**AuthController – AddUserValidations**

Rename Account Controller to Home Controller

Please add **@Valid** for Validations

public ResponseEntity<TokenDTO> token(@Valid @RequestBody UserLoginDTO userLogin) throws AuthenticationException

public ResponseEntity<String> addUser(@Valid @RequestBody AccountDTO accountDTO)

**Making Changes in DTO – AccountDTO.java and UserLoginDTO.java**

**AccountDTO.java**

package org.studyeasy.SpringRestdemo.payload.auth;

import io.swagger.v3.oas.annotations.media.Schema;

import io.swagger.v3.oas.annotations.media.Schema.RequiredMode;

import jakarta.validation.constraints.Email;

import jakarta.validation.constraints.Size;

import lombok.Getter;

import lombok.Setter;

@Setter

@Getter

public class AccountDTO {

    @Email

    @Schema(description = "Email Address", example = "admin@studyeasy.org", requiredMode = RequiredMode.REQUIRED)

    private String email;

    @Size(min = 6, max = 20)

    @Schema(description = "Password", example = "Password", requiredMode = RequiredMode.REQUIRED, maxLength = 20, minLength = 6)

    private String password;

}

**UserLoginDTO.java**

package org.studyeasy.SpringRestdemo.payload.auth;

import io.swagger.v3.oas.annotations.media.Schema;

import io.swagger.v3.oas.annotations.media.Schema.RequiredMode;

import jakarta.validation.constraints.Email;

import jakarta.validation.constraints.Size;

import lombok.Getter;

import lombok.Setter;

@Getter

@Setter

public class UserLoginDTO {

    @Email

    @Schema(description = "Email Address", example = "admin@studyeasy.org", requiredMode = RequiredMode.REQUIRED)

    private String email;

    @Size(min = 6, max = 20)

    @Schema(description = "Password", example = "Password", requiredMode = RequiredMode.REQUIRED, maxLength = 20, minLength = 6)

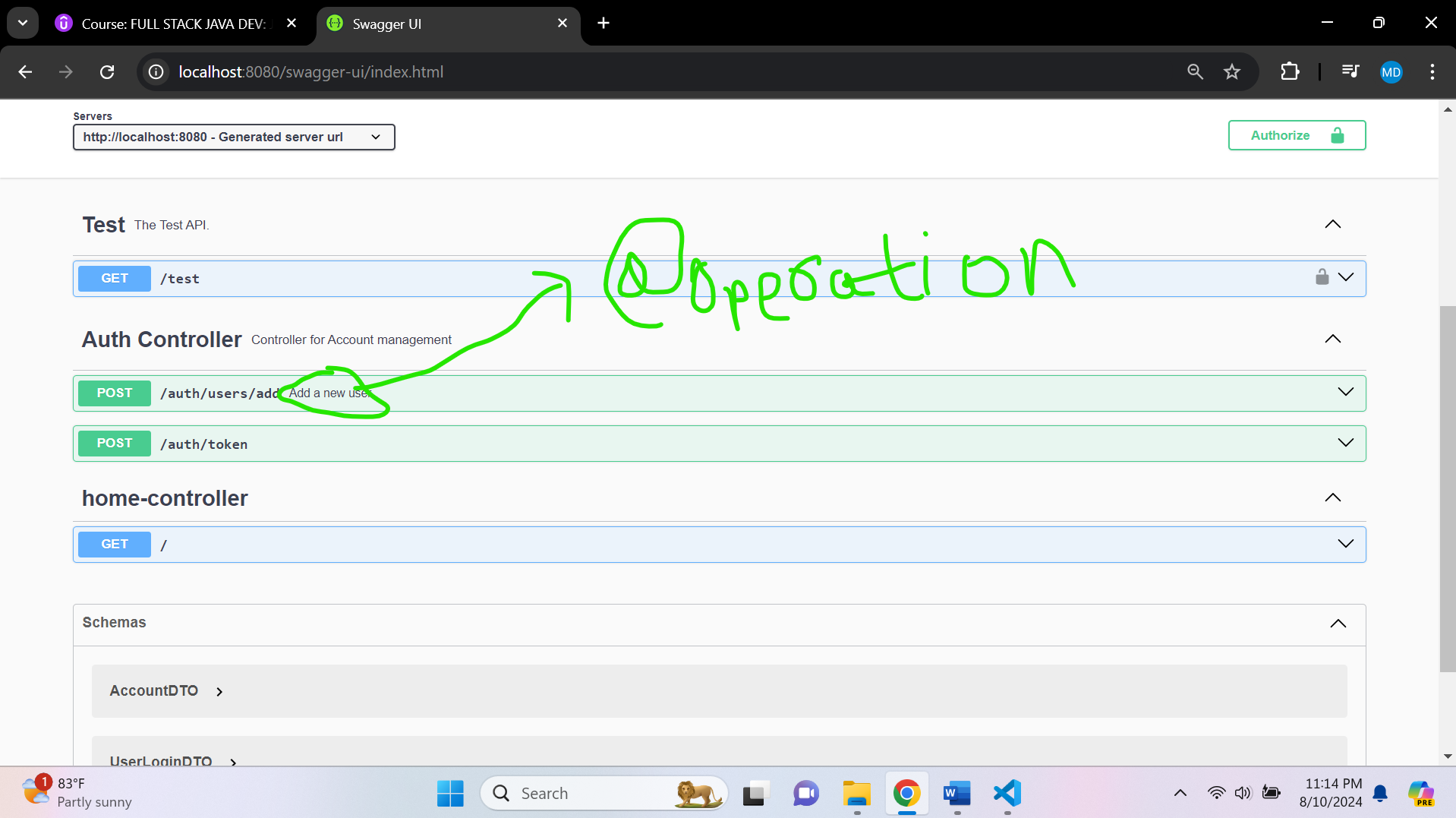
    private String password;

}

**@Operation in Controller**

 @Operation(summary = "Add a new user")

**Output**

****

**Update addUser method in AuthController**

  @PostMapping(value = "/users/add", consumes = "application/json", produces = "application/json")

    @ResponseStatus(HttpStatus.CREATED)

    @ApiResponse(responseCode = "400", description = "Please enter a valid email and Password length between 6 to 20 characters")

    @ApiResponse(responseCode = "201", description = "Account Added")

    @Operation(summary = "Add a new user")

    public ResponseEntity<String> addUser(@Valid @RequestBody AccountDTO accountDTO){

        try {

            Account account = new Account();

            account.setEmail(accountDTO.getEmail());

            account.setPassword(accountDTO.getPassword());

            account.setRole("ROLE\_USER");

            accountService.save(account);

            return ResponseEntity.ok(AccountSuccess.ACCOUNT\_ADDED.toString());

        } catch (Exception e) {

            log.debug(AccountError.ADD\_ACCOUNT\_ERROR.toString() + ": "+e.getMessage());

            return ResponseEntity.status(HttpStatus.BAD\_REQUEST).body(null);

        }

    }