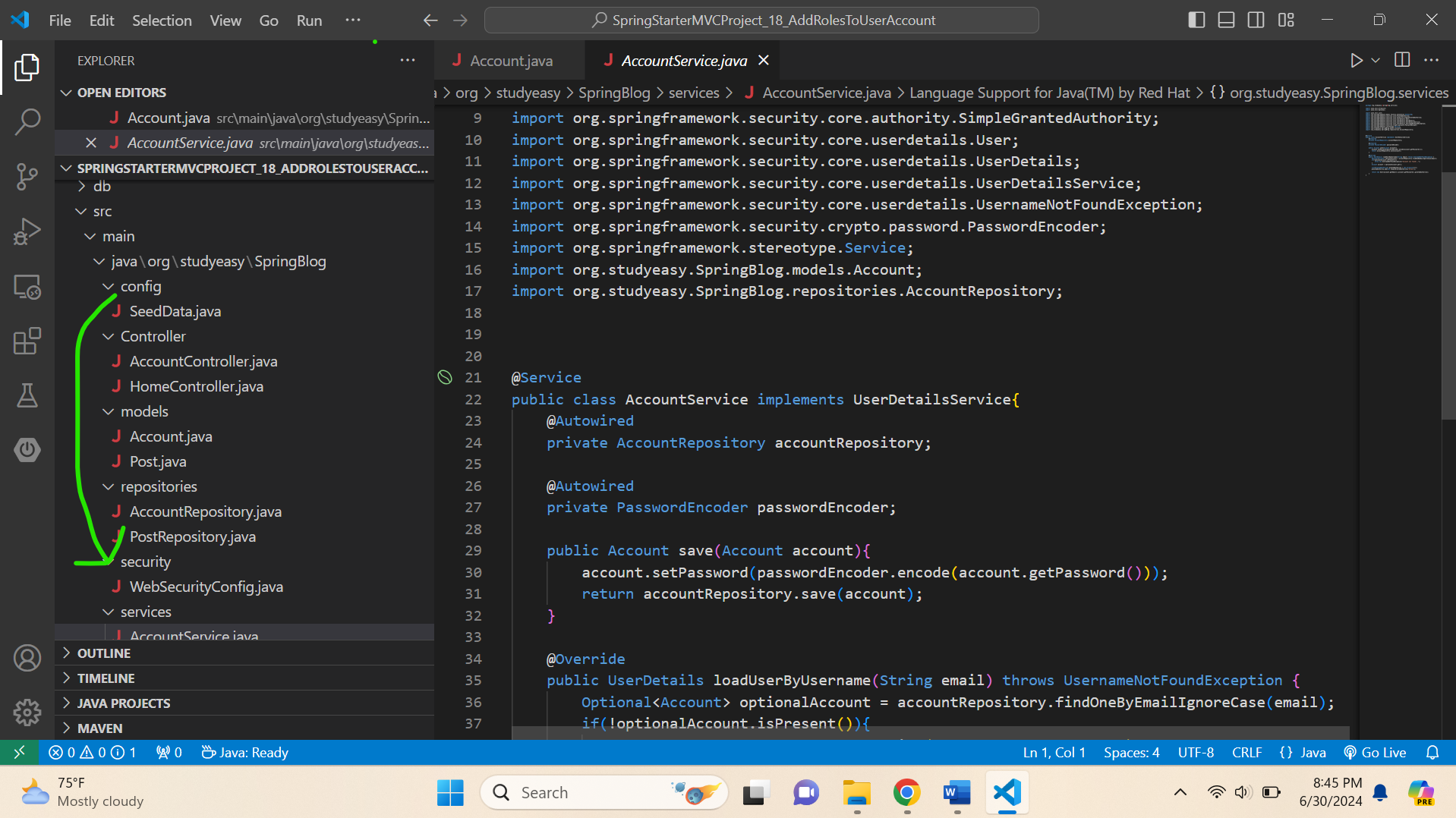
**SpringBoot – Add Roles To User Account**

**Change SpringStarter to SpringBlog**

****

**Move WebConfig.java from config folder to security folder.**

**Create new folder -> utils/constant -> Then new file ->**

**1. Roles.java**

**2. Authorities.java**

**Make it enum.**

**Roles.java**

package org.studyeasy.SpringBlog.utils.constant;

public enum Roles {

    USER("ROLE\_USER"),ADMIN("ROLE\_ADMIN"),EDITOR("ROLE\_EDITOR");

    private String role;

    private Roles(String role) {

        this.role = role;

    }

    public String getRole() {

        return role;

    }

}

**In AccountService.java**

 public Account save(Account account){

        account.setPassword(passwordEncoder.encode(account.getPassword()));

        account.setRole(Roles.USER.getRole());

        return accountRepository.save(account);

    }

**Add this line ->**

account.setRole(Roles.USER.getRole());

**Final Code of AccountService.java**

package org.studyeasy.SpringBlog.services;

import java.util.ArrayList;

import java.util.Optional;

import java.util.List;

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

import org.springframework.security.core.GrantedAuthority;

import org.springframework.security.core.authority.SimpleGrantedAuthority;

import org.springframework.security.core.userdetails.User;

import org.springframework.security.core.userdetails.UserDetails;

import org.springframework.security.core.userdetails.UserDetailsService;

import org.springframework.security.core.userdetails.UsernameNotFoundException;

import org.springframework.security.crypto.password.PasswordEncoder;

import org.springframework.stereotype.Service;

import org.studyeasy.SpringBlog.models.Account;

import org.studyeasy.SpringBlog.repositories.AccountRepository;

import org.studyeasy.SpringBlog.utils.constant.Roles;

@Service

public class AccountService implements UserDetailsService{

    @Autowired

    private AccountRepository accountRepository;

    @Autowired

    private PasswordEncoder passwordEncoder;

    public Account save(Account account){

        account.setPassword(passwordEncoder.encode(account.getPassword()));

        account.setRole(Roles.USER.getRole());

        return accountRepository.save(account);

    }

    @Override

    public UserDetails loadUserByUsername(String email) throws UsernameNotFoundException {

        Optional<Account> optionalAccount = accountRepository.findOneByEmailIgnoreCase(email);

        if(!optionalAccount.isPresent()){

            throw new UsernameNotFoundException("Account not found!..");

        }

        Account account = optionalAccount.get();

        List<GrantedAuthority> grantedAuthority = new ArrayList<>();

        grantedAuthority.add(new SimpleGrantedAuthority(account.getRole()));

        return new User(account.getEmail(),account.getPassword(),grantedAuthority);

    }

}

**Final Code of Account.java**

package org.studyeasy.SpringBlog.models;

import java.util.List;

import jakarta.persistence.Entity;

import jakarta.persistence.GeneratedValue;

import jakarta.persistence.GenerationType;

import jakarta.persistence.Id;

import jakarta.persistence.OneToMany;

import lombok.Getter;

import lombok.NoArgsConstructor;

import lombok.Setter;

@Entity

@Getter

@Setter

@NoArgsConstructor

public class Account {

    @Id

    @GeneratedValue(strategy = GenerationType.SEQUENCE)

    private Long id;

    private String email;

    private String password;

    private String firstname;

    private String lastname;

    private String role;

    @OneToMany(mappedBy = "account")

    private List<Post> posts;

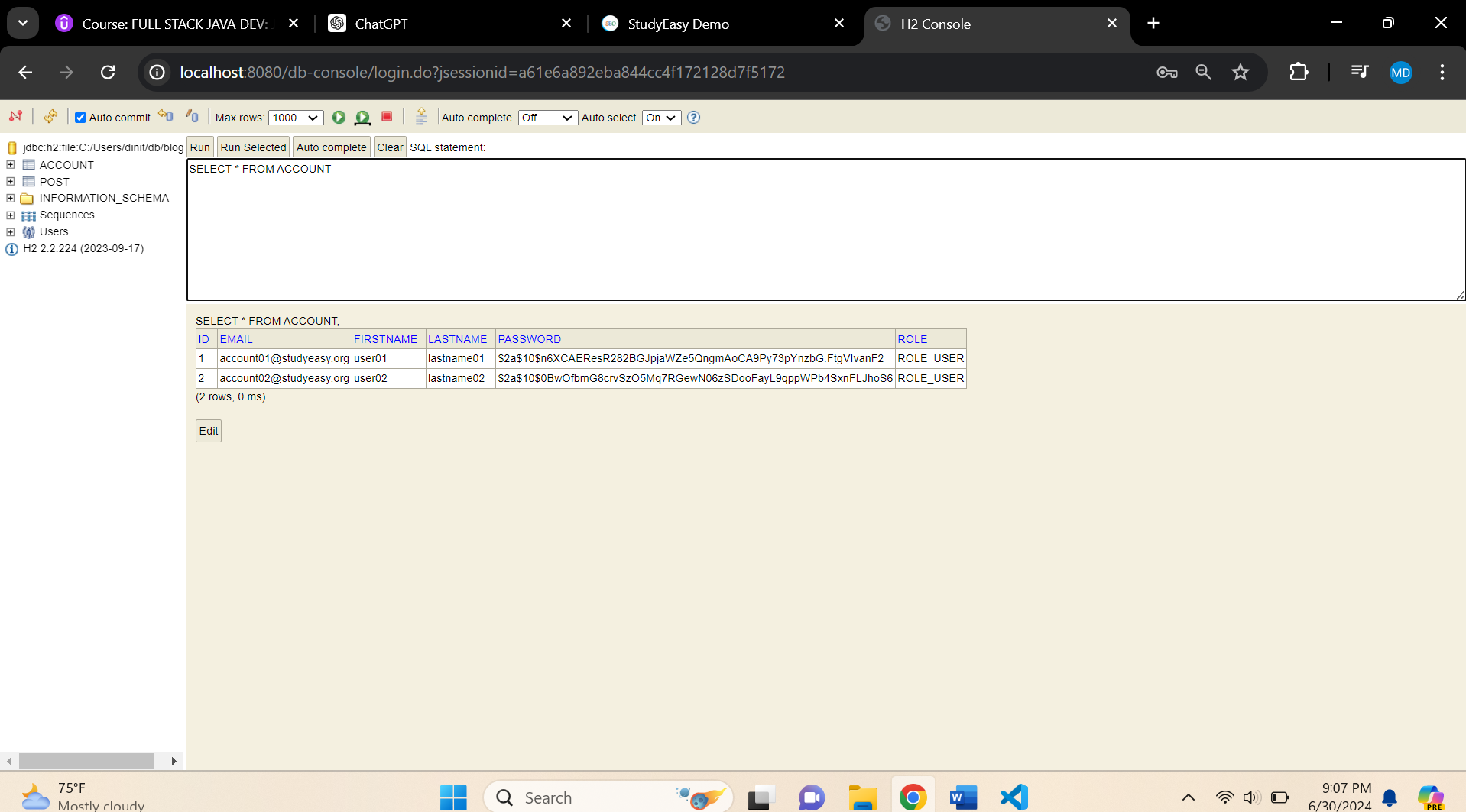
}

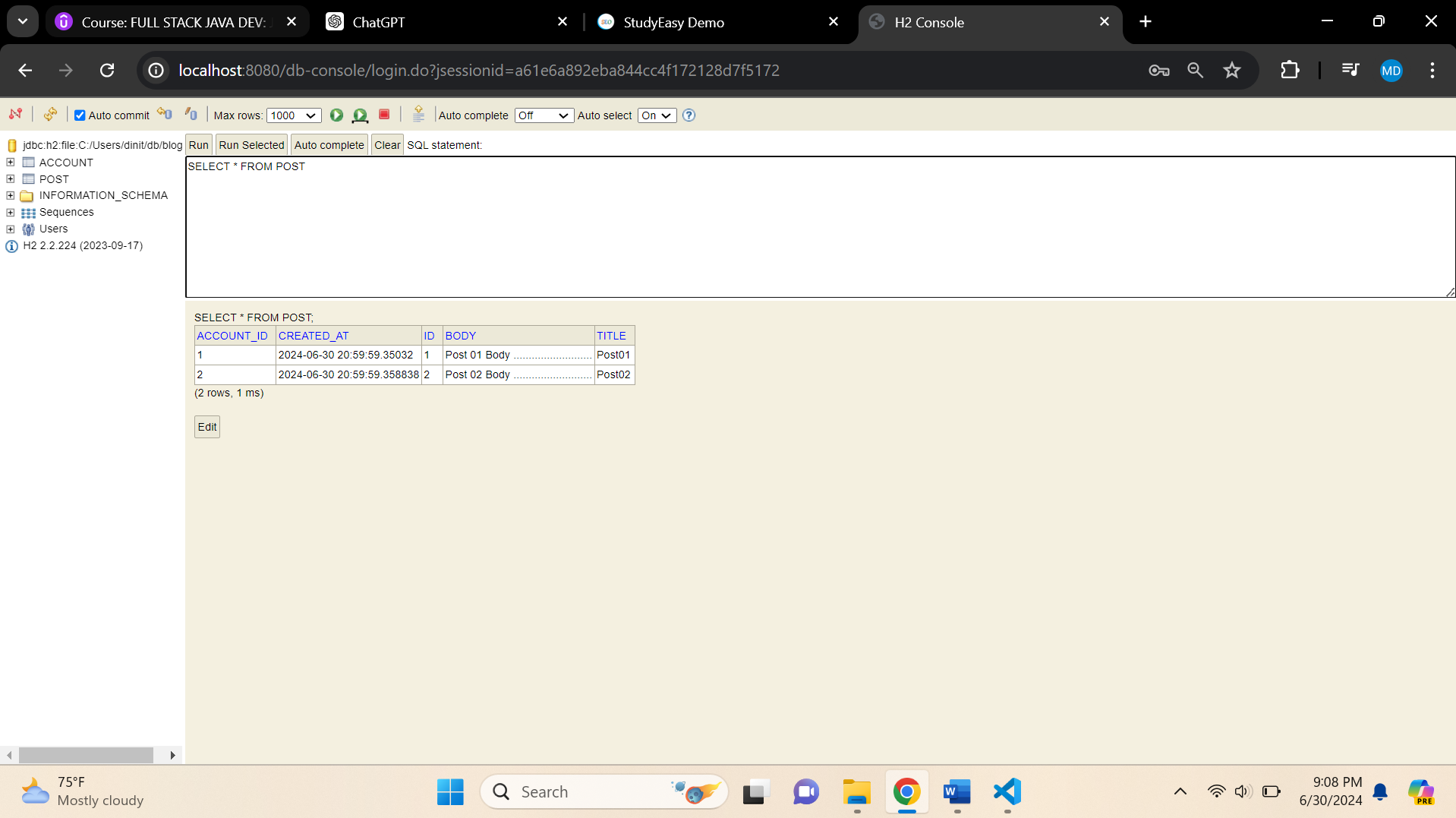
**Modify Seed Data.java -> Add below line**

account01.setLastname("lastname01");

account02.setLastname("lastname02");

**Output:-**

****

****