Exception Handling:-

Spring mvc provides 3 ways achieve exception handling

1. Controller Based Exception

2. Global Exception Handler

3. Handler Exception Resolver

1. Controller Based Exception:-

We can define exception handler methods in our controller class only. All we need is to annotate these methods with @ExceptionHandler annotation.

Steps to develop project:-

1)

@Controller

**public** **class** WelcomeRestController {

@GetMapping("/welcome")

**public** String getWelcome(Model model)

{

model.addAttribute("msg","welcome to java it");

String name=**null**;

name.length(); //wontedly write code

**return** "welcome";

}

@ExceptionHandler(value = NullPointerException.**class**)

**public** String handleNullException(Model model)

{

model.addAttribute("errMsg", "Some proble occured ,please try after some times");

**return** "error";

}

}

2)welcome.jsp

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

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

${msg}

</body>

</html>

3)error.jsp

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

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

${errMsg}

</body>

</html>

4)application.properties

server.port=9888

spring.mvc.view.suffix=.jsp

spring.mvc.view.prefix=/views/

2. Global Exception Handler:-

In Earlier we have seen controller based mechanism which is applicable for only one controller.

Using global mechanism we can handle the exceptions all controllers.

In spring provided @ControllerAdvice annotation to handle the Exceptions globally

Steps to develop project:-

1)

@Controller

**public** **class** WelcomeRestController {

@SuppressWarnings("null")

@GetMapping("/welcome")

**public** String getWelcome(Model model)

{

model.addAttribute("msg","welcome to java it");

String name=**null**;

name.length();

**return** "welcome";

}

}

@Controller

**public** **class** DateRestController {

@GetMapping("/date")

**public** String dateMsg(Model model) {

model.addAttribute("dateMsg", "Today's date is : "+**new** Date());

String s = **null**;

s.length();

**return** "dateDisplay";

}

}

2)error.jsp

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

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

${errMsg}

</body>

</html>

4)

@Controller

@ControllerAdvice

**public** **class** TestGlobalException {

@ExceptionHandler(value = NullPointerException.**class**)

**public** String handleNullExcepion(Model model)

{

model.addAttribute("errMsg", "some proble occured");

**return** "error";

}

}

3. Handler Exception Resolver:-

Steps to develop project:-

User Defined Exception:-

1)

@Controller

**public** **class** BookController {

@Autowired

**private** BookService bookService;

@GetMapping("/price")

**public** String getPrice(@RequestParam("bookId")String bookId,Model model)

{

Double price = bookService.findPriceById(bookId);

model.addAttribute("price", "Book Price: "+price);

**return** "display";

}

}

2)

**public** **interface** BookService {

**public** Double findPriceById(String bookId);

}

3)

@Service

**public** **class** BookServiceImpl **implements** BookService {

@Override

**public** Double findPriceById(String bookId) {

**if**(bookId.equals("B101")) {

**return** 450.00;

}

**else**

{

**throw** **new** NoBookFound("No book found");

}

}

}

4)display.jsp

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

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<h3>No Book Found </h3>

</body>

</html>

5)

**public** **class** NoBookFound **extends** RuntimeException {

/\*\*

\*

\*/

**private** **static** **final** **long** ***serialVersionUID*** = 226844612345870049L;

**public** NoBookFound(String message) {

**super**(message);

}

}

6)

@Controller

@ControllerAdvice

**public** **class** TestGlobalException {

@ExceptionHandler(value = NoBookFound.**class**)

**public** String NoBookPriceException()

{

**return** "noBook";

}

}

7)

URL: <http://localhost:9888/price?bookId=B102>

1)

@Data

@AllArgsConstructor

@RequiredArgsConstructor

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

**private** String pid;

**private** String pname;

**private** Double price;

}

2)

**public** **interface** ProdectService {

**public** Product findByProductId(Integer pid);

}

3)

@Service

**public** **class** ProdectServiceImpl **implements** ProdectService {

@Override

**public** Product findByProductId(Integer pid) {

**if**(pid==101)

{

**return** **new** Product("101", "KeyBoard", 800.00);

}

**else**

{

**throw** **new** ProductFoundException("No product found");

}

}

}

4)

**public** **class** ProductFoundException **extends** RuntimeException {

**private** **static** **final** **long** ***serialVersionUID*** = 717315701895271972L;

**public** ProductFoundException(String message) {

**super**(message);

}

}

5)

@RestController

**public** **class** ProductRestController {

@Autowired

**private** ProdectService prodectService;

@GetMapping(value = "/getProduct",produces= {"application/json"})

**public** Product getProductById(@RequestParam("pid")String pid,Model model )

{

Product findByProductId = prodectService.findByProductId(Integer.*parseInt*(pid));

**return** findByProductId;

}

}

6)

**public** **class** ProductFoundException **extends** RuntimeException {

**private** **static** **final** **long** ***serialVersionUID*** = 717315701895271972L;

**public** ProductFoundException(String message) {

**super**(message);

}

}

7)

@RestController

@RestControllerAdvice

**public** **class** RestExceptionHandler {

@ExceptionHandler(value = ProductFoundException.**class**)

**public** ResponseEntity<String> handleNoProductFound()

{

**return** **new** ResponseEntity<String>("No Product found",HttpStatus.***BAD\_REQUEST***);

}

}

8)Run

URL: <http://localhost:9888/getProduct?pid=102>