Implement Image Upload Functionality using Multipart File

1. Adding dependencies related Image Upload functionality within pom.xml file of Backend project as following

<!-- SPRING WEB -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-web</artifactId>

<version>${spring.version}</version>

</dependency>

<!-- COMMONS FILEUPLOAD -->

<dependency>

<groupId>commons-fileupload</groupId>

<artifactId>commons-fileupload</artifactId>

<version>${commons-fileupload.version}</version>

</dependency>

<!-- COMMONS IO -->

<dependency>

<groupId>commons-io</groupId>

<artifactId>commons-io</artifactId>

<version>${commons-io.version}</version>

</dependency>

1. Save the project.
2. Open productCRUD.jsp page replace the existing code with following code:

<div class=*"row"*>

<div class=*"form-group col-md-12"*>

<label class=*"col-md-4 control-lable"* for=*"imageUrl"*>Upload Product Image</label>

<div class=*"col-md-7"*>

<form:input type=*"file"* path=*"imageUrl"* id=*"imageUrl"*

class=*"form-control input-sm"* />

<div class=*"has-error"*>

<form:errors path=*"imageUrl"* class=*"help-inline"* style=*"color:red"*/>

</div>

</div>

</div>

</div>

1. Open Product.java model and add following code:

@Transient

@JsonIgnore

MultipartFile file;

**public** MultipartFile getFile() {

**return** file;

}

**public** **void** setFile(MultipartFile file) {

**this**.file = file;

}

1. Open FrontEndProductController.java and create object of HttpRequest as following:

@Autowired

HttpRequest request;

1. Add following code inside the FrontEndProductController of the same class:

**protected** String uploadImage(MultipartFile multipart){

System.***out***.println("Method reached");

String folderToUpload="/resources/images/";

String fileName=multipart.getOriginalFilename();

String realPath=request.getServletContext().getRealPath(folderToUpload);

**if**(!**new** File(realPath).exists()){

**new** File(realPath).mkdirs();

}

String filePath=realPath+fileName;

File destination=**new** File(filePath);

**try**{

multipart.transferTo(destination);

}

**catch**(Exception e){

}

**return** fileName;

}

1. Add following two line inside the add case of doAction()

**if**(!(product.getFile().getOriginalFilename().equals(""))){

product.setImageUrl(uploadImage(product.getFile()));

}

1. Remove @NotBlank annotation from product.java from the field imageUrl.