgsConstructor

*@*Entity

*@*Table(name = "user\_master")

public class UserMasterEntity {

*@*Id

*@*GeneratedValue(strategy = GenerationType.IDENTITY)

*@*Column(name = "userId")

Long userId;

*@*Column(name = "userName")

String userName;

*@*Column(name = "userPassword")

String userPassword;

*@*Column(name = "userEmail")

String userEmail;

*@*Column(name = "userMobile")

String userMobile;

*@*Column(name = "userActivationStatus")

boolean userActivationStatus;

*@*Column(name = "userActivationTime")

Timestamp userActivationTime;

public Long getUserId() {

return userId;

}

public void setUserId(Long userId) {

this.userId = userId;

}

public String getUserName() {

return userName;

}

public void setUserName(String userName) {

this.userName = userName;

}

public String getUserPassword() {

return userPassword;

}

public void setUserPassword(String userPassword) {

this.userPassword = userPassword;

}

public String getUserEmail() {

return userEmail;

}

public void setUserEmail(String userEmail) {

this.userEmail = userEmail;

}

public String getUserMobile() {

return userMobile;

}

public void setUserMobile(String userMobile) {

this.userMobile = userMobile;

}

public boolean isUserActivationStatus() {

return userActivationStatus;

}

public void setUserActivationStatus(boolean userActivationStatus) {

this.userActivationStatus = userActivationStatus;

}

public Timestamp getUserActivationTime() {

return userActivationTime;

}

public void setUserActivationTime(Timestamp userActivationTime) {

this.userActivationTime = userActivationTime;

}

public UserMasterEntity(Long userId, String userName, String userPassword, String userEmail, String userMobile,

boolean userActivationStatus, Timestamp userActivationTime) {

super();

this.userId = userId;

this.userName = userName;

this.userPassword = userPassword;

this.userEmail = userEmail;

this.userMobile = userMobile;

this.userActivationStatus = userActivationStatus;

this.userActivationTime = userActivationTime;

}

public UserMasterEntity() {

super();

}

*@*Override

public String toString() {

return "UserMasterEntity [userId=" + userId + ", userName=" + userName + ", userPassword=" + userPassword

+ ", userEmail=" + userEmail + ", userMobile=" + userMobile + ", userActivationStatus="

+ userActivationStatus + ", userActivationTime=" + userActivationTime + "]";

}

}

DAO or Repository Class

package com.sujitmaity.bdlive.dao;

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

import org.springframework.stereotype.Repository;

import com.sujitmaity.bdlive.entity.UserMasterEntity;

@Repository

public interface UserMasterDao extends JpaRepository<UserMasterEntity, Long> {

}

Services and Service Implementation Class

package com.sujitmaity.bdlive.services;

import java.util.List;

import java.util.Optional;

import com.sujitmaity.bdlive.entity.UserMasterEntity;

public interface UserMasterService {

List<UserMasterEntity> findAllUser();

Optional<UserMasterEntity> findByUserId(Long userId);

UserMasterEntity saveUser(UserMasterEntity userMaster);

}

Service Implementation:

package com.sujitmaity.bdlive.services;

import java.util.List;

import java.util.Optional;

import org.springframework.stereotype.Service;

import com.sujitmaity.bdlive.dao.UserMasterDao;

import com.sujitmaity.bdlive.entity.UserMasterEntity;

*@Service*

public class UserMasterServicesImplement implements UserMasterService{

private final UserMasterDao userdao ;

public UserMasterServicesImplement(UserMasterDao userdao) {

this.userdao = userdao;

}

*@Override*

public List<UserMasterEntity> findAllUser() {

return userdao.findAll() ;

}

*@Override*

public Optional<UserMasterEntity> findByUserId(Long userId) {

return userdao.findById(userId);

}

*@Override*

public UserMasterEntity saveUser(UserMasterEntity userMaster) {

return userdao.save(userMaster);

}

}

Controller Class

package com.sujitmaity.bdlive.controller;

import java.util.List;

import java.util.Optional;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.RequestBody;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RestController;

import com.sujitmaity.bdlive.entity.UserMasterEntity;

import com.sujitmaity.bdlive.services.UserMasterService;

@RestController

@RequestMapping("/userMaster")

public class UserMasterController {

private final UserMasterService userService;

public UserMasterController(UserMasterService userService) {

this.userService = userService;

}

@GetMapping("/userall")

public List<UserMasterEntity> findAllUser(){

return userService.findAllUser();

}

@GetMapping("/{userId}")

public Optional<UserMasterEntity> findByUserId(@PathVariable("userId") Long userId){

return userService.findByUserId(userId);

}

@PostMapping

public UserMasterEntity saveUser(@RequestBody UserMasterEntity userMasterEntity) {

return userService.saveUser(userMasterEntity);

}

}