FarmFresh/pom.xml

1 <?xml version="1.0" encoding="UTF-8"?>

2 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

3 xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">

4 <modelVersion>4.0.0</modelVersion>

5 <parent>

6 <groupId>org.springframework.boot</groupId>

7 <artifactId>spring-boot-starter-parent</artifactId>

8 <version>2.1.8.RELEASE</version>

9 <relativePath/> <!-- lookup parent from repository -->

10 </parent>

11 <groupId>com.example</groupId>

12 <artifactId>FarmFresh</artifactId>

13 <version>0.0.1-SNAPSHOT</version>

14 <name>FarmFresh</name>

15 <description>Demo project for Spring Boot</description>

16 <properties>

17 <java.version>1.8</java.version>

18 </properties>

19

20 <dependencies>

21 <dependency>

22 <groupId>org.springframework.boot</groupId>

23 <artifactId>spring-boot-starter-web</artifactId>

24 </dependency>

25

26 <dependency>

27 <groupId>org.apache.tomcat.embed</groupId>

28 <artifactId>tomcat-embed-jasper</artifactId>

29 <scope>provided</scope>

30 </dependency>

31

32 <dependency>

33 <groupId>javax.servlet</groupId>

34 <artifactId>servlet-api</artifactId>

35 <version>2.5</version>

36 <scope>provided</scope>

37 </dependency>

38 <dependency>

39 <groupId>javax.servlet.jsp</groupId>

40 <artifactId>jsp-api</artifactId>

41 <version>2.1</version>

42 <scope>provided</scope>

43 </dependency>

44

45 <dependency>

46 <groupId>taglibs</groupId>

47 <artifactId>standard</artifactId>

48 <version>1.1.2</version>

49 </dependency>

50 <dependency>

51 <groupId>javax.servlet</groupId>

52 <artifactId>jstl</artifactId>

53 <version>1.2</version>

54 </dependency>

55 <!-- Spring test dependencies -->

56 <dependency>

57 <groupId>org.mockito</groupId>

58 <artifactId>mockito-all</artifactId>

59 <version>1.10.19</version>

60 <scope>test</scope>

61 </dependency>

62

63 <!-- <dependency>

64 <groupId>org.seleniumhq.selenium</groupId>

65 <artifactId>selenium-java</artifactId>

66 <version>2.53.0</version>

67 </dependency>-->

68

69 <dependency>

70 <groupId>org.seleniumhq.selenium</groupId>

71 <artifactId>htmlunit-driver</artifactId>

72 <version>2.26</version>

73 </dependency>

74

75 <!-- https://mvnrepository.com/artifact/org.w3c.css/sac -->

76 <dependency>

77 <groupId>org.w3c.css</groupId>

78 <artifactId>sac</artifactId>

79 <version>1.3</version>

80 </dependency>

81

82

83 <dependency>

84 <groupId>org.springframework</groupId>

85 <artifactId>spring-test</artifactId>

86 <version>5.1.7.RELEASE</version> <!-- 4.0.5 -->

87 <scope>test</scope>

88 </dependency>

89 <!-- End of spring test dependencies -->

90

91

92

93 <dependency>

94 <groupId>org.springframework.boot</groupId>

95 <artifactId>spring-boot-starter-test</artifactId>

96 <scope>test</scope>

97 <exclusions>

98 <exclusion>

99 <groupId>org.junit.vintage</groupId>

100 <artifactId>junit-vintage-engine</artifactId>

101 </exclusion>

102 </exclusions>

103 </dependency>

104 </dependencies>

105

106 <build>

107 <plugins>

108 <plugin>

109 <groupId>org.springframework.boot</groupId>

110 <artifactId>spring-boot-maven-plugin</artifactId>

111 </plugin>

112 </plugins>

113 </build>

114

115 </project>

116

FarmFresh/src/main/java/com/controller/ShoppingController.java

1 *package* com.controller;

2

3

4 *import* java.util.ArrayList;

5 *import* java.util.List;

6 *import* javax.validation.Valid;

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

8 *import* org.springframework.stereotype.Controller;

9 *import* org.springframework.ui.ModelMap;

10 *import* org.springframework.validation.BindingResult;

11

12 *import* com.exception.NoStockException;

13 *import* com.model.Product;

14 *import* com.service.ShoppingService;

15 *import* org.springframework.web.bind.annotation.ExceptionHandler;

16 *import* org.springframework.web.bind.annotation.ModelAttribute;

17 *import* org.springframework.web.bind.annotation.RequestMapping;

18 *import* org.springframework.web.bind.annotation.RequestMethod;

19 *import* org.springframework.web.servlet.ModelAndView;

20

21 //use appropriate annotation to configure ShoppingController as Controller

22 @Controller

23 *public* *class* ShoppingController {

24

25 @Autowired

26 *private* ShoppingService shoppingService;

27

28 @RequestMapping(value = "/showPage", method = RequestMethod.GET)

29 *public* String showPage(@ModelAttribute("product")Product product) {

30

31 *return* "showPage";

32 }

33

34 @ModelAttribute("productNames")

35 *public* List<String> populateProduct(){

36 List<String> productNames = *new* ArrayList<String>();

37 productNames.add("Apple");

38 productNames.add("Orange");

39 productNames.add("Grapes");

40 productNames.add("Mango");

41 productNames.add("JackFruit");

42 *return* productNames;

43 }

44

45 //invoke the service class - calculateCostAndUpdate method.

46 @RequestMapping(value = "/calculate", method = RequestMethod.POST)

47 *public* String calculateCost(@ModelAttribute("product")@Valid Product product, BindingResult result,ModelMap model) throws NoStockException {

48

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

50

51 *return* ("showPage");

52 }

53 *else*{

54 *double* cost=shoppingService.calculateCost(product);

55 model.addAttribute("cost",cost);

56 model.addAttribute("costPerKg",product.getCostPerKg());

57 model.addAttribute("name",product.getProductName());

58 model.addAttribute("quantity",product.getQuantity());

59

60 }

61

62 *return* ("billDesk");

63 }

64

65

66 @ExceptionHandler(NoStockException. *class*)

67 *public* ModelAndView exceptionHandler(Exception e) {

68

69 ModelAndView mav = *new* ModelAndView();

70 mav.addObject("message", e.getMessage());

71 mav.setViewName("exceptionPage");

72 *return* mav;

73 }

74

75 }

76

77

FarmFresh/src/main/java/com/example/demo/FarmFreshApplication.java

1 *package* com.example.demo;

2

3 *import* org.springframework.boot.SpringApplication;

4 *import* org.springframework.boot.autoconfigure.SpringBootApplication;

5 *import* org.springframework.context.annotation.ComponentScan;

6

7 @SpringBootApplication

8 @ComponentScan({"com.\*"})

9 *public* *class* FarmFreshApplication {

10

11 *public* *static* *void* main(String[] args) {

12 SpringApplication.run(FarmFreshApplication.*class*, args);

13 }

14

15 }

16

FarmFresh/src/main/java/com/exception/NoStockException.java

1 *package* com.exception;

2

3 *public* *class* NoStockException *extends* Exception{

4 *private* *static* *final* *long* serialVersionUID = 1L;

5 *public* NoStockException(String msg) {

6

7 *super*(msg);

8 }

9

10 }

11

FarmFresh/src/main/java/com/model/Product.java

1 *package* com.model;

2

3 *import* javax.validation.constraints.Min;

4 *import* javax.validation.constraints.NotEmpty;

5

6 *import* org.springframework.stereotype.Component;

7

8 @Component

9 *public* *class* Product {

10 *private* String productName ;

11 @Min(value=1,message="Minimum quantity should be 1Kg")

12 *private* *int* quantity;

13 *private* *double* costPerKg;

14

15

16 *public* String getProductName() {

17 *return* productName;

18 }

19

20 *public* *void* setProductName(String productName) {

21 *this*.productName = productName;

22 }

23

24

25 *public* *int* getQuantity() {

26 *return* quantity;

27 }

28

29

30 *public* *void* setQuantity(*int* quantity) {

31 *this*.quantity = quantity;

32 }

33

34

35 *public* *double* getCostPerKg() {

36 *return* costPerKg;

37 }

38

39

40 *public* *void* setCostPerKg(*double* costPerKg) {

41 *this*.costPerKg = costPerKg;

42 }

43

44

45 *public* Product() {}

46 }

47

FarmFresh/src/main/java/com/service/ShoppingService.java

1 *package* com.service;

2

3 *import* java.util.HashMap;

4 *import* java.util.Map;

5 *import* java.util.Map.Entry;

6

7 *import* org.springframework.stereotype.Service;

8

9 *import* com.exception.NoStockException;

10 *import* com.model.Product;

11

12 //use appropriate annotation to configure ElectricityService as a Service

13 @Service

14 *public* *class* ShoppingService {

15

16 *public* *double* calculateCost(Product product) throws NoStockException {

17 *double* totalCost=0.0;

18

19 Map<String,Integer> stock=getProductStock();

20 *for* (Entry<String, Integer> entry : stock.entrySet()) {

21 *if*(product.getProductName().equalsIgnoreCase(entry.getKey())) {

22

23 *if*(product.getQuantity()>entry.getValue()) {

24 *throw* *new* NoStockException("No enough stock for product "+product.getProductName());

25 }

26 }

27 }

28

29 *if*(product.getProductName().equals("Apple")) {

30 product.setCostPerKg(250);

31 }

32 *else* *if*(product.getProductName().equals("JackFruit")) {

33 product.setCostPerKg(75);

34 }

35 *else* *if*(product.getProductName().equals("Orange")) {

36 product.setCostPerKg(90);

37 }

38 *else* *if*(product.getProductName().equals("Mango")) {

39 product.setCostPerKg(60);

40 }

41 *else* *if*(product.getProductName().equals("Grapes")) {

42 product.setCostPerKg(150);

43 }

44

45 totalCost=product.getQuantity()\*product.getCostPerKg();

46

47 *return* totalCost;

48 }

49

50 *public* Map<String,Integer> getProductStock(){

51 Map<String,Integer> stock = *new* HashMap<String,Integer>();

52 stock.put("Apple",50);

53 stock.put("Grapes",10);

54 stock.put("Orange",30);

55 stock.put("Mango",75);

56 stock.put("JackFruit",25);

57 *return* stock;

58 }

59 }

60

FarmFresh/src/main/resources/application.properties

1 server.port=9088

2 spring.mvc.view.prefix = /WEB-INF/views/

3 spring.mvc.view.suffix = .jsp

4 spring.mvc.static-path-pattern=/resources/\*\*

FarmFresh/src/main/webapp/WEB-INF/views/billDesk.jsp

1 <%@page isELIgnored="false" %>

2 <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

3 <%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>

4 <html>

5 <body bgcolor="lavender"> <center>

6 <h2>Farm Fresh - A natural farm store!!</h2>

7

8 <table >

9 <tr>

10 <td>Product Name <td>${name}</td>

11 </tr>

12 <tr>

13 <td>Quantity <td>${quantity}</td>

14 </tr>

15 <tr>

16 <td>Cost Per Kg <td>${costPerKg}</td>

17 </tr>

18 <tr>

19 <td>Total cost Rs <td>${cost}</td>

20 </tr>

21 </table>

22 </center>

23

24 </body>

25 </html>

FarmFresh/src/main/webapp/WEB-INF/views/exceptionPage.jsp

1 <%@page isELIgnored="false" %>

2 <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

3 <%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>

4 <html>

5 <body bgcolor="lavender">

6

7 <center> <h2>Farm Fresh - A natural farm store!!</h2>

8 <h3>

9 <%

10 if (request.getAttribute("message") != null) {

11 %>

12 <%=request.getAttribute("message")%>

13 <%

14 }

15 %>

16 </h3>

17 <div>

18 <center> <a href="showPage"> Home</a></center>

19 </div>

20 </center>

21

22 </body>

23 </html>

FarmFresh/src/main/webapp/WEB-INF/views/showPage.jsp

1 <%@ taglib prefix="sf" uri="http://www.springframework.org/tags/form" %>

2 <%@ taglib uri="http://www.springframework.org/tags/form" prefix="form"%>

3 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"

4 pageEncoding="ISO-8859-1" isELIgnored="false"%>

5

6

7 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

8 <html>

9

10 <body style="background-color:lavender">

11

12 <!-- Design the page as per the requirements-->

13

14 <form:form method="post" action="calculate" modelAttribute="product">

15 <center><h2>Farm Fresh - A natural farm store!!</h2></center>

16 <table style="margin: 0px auto; margin-left: auto; margin-right:auto">

17 <tr>

18 <td>Select product</td>

19 <td>

20 <form:select path="productName" id="productName">

21 <form:options items="${productNames}"/>

22 </form:select>

23 </td>

24 </tr>

25 <tr>

26 <td>Quantity</td>

27 <td><form:input path="quantity" id="quantity"/></td>

28 <td><form:errors path="quantity"/></td>

29 </tr>

30

31

32 </table>

33

34 <center><input type="submit" value="Calculate Cost" id="submit" name="submit"/></center>

35

36 <br>

37

38 </form:form>

39

40

41 </body>

42 </html>

43