**Online Collaboration**

**🡪** Create a SpringBoot project named OnlineCollaborate with web, Spring Data JPA, SpringBoot Dev Tools and MySQL Server Driver packages. Extract that project

🡪 Import the project in Eclipse.

🡪Open **package** com.coll.OnlineCollaborate;

🡪Create **package** com.coll.OnlineCollaborate.config;

🡪Configuration File as HibernateConfig:-

package com.coll.OnlineCollaborate.config;

import java.util.Properties;

import javax.sql.DataSource;

import org.hibernate.SessionFactory;

import org.springframework.boot.autoconfigure.EnableAutoConfiguration;

import org.springframework.boot.autoconfigure.orm.jpa.HibernateJpaAutoConfiguration;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.ComponentScan;

import org.springframework.context.annotation.ComponentScans;

import org.springframework.context.annotation.Configuration;

import org.springframework.http.converter.json.MappingJackson2HttpMessageConverter;

import org.springframework.jdbc.datasource.DriverManagerDataSource;

import org.springframework.orm.hibernate5.HibernateTransactionManager;

import org.springframework.orm.hibernate5.LocalSessionFactoryBean;

import org.springframework.orm.hibernate5.LocalSessionFactoryBuilder;

import org.springframework.transaction.annotation.EnableTransactionManagement;

import org.springframework.web.servlet.ViewResolver;

import org.springframework.web.servlet.view.InternalResourceViewResolver;

import com.fasterxml.jackson.databind.DeserializationFeature;

import com.fasterxml.jackson.databind.ObjectMapper;

@Configuration

@ComponentScans(value= {@ComponentScan("com.coll.OnlineCollaborate"),

@ComponentScan("model"),

@ComponentScan("controller"),

@ComponentScan("dao"),

@ComponentScan("daoImpl"),

@ComponentScan("config"),

@ComponentScan("serviceImpl"),

@ComponentScan("service"),

@ComponentScan("service")})

@EnableAutoConfiguration(exclude = {HibernateJpaAutoConfiguration.class})

@EnableTransactionManagement

public class HibernateConfig {

public static final String DATABASE\_URL="jdbc:mysql://localhost:3306/collaboration";

public static final String DATABASE\_DRIVER="com.mysql.cj.jdbc.Driver";

public static final String DATABASE\_DIALECT="org.hibernate.dialect.MySQLDialect";

public static final String DATABASE\_USERNAME="root";

public static final String DATABASE\_PASSWORD="ganesh@7";

@Bean(name="dataSource")

public DataSource getDataSource() {

DriverManagerDataSource dataSource=new DriverManagerDataSource();

dataSource.setDriverClassName(DATABASE\_DRIVER);

dataSource.setUrl(DATABASE\_URL);

dataSource.setUsername(DATABASE\_USERNAME);

dataSource.setPassword(DATABASE\_PASSWORD);

return dataSource;

}

@Bean

public LocalSessionFactoryBean getSessionFactory() {

LocalSessionFactoryBean sessionFactory = new LocalSessionFactoryBean();

sessionFactory.setDataSource(getDataSource());

sessionFactory.setPackagesToScan("com.coll.OnlineCollaborate");

Properties hibernateProperties =new Properties();

hibernateProperties.put("hibernate.dialect",DATABASE\_DIALECT);

hibernateProperties.put("hibernate.show\_sql","true");

hibernateProperties.put("hibernate.hbm2ddl.auto","update");

sessionFactory.setHibernateProperties(hibernateProperties);

return sessionFactory;

}

@Bean

public HibernateTransactionManager getTransactionManger() {

HibernateTransactionManager txm=new HibernateTransactionManager();

txm.setSessionFactory(getSessionFactory().getObject());

return txm;

}

@Bean

public ViewResolver jspViewResolver() {

InternalResourceViewResolver viewResolver= new InternalResourceViewResolver();

viewResolver.setPrefix("/views/");

viewResolver.setSuffix(".jsp");

return viewResolver;

}

@Bean

public MappingJackson2HttpMessageConverter mappingJackson2HttpMessageConverter() {

MappingJackson2HttpMessageConverter jsonConverter = new MappingJackson2HttpMessageConverter();

ObjectMapper objectMapper = new ObjectMapper();

objectMapper.configure(DeserializationFeature.FAIL\_ON\_UNKNOWN\_PROPERTIES, false);

jsonConverter.setObjectMapper(objectMapper);

return jsonConverter;

}

}

🡪Create Schema as Collaboration

🡪Active Collaboration Schema by following commend

Use Collaboration;

🡪Create Tables

1**.User Table:-**

create table User(

userId int not null auto\_increment,

firstName varchar(30),

lastName varchar(30),

username varchar(20),

password varchar(30),

email varchar(30),

role varchar(30),

status varchar(10),

isOnline boolean,

enabled boolean,

primary key(userId)

);

2.**Blog Table**

create table Blog (

blogId integer not null auto\_increment,

blogContent varchar(255),

blogPosted date,

blogTitel varchar(255),

noFoComments integer not null,

noOfLikes integer not null,

noOfViews integer not null,

status varchar(255),

userId integer not null,

username varchar(255),

primary key (blogId));

**3.BlogComments Table;-**

create table BlogComments (

blogCommentId integer not null auto\_increment,

blogComment varchar(255),

currentDate date,

onOfLikes integer not null,

titel varchar(255),

userId integer not null,

userProfileId varchar(255),

username varchar(255),

blogId integer ,

primary key (blogCommentId));

🡪Create **package** com.coll.OnlineCollaborate.model

🡪 Create model classes:

**1.User Class**

package com.coll.OnlineCollaborate.model;

import java.io.Serializable;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.Transient;

import org.springframework.stereotype.Component;

@Component

@Entity

public class User extends DomainResponse implements Serializable {

private static final long serialVersionUID=1L;

@Id

@GeneratedValue(strategy=GenerationType.IDENTITY)

private int userId;

private String firstName;

private String lastName;

private String username;

private String password;

private String email;

private String role;

private String status;

private String isOnline;

private String enabled;

public int getUserId() {

return userId;

}

public void setUserId(int userId) {

this.userId = userId;

}

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 String getUsername() {

return username;

}

public void setUsername(String username) {

this.username = username;

}

public String getPassword() {

return password;

}

public void setPassword(String password) {

this.password = password;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

public String getRole() {

return role;

}

public void setRole(String role) {

this.role = role;

}

public String getStatus() {

return status;

}

public void setStatus(String status) {

this.status = status;

}

public String getIsOnline() {

return isOnline;

}

public void setIsOnline(String isOnline) {

this.isOnline = isOnline;

}

public String getEnabled() {

return enabled;

}

public void setEnabled(String enabled) {

this.enabled = enabled;

}

}

**2.Blog:-**

package com.coll.OnlineCollaborate.model;

import java.io.Serializable;

import java.time.LocalDate;

import java.time.LocalDateTime;

import java.util.List;

import javax.persistence.CascadeType;

import javax.persistence.Entity;

import javax.persistence.FetchType;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.OneToMany;

import org.springframework.stereotype.Component;

import com.fasterxml.jackson.annotation.JsonManagedReference;

import com.fasterxml.jackson.databind.annotation.JsonDeserialize;

import com.fasterxml.jackson.datatype.jsr310.deser.LocalDateTimeDeserializer;

import com.fasterxml.jackson.datatype.jsr310.ser.LocalDateTimeSerializer;

@Component

@Entity

public class Blog extends DomainResponse implements Serializable{

private static final long serialVersionUID=1L;

@Id

@GeneratedValue(strategy=GenerationType.IDENTITY)

int blogId;

String blogTitel, blogContent;

@JsonDeserialize(using = LocalDateTimeDeserializer.class)

LocalDateTime blogPosted;

String status;

int noOfLikes,noFoComments, noOfViews;

int userId;

String username;

@OneToMany(mappedBy="blog", fetch=FetchType.EAGER,cascade=CascadeType.ALL)

@JsonManagedReference

List<BlogComments> blogComments;

public int getBlogId() {

return blogId;

}

public void setBlogId(int blogId) {

this.blogId = blogId;

}

public String getBlogTitel() {

return blogTitel;

}

public void setBlogTitel(String blogTitel) {

this.blogTitel = blogTitel;

}

public String getBlogContent() {

return blogContent;

}

public void setBlogContent(String blogContent) {

this.blogContent = blogContent;

}

public LocalDateTime getBlogPosted() {

return blogPosted;

}

public void setBlogPosted(LocalDateTime blogPosted) {

this.blogPosted = blogPosted;

}

public String getStatus() {

return status;

}

public void setStatus(String status) {

this.status = status;

}

public int getNoOfLikes() {

return noOfLikes;

}

public void setNoOfLikes(int noOfLikes) {

this.noOfLikes = noOfLikes;

}

public int getNoFoComments() {

return noFoComments;

}

public void setNoFoComments(int noFoComments) {

this.noFoComments = noFoComments;

}

public int getNoOfViews() {

return noOfViews;

}

public void setNoOfViews(int noOfViews) {

this.noOfViews = noOfViews;

}

public int getUserId() {

return userId;

}

public void setUserId(int userId) {

this.userId = userId;

}

public String getUsername() {

return username;

}

public void setUsername(String username) {

this.username = username;

}

public List<BlogComments> getBlogComments() {

return blogComments;

}

public void setBlogComments(List<BlogComments> blogComments) {

this.blogComments = blogComments;

}

}

**3.BlogComments:-**

package com.coll.OnlineCollaborate.model;

import java.io.Serializable;

import java.time.LocalDate;

import java.util.Locale;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.JoinColumn;

import javax.persistence.ManyToOne;

import org.springframework.stereotype.Component;

import com.fasterxml.jackson.annotation.JsonBackReference;

@Component

@Entity

public class BlogComments implements Serializable {

private static final long serialVersionUID=1L;

@Id

@GeneratedValue(strategy=GenerationType.IDENTITY)

int blogCommentId;

int userId;

String username;

String userProfileId;

String titel;

int onOfLikes;

String blogComment;

LocalDate currentDate;

@ManyToOne

@JoinColumn(name="BlogId")

@JsonBackReference

Blog blog;

public int getBlogCommentId() {

return blogCommentId;

}

public void setBlogCommentId(int blogCommentId) {

this.blogCommentId = blogCommentId;

}

public int getUserId() {

return userId;

}

public void setUserId(int userId) {

this.userId = userId;

}

public String getUsername() {

return username;

}

public void setUsername(String username) {

this.username = username;

}

public String getUserProfileId() {

return userProfileId;

}

public void setUserProfileId(String userProfileId) {

this.userProfileId = userProfileId;

}

public String getTitel() {

return titel;

}

public void setTitel(String titel) {

this.titel = titel;

}

public int getOnOfLikes() {

return onOfLikes;

}

public void setOnOfLikes(int onOfLikes) {

this.onOfLikes = onOfLikes;

}

public static long getSerialversionuid() {

return serialVersionUID;

}

public String getBlogComment() {

return blogComment;

}

public void setBlogComment(String blogComment) {

this.blogComment = blogComment;

}

public LocalDate getCurrentDate() {

return currentDate;

}

public void setCurrentDate(LocalDate currentDate) {

this.currentDate = currentDate;

}

public Blog getBlog() {

return blog;

}

public void setBlog(Blog blog) {

this.blog = blog;

}

}

🡪Create **package** com.coll.OnlineCollaborate.dao;

🡪Create Interface :-

1**.IUserDao:-**

package com.coll.OnlineCollaborate.dao;

import java.util.List;

import com.coll.OnlineCollaborate.model.User;

public interface IUserDao {

List<User> userListbyStatus(String status);

List<User> getAllusers();

User getUserById(int userId);

User getUserByName(String username);

User validateUser(User user);

boolean addUser(User user);

boolean updateUser(User user);

boolean deleteUser(int userId);

boolean deactiveUser(int userId);

boolean activeUser(int userId);

boolean logoutUser(int userId);

List<User> getAllDeactiveUser();

boolean updateUserProfile(String file, Integer userId);

}

**2.IBlogDao:-**

package com.coll.OnlineCollaborate.dao;

import java.util.List;

import com.coll.OnlineCollaborate.model.Blog;

import com.coll.OnlineCollaborate.model.User;

public interface IBlogDao {

List<Blog> getAllBlogs();

List<Blog> getBlogsByStatus(String status);

List<Blog> getUsersBlogs(int id);

Blog getBlogById(int blogId);

boolean addBlog(Blog blog);

boolean updateBlog(Blog blog);

boolean activeBlog(int blogId);

List<Blog> getAllDeactiveBlog();

boolean deleteBlog(int blogId);

List<Blog> getBlogbyUserId(int userId);

}

**3.IBlogCommentsDao:-**

package com.coll.OnlineCollaborate.dao;

import java.util.List;

import com.coll.OnlineCollaborate.model.Blog;

import com.coll.OnlineCollaborate.model.BlogComments;

public interface IBlogCommentsDao {

List<BlogComments> getAllBlogComments();

List<BlogComments> getCommentsbyBlogId(int blogId);

BlogComments getBlogCommentsById(int blogCommentsId);

boolean addBlogComments(BlogComments blogComments);

boolean updateBlogComments(BlogComments blogComments);

boolean deleteBlogComments(int blogCommentsId);

}

🡪Create **package** com.coll.OnlineCollaborate.daoImpl;

🡪Create DaoImpl Class

**1.UserDaoImpl:-**

package com.coll.OnlineCollaborate.daoImpl;

import java.util.List;

import org.hibernate.SessionFactory;

import org.hibernate.query.Query;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Repository;

import org.springframework.transaction.annotation.Transactional;

import com.coll.OnlineCollaborate.dao.IUserDao;

import com.coll.OnlineCollaborate.model.User;

@Repository("userDao")

@Transactional

public class UserDaoImpl implements IUserDao {

@Autowired

SessionFactory sessionFactory;

@Override

public List<User> userListbyStatus(String status) {

String q="from User where status="+status+"'";

Query query=sessionFactory.getCurrentSession().createQuery(q);

return query.getResultList();

}

@Override

public List<User> getAllusers() {

return sessionFactory.getCurrentSession().createQuery("from User where enabled='true'",User.class).getResultList();

}

@Override

public User getUserById(int userId) {

return sessionFactory.getCurrentSession().get(User.class, Integer.valueOf(userId));

}

@Override

public User getUserByName(String username) {

String query="from User where username=:username";

return sessionFactory.getCurrentSession().createQuery(query,User.class).setParameter("username", username).getSingleResult();

}

@Override

public User validateUser(User user) {

String username=user.getUsername();

String password=user.getPassword();

String q="from User where username='"+username+"' and password='"+password+"' and enabled='true'";

Query query=sessionFactory.getCurrentSession().createQuery(q);

try {

user=(User)query.getSingleResult();

user.setIsOnline("true");

return user;

}

catch(Exception e) {

e.printStackTrace();

return null;

}

}

@Override

public boolean addUser(User user) {

try {

sessionFactory.getCurrentSession().save(user);

return true;

}

catch(Exception ex) {

ex.printStackTrace();

return false;

}

}

@Override

public boolean updateUser(User user) {

try {

sessionFactory.getCurrentSession().update(user);

return true;

}

catch(Exception ex) {

ex.printStackTrace();

return false;

}

}

@Override

public boolean deleteUser(int userId) {

try {

sessionFactory.getCurrentSession().delete(getUserById(userId));

return true;

}

catch(Exception ex) {

ex.printStackTrace();

return false;

}

}

@Override

public boolean deactiveUser(int userId) {

try {

User user=getUserById(userId);

user.setEnabled("false");

user.setStatus("Inactive");

sessionFactory.getCurrentSession().update(user);

return true;

}

catch(Exception ex) {

ex.printStackTrace();

return false;

}

}

@Override

public boolean updateUserProfile(String file, Integer userId) {

String q="update User set profile=:filename where userId=:id";

Query query=sessionFactory.getCurrentSession().createQuery(q);

query.setParameter("id",(Integer)userId);

query.setParameter("filename", file);

try {

query.executeUpdate();

return true;

}

catch(Exception e) {

e.printStackTrace();

return false;

}

}

@Override

public boolean activeUser(int userId) {

try {

User user=getUserById(userId);

user.setEnabled("true");

user.setStatus("Active");

sessionFactory.getCurrentSession().update(user);

return true;

}

catch(Exception ex) {

ex.printStackTrace();

return false;

}

}

@Override

public List<User> getAllDeactiveUser() {

return sessionFactory.getCurrentSession().createQuery("from User where enabled='false'",User.class).getResultList();

}

@Override

public boolean logoutUser(int userId) {

try {

User user=getUserById(userId);

user.setIsOnline("false");

sessionFactory.getCurrentSession().update(user);

return true;

}

catch(Exception ex) {

ex.printStackTrace();

return false;

}

}

}

**2.BlogDaoImpl:-**

package com.coll.OnlineCollaborate.daoImpl;

import java.time.LocalDate;

import java.util.List;

import org.hibernate.SessionFactory;

import org.hibernate.query.Query;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Repository;

import org.springframework.transaction.annotation.Transactional;

import com.coll.OnlineCollaborate.dao.IBlogDao;

import com.coll.OnlineCollaborate.model.Blog;

import com.coll.OnlineCollaborate.model.User;

@Repository("blogDao")

@Transactional

public class BlogDaoImpl implements IBlogDao{

@Autowired

SessionFactory sessionFactory;

@Override

public List<Blog> getAllBlogs() {

return sessionFactory.getCurrentSession().createQuery("from Blog where status='Posted'",Blog.class).getResultList();

}

@Override

public List<Blog> getBlogsByStatus(String status) {

String q="from Blog where status='"+status+"'";

Query query=sessionFactory.getCurrentSession().createQuery(q);

return query.getResultList();

}

@Override

public List<Blog> getUsersBlogs(int id) {

return sessionFactory.getCurrentSession().createQuery("from Blog",Blog.class).getResultList();

}

@Override

public Blog getBlogById(int blogId) {

return sessionFactory.getCurrentSession().get(Blog.class, Integer.valueOf(blogId));

}

@Override

public boolean addBlog(Blog blog) {

try {

System.out.println(blog.getBlogPosted());

sessionFactory.getCurrentSession().save(blog);

return true;

}

catch(Exception ex) {

ex.printStackTrace();

return false;

}

}

@Override

public boolean updateBlog(Blog blog) {

try {

sessionFactory.getCurrentSession().update(blog);

return true;

}

catch(Exception ex) {

ex.printStackTrace();

return false;

}

}

@Override

public boolean deleteBlog(int blogId) {

try {

System.out.println(blogId);

sessionFactory.getCurrentSession().delete(getBlogById(blogId));

return true;

}

catch(Exception ex) {

ex.printStackTrace();

return false;

}

}

@Override

public boolean activeBlog(int blogId) {

try {

Blog blog=getBlogById(blogId);

blog.setStatus("Posted");

sessionFactory.getCurrentSession().update(blog);

return true;

}

catch(Exception ex) {

ex.printStackTrace();

return false;

}

}

@Override

public List<Blog> getAllDeactiveBlog() {

return sessionFactory.getCurrentSession().createQuery("from Blog where status='Waiting For Apporved'",Blog.class).getResultList();

}

@Override

public List<Blog> getBlogbyUserId(int userId) {

return sessionFactory.getCurrentSession().createQuery("from Blog where userId='"+userId+"'",Blog.class).getResultList();

}

}

**3.BlogComentsDaoImpl:-**

package com.coll.OnlineCollaborate.daoImpl;

import java.util.List;

import org.hibernate.SessionFactory;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Repository;

import org.springframework.transaction.annotation.Transactional;

import com.coll.OnlineCollaborate.dao.IBlogCommentsDao;

import com.coll.OnlineCollaborate.model.Blog;

import com.coll.OnlineCollaborate.model.BlogComments;

@Repository("blogCommentsDao")

@Transactional

public class BlogCommentsDaoImpl implements IBlogCommentsDao {

@Autowired

SessionFactory sessionFactory;

@Override

public List<BlogComments> getAllBlogComments() {

return sessionFactory.getCurrentSession().createQuery("from BlogComments", BlogComments.class).getResultList();

}

@Override

public BlogComments getBlogCommentsById(int blogId) {

return sessionFactory.getCurrentSession().get(BlogComments.class, Integer.valueOf(blogId));

}

@Override

public boolean addBlogComments(BlogComments blogComments) {

try {

sessionFactory.getCurrentSession().save(blogComments);

return true;

}

catch(Exception ex) {

ex.printStackTrace();

return false;

}

}

@Override

public boolean updateBlogComments(BlogComments blogComments) {

try {

sessionFactory.getCurrentSession().update(blogComments);

return true;

}

catch(Exception ex) {

ex.printStackTrace();

return false;

}

}

@Override

public boolean deleteBlogComments(int blogCommentsId) {

try {

sessionFactory.getCurrentSession().delete(getBlogCommentsById(blogCommentsId));

return true;

}

catch(Exception ex) {

ex.printStackTrace();

return false;

}

}

@Override

public List<BlogComments> getCommentsbyBlogId(int blogId) {

return (List<BlogComments>) sessionFactory.getCurrentSession().get(BlogComments.class,Integer.valueOf(blogId));

}

}

🡪Create **package** com.coll.OnlineCollaborate.service;

🡪Create **Service Inteface**

**1.IUserService:**

package com.coll.OnlineCollaborate.service;

import java.util.List;

import com.coll.OnlineCollaborate.model.User;

public interface IUserService {

List<User> userListbyStatus(String status);

List<User> getAllusers();

User getUserById(int userId);

User getUserByName(String username);

User validateUser(User user);

boolean addUser(User user);

boolean updateUser(User user);

boolean deleteUser(int userId);

boolean deactiveUser(int userId);

boolean activeUser(int userId);

boolean logoutUser(int userId);

List<User> getAllDeactiveUser();

boolean updateUserProfile(String file, Integer userId);

}

**2.IBlogService:-**

package com.coll.OnlineCollaborate.service;

import java.util.List;

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

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

import com.coll.OnlineCollaborate.model.Blog;

public interface IBlogService {

List<Blog> getAllBlogs();

List<Blog> getBlogsByStatus(String status);

List<Blog> getUsersBlogs(int id);

Blog getBlogById(int blogId);

boolean addBlog(Blog blog);

boolean updateBlog(Blog blog);

boolean activeBlog(int blogId);

List<Blog> getAllDeactiveBlog();

boolean deleteBlog(int blogId);

List<Blog> getBlogbyUserId(int userId);

}

**3.IBlogCommentsService:-**

package com.coll.OnlineCollaborate.service;

import java.util.List;

import com.coll.OnlineCollaborate.model.BlogComments;

public interface IBlogCommentsService {

List<BlogComments> getAllBlogComments();

List<BlogComments> getCommentsbyBlogId(int blogId);

BlogComments getBlogCommentsById(int blogCommentsId);

boolean addBlogComments(BlogComments blogComments);

boolean updateBlogComments(BlogComments blogComments);

boolean deleteBlogComments(int blogCommentsId);

}

🡪Create **package** com.coll.OnlineCollaborate.serviceImpl;

🡪Create **ServiceImpl Class:-**

**1. UserServiceImpl**:-

package com.coll.OnlineCollaborate.serviceImpl;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

import com.coll.OnlineCollaborate.dao.IUserDao;

import com.coll.OnlineCollaborate.model.User;

import com.coll.OnlineCollaborate.service.IUserService;

@Service

@Transactional

public class UserServiceImpl implements IUserService {

@Autowired

IUserDao userDao;

@Override

public List<User> userListbyStatus(String status) {

// TODO Auto-generated method stub

return userDao.userListbyStatus(status);

}

@Override

public List<User> getAllusers() {

// TODO Auto-generated method stub

return userDao.getAllusers();

}

@Override

public User getUserById(int userId) {

// TODO Auto-generated method stub

return userDao.getUserById(userId);

}

@Override

public User getUserByName(String username) {

// TODO Auto-generated method stub

return userDao.getUserByName(username);

}

@Override

public User validateUser(User user) {

// TODO Auto-generated method stub

return userDao.validateUser(user);

}

@Override

public boolean addUser(User user) {

// TODO Auto-generated method stub

return userDao.addUser(user);

}

@Override

public boolean updateUser(User user) {

// TODO Auto-generated method stub

return userDao.updateUser(user);

}

@Override

public boolean deleteUser(int userId) {

// TODO Auto-generated method stub

return userDao.deleteUser(userId);

}

@Override

public boolean deactiveUser(int userId) {

// TODO Auto-generated method stub

return userDao.deactiveUser(userId);

}

@Override

public boolean updateUserProfile(String file, Integer userId) {

// TODO Auto-generated method stub

return userDao.updateUserProfile(file, userId);

}

@Override

public boolean activeUser(int userId) {

// TODO Auto-generated method stub

return userDao.activeUser(userId);

}

@Override

public List<User> getAllDeactiveUser() {

// TODO Auto-generated method stub

return userDao.getAllDeactiveUser();

}

@Override

public boolean logoutUser(int userId) {

// TODO Auto-generated method stub

return userDao.logoutUser(userId);

}

}

**2.BlogServiceImpl:-**

package com.coll.OnlineCollaborate.serviceImpl;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

import com.coll.OnlineCollaborate.dao.IBlogDao;

import com.coll.OnlineCollaborate.model.Blog;

import com.coll.OnlineCollaborate.service.IBlogService;

@Service

@Transactional

public class BlogServiceImpl implements IBlogService {

@Autowired

IBlogDao blogDao;

@Override

public List<Blog> getAllBlogs() {

return blogDao.getAllBlogs();

}

@Override

public List<Blog> getBlogsByStatus(String status) {

return blogDao.getBlogsByStatus(status);

}

@Override

public List<Blog> getUsersBlogs(int blogId) {

return blogDao.getUsersBlogs(blogId);

}

@Override

public Blog getBlogById(int blogId) {

return blogDao.getBlogById(blogId);

}

@Override

public boolean addBlog(Blog blog) {

return blogDao.addBlog(blog);

}

@Override

public boolean updateBlog(Blog blog) {

return blogDao.updateBlog(blog);

}

@Override

public boolean deleteBlog(int blogId) {

return blogDao.deleteBlog(blogId);

}

@Override

public boolean activeBlog(int blogId) {

// TODO Auto-generated method stub

return blogDao.activeBlog(blogId);

}

@Override

public List<Blog> getAllDeactiveBlog() {

// TODO Auto-generated method stub

return blogDao.getAllDeactiveBlog();

}

@Override

public List<Blog> getBlogbyUserId(int userId) {

// TODO Auto-generated method stub

return blogDao.getBlogbyUserId(userId);

}

}

**3.BlogCommentsServiceImpl:-**

package com.coll.OnlineCollaborate.serviceImpl;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

import com.coll.OnlineCollaborate.dao.IBlogCommentsDao;

import com.coll.OnlineCollaborate.model.BlogComments;

import com.coll.OnlineCollaborate.service.IBlogCommentsService;

@Service

@Transactional

public class BlogCommentsServiceImpl implements IBlogCommentsService {

@Autowired

IBlogCommentsDao blogCommentsDao;

@Override

public List<BlogComments> getAllBlogComments() {

return blogCommentsDao.getAllBlogComments();

}

@Override

public BlogComments getBlogCommentsById(int blogCommentsId) {

return blogCommentsDao.getBlogCommentsById(blogCommentsId);

}

@Override

public boolean addBlogComments(BlogComments blogComments) {

return blogCommentsDao.addBlogComments(blogComments);

}

@Override

public boolean updateBlogComments(BlogComments blogComments) {

return blogCommentsDao.updateBlogComments(blogComments);

}

@Override

public boolean deleteBlogComments(int blogCommentsId) {

return blogCommentsDao.deleteBlogComments(blogCommentsId);

}

@Override

public List<BlogComments> getCommentsbyBlogId(int blogId) {

return blogCommentsDao.getCommentsbyBlogId(blogId);

}

}

🡪Create **package** com.coll.OnlineCollaborate.controller;

🡪Create **Controllers:**

**1.UserController**

package com.coll.OnlineCollaborate.controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

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

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

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.coll.OnlineCollaborate.model.User;

import com.coll.OnlineCollaborate.service.IUserService;

@RestController

@CrossOrigin(origins="http://localhost:4200")

@RequestMapping(value="/api")

public class UserController {

@Autowired

IUserService userService;

@PostMapping("save-user")public boolean saveUser(@RequestBody User user) {

return userService.addUser(user);

}

@GetMapping("user-list")

public List<User> allUsers(){

return userService.getAllusers();

}

@DeleteMapping("delete-user/{userId}")

public boolean deleteUser(@PathVariable("userId") int userId) {

return userService.deleteUser(userId);

}

@GetMapping("user/{userId}")

public User userById(@PathVariable("userId") int userId) {

return userService.getUserById(userId);

}

@PostMapping("update-user/{userId}")

public boolean updateUser(@RequestBody User user, @PathVariable("userId") int userId) {

user.setUserId(userId);

return userService.updateUser(user);

}

@GetMapping("deactive-list")

public List<User> AllDeactiveUser(){

return userService.getAllDeactiveUser();

}

@PostMapping("active-user/{userId}")

public boolean activeUser(@RequestBody User user, @PathVariable("userId") int userId) {

return userService.activeUser(userId);

}

@PostMapping("validate-user")

public User validateUser(@RequestBody User user) {

return userService.validateUser(user);

}

@PostMapping("logout-user/{userId}")

public boolean logoutUser(@RequestBody User user, @PathVariable("userId") int userId) {

return userService.logoutUser(userId);

}

}

**2.BlogController:-**

package com.coll.OnlineCollaborate.controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

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

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

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.coll.OnlineCollaborate.model.Blog;

import com.coll.OnlineCollaborate.model.User;

import com.coll.OnlineCollaborate.service.IBlogService;

@RestController

@CrossOrigin(origins="http://localhost:4200")

@RequestMapping(value="/api")

public class BlogController {

@Autowired

IBlogService blogService;

@PostMapping("save-blog")public boolean saveBlog(@RequestBody Blog blog)

{

return blogService.addBlog(blog);

}

@GetMapping("blog-list")

public List<Blog> allBlog()

{

return blogService.getAllBlogs();

}

@DeleteMapping("delete-blog/{blogId}")

public boolean deleteBlog(@PathVariable("blogId") int blogId) {

return blogService.deleteBlog(blogId);

}

@GetMapping("blog/{blogId}")

public Blog getblogById(@PathVariable("blogId") int blogId)

{

return blogService.getBlogById(blogId);

}

@PostMapping("update-blog/{blogId}")

public boolean updateBlog(@RequestBody Blog blog, @PathVariable("blogId") int blogId)

{

blog.setBlogId(blogId);

return blogService.updateBlog(blog);

}

@GetMapping("deactiveBlog-list")

public List<Blog> AllDeactiveBlog(){

return blogService.getAllDeactiveBlog();

}

@PostMapping("active-blog/{blogId}")

public boolean activeBlog(@RequestBody Blog blog, @PathVariable("blogId") int blogId) {

return blogService.activeBlog(blogId);

}

@GetMapping("myblogs/{userId}")

public List<Blog> getBlogbyUserId(@PathVariable("userId") int userId){

return blogService.getBlogbyUserId(userId);

}

}

**3.BlogCommentsController:-**

package com.coll.OnlineCollaborate.controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

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

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

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.coll.OnlineCollaborate.model.Blog;

import com.coll.OnlineCollaborate.model.BlogComments;

import com.coll.OnlineCollaborate.service.IBlogCommentsService;

@RestController

@CrossOrigin(origins="http://localhost:4200")

@RequestMapping(value="/api")

public class BlogCommentsController {

@Autowired

IBlogCommentsService blogCommentsService;

@PostMapping("save-blogComments")

public boolean saveBlog(@RequestBody BlogComments blogComments)

{

return blogCommentsService.addBlogComments(blogComments);

}

@GetMapping("blogComments-list")

public List<BlogComments> allBlogComments()

{

return blogCommentsService.getAllBlogComments();

}

@DeleteMapping("delete-blogComments/{blogComments\_id}")

public boolean deleteBlogComments(@PathVariable("blogComments\_id") int blogComments\_id) {

return blogCommentsService.deleteBlogComments(blogComments\_id);

}

@GetMapping("blogComments/{blogComments\_id}")

public BlogComments blogCommentsById(@PathVariable("blogComments\_id") int blogComments\_id)

{

return blogCommentsService.getBlogCommentsById(blogComments\_id);

}

@PostMapping("update-blogComments/{blogComments}")

public boolean updateBlogComments(@PathVariable("blogComments") BlogComments blogComments)

{

return blogCommentsService.updateBlogComments(blogComments);

}

}

🡪 Open **OnlineCollaborateApplication**

package com.coll.OnlineCollaborate;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class OnlineCollaborateApplication {

public static void main(String[] args) {

SpringApplication.run(OnlineCollaborateApplication.class, args);

}

}

🡪 Right Click OnlineCollaborateApplication

🡪Click Run as

🡪 Click JavaApplication