Hibernate Validation

1. Open pom.xml of Front end application
2. Add version inside the <properties> tag

<hibernate.validator.version>5.1.2.Final</hibernate.validator.version>

1. Add following code to add hibernate validator dependency

<dependency>

          <groupId>org.hibernate</groupId>

             <artifactId>hibernate-validator</artifactId>

             <version>${hibernate.validator.version}</version>

        </dependency>

1. Open Product.java class and annotate with the validation properties.

**import** javax.validation.constraints.Size;

**import** org.hibernate.validator.constraints.NotBlank;

**import** org.hibernate.validator.constraints.NotEmpty;

**import** org.hibernate.validator.constraints.Range;

@Entity

**public** **class** Product {

@Id

@GeneratedValue(strategy=GenerationType.***IDENTITY***)//for autonumber

**int** productId;

@NotBlank(message="Can not leave this field witout entering any Product Name")

@Size(min=3,message="Product Name must be greater than 3 character")

String productName;

@NotEmpty(message="Can not leave this field witout entering any Product Description")

@Size(min=10, message="Product Name must be greater than 10 characters")

String productDescription;

@Range(min=1,message="Product Price value must be greater than 1")

**int** productPrice;

@Range(min=0, max=200, message="Product Quantity must be within 0 and 200")

**int** productQuantity;

@NotBlank(message="Product Image must be selected")

String imageUrl;

1. Add the following code with all input fields:

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

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

</div>

1. Open FrontEndProductController and update doAction() as following:

@RequestMapping(value="/productData.do", method=RequestMethod.***POST***)

**private** ModelAndView doActions(@ModelAttribute("product") @Valid Product product,BindingResult result, @RequestParam String action) {

Product productResult=**new** Product();

**if**(result.hasErrors()) {

ModelAndView model1 =**new** ModelAndView("page");

model1.addObject("title","Product Management");

model1.addObject("userClickProductCRUD","true");

**return** model1;

}

**switch**(action.toLowerCase()){

**case** "add":

productDAO.addProducts(product);

productResult=product;

**break**;

**case** "update":

productDAO.updateProducts(product);

productResult=product;

**break**;

**case** "delete":

productDAO.deleteProducts(product.getProductId());

productResult=product;

**break**;

**case** "search":

Product searchedProduct=productDAO.getProduct(product.getProductId());

productResult=searchedProduct!=**null**?searchedProduct:**new** Product();

**break**;

}

ModelAndView model1 =**new** ModelAndView("page");

model1.addObject("title","Product Management");

model1.addObject("userClickProductCRUD","true");

model1.addObject("product", productResult);

model1.addObject("productList",productDAO.getAllProducts());

**return** model1;

}