**Get and Post Mapping Using the h2 Database**

Step 1:

First we create an spring boot application <https://start.spring.io/>

Add the following details:

Group : com.wave14.c2s1

Artifact: H2GetPost

Add dependencies:

**Dependencies**ADD DEPENDENCIES...CTRL + B

* **Spring Web WEB**Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.
* **Spring Data JPA SQL**Persist data in SQL stores with Java Persistence API using Spring Data and Hibernate.
* **H2 Database SQL**Provides a fast in-memory database that supports JDBC API and

Now import the project in intelij

Create a package controller

And create a class UserController.java in it

package com.wave14.c2s1.GetPostH2.controller;  
import com.wave14.c2s1.GetPostH2.service.UserService;  
import com.wave14.c2s1.GetPostH2.model.User;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.http.HttpStatus;  
import org.springframework.http.ResponseEntity;  
import org.springframework.web.bind.annotation.\*;  
import java.util.Map;  
  
@RestController  
@RequestMapping("/api/v1")  
public class UserController {  
 private UserService userService;  
  
  
  
 @Autowired  
 public UserController(UserService userService) {  
 this.userService = userService;  
  
 }  
  
 @PostMapping("/user")  
 public ResponseEntity<?> saveUser(@RequestBody User user){  
 return new ResponseEntity<>(userService.saveUser(user), HttpStatus.*CREATED*);  
 }  
  
 @GetMapping("/users")  
 public ResponseEntity<?> updateUser(@RequestBody User user) {  
 return new ResponseEntity<>(userService.getAllUsers(), HttpStatus.*OK*);  
 }  
  
}

Create a model package and create a file User.java in it

package com.wave14.c2s1.GetPostH2.model;  
import javax.persistence.Entity;  
import javax.persistence.Id;  
@Entity  
public class User {  
 @Id  
 private String email;  
 private String password;  
  
 public User(String email, String password) {  
 this.email = email;  
 this.password = password;  
 }  
  
 public User() {  
 }  
  
 public String getEmail() {  
 return email;  
 }  
  
 public void setEmail(String email) {  
 this.email = email;  
 }  
  
 public String getPassword() {  
 return password;  
 }  
  
 public void setPassword(String password) {  
 this.password = password;  
 }  
  
 @Override  
 public String toString() {  
 return "User{" +  
 "email='" + email + '\'' +  
 ", password='" + password + '\'' +  
 '}';  
 }  
  
}

Create a package repository and add an interface UserRepository

import com.wave14.c2s1.GetPostH2.model.User;  
import org.springframework.data.jpa.repository.JpaRepository;  
import org.springframework.data.repository.CrudRepository;  
import org.springframework.stereotype.Repository;  
  
@Repository  
public interface UserRepository extends CrudRepository<User,String> {  
  
  
}

add a package service and add an interface

Add an interface UserService

package com.wave14.c2s1.GetPostH2.service;  
import com.wave14.c2s1.GetPostH2.model.User;  
  
import java.util.List;  
  
public interface UserService {  
  
 User saveUser(User user) ;  
  
 List<User> getAllUsers();  
}

Add a class UserServiceImpl

package com.wave14.c2s1.GetPostH2.service;  
import com.wave14.c2s1.GetPostH2.repository.UserRepository;  
import com.wave14.c2s1.GetPostH2.model.User;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.stereotype.Service;  
  
import java.util.List;  
  
@Service  
public class UserServiceImpl implements UserService {  
 private UserRepository userRepository;  
  
 @Autowired  
 public UserServiceImpl(UserRepository userRepository) {  
 this.userRepository = userRepository;  
 }  
  
 @Override  
 public User saveUser(User user) {  
  
 return userRepository.save(user);  
 }  
  
 @Override  
 public List<User> getAllUsers() {  
 return (List<User>) userRepository.findAll();  
 }  
}

in main file

@SpringBootApplication  
@EnableAutoConfiguration  
public class GetPostH2Application {  
  
 public static void main(String[] args) {  
  
 SpringApplication.*run*(GetPostH2Application.class, args);  
 }  
  
}

And add the following entry to application.properties

spring.h2.console.enabled=true