//------------With Multiple URIs:--------------//

@RestController

@RequestMapping("/home")

public class IndexController {

@RequestMapping(value = {

"",

"/page",

"page\*",

"view/\*,\*\*/msg"

})

String indexMultipleMapping() {

return "Hello from index multiple mapping.";

}

}

localhost:8080/home

localhost:8080/home/

localhost:8080/home/page

localhost:8080/home/pageabc

localhost:8080/home/view/

localhost:8080/home/view/view

//------------Using @RequestMapping With Dynamic URIs-------------------//

The @RequestMapping annotation is used in combination with the @PathVaraible annotation to handle dynamic URIs. In this use case, the URI values can act as the parameter of the handler methods in the controller. You can also use regular expressions to only accept the dynamic URI values that match the regular expression.

@RestController

@RequestMapping("/home")

public class IndexController {

@RequestMapping(value = "/fetch/{id}", method = RequestMethod.GET)

String getDynamicUriValue(@PathVariable String id) {

System.out.println("ID is " + id);

return "Dynamic URI parameter fetched";

}

@RequestMapping(value = "/fetch/{id:[a-z]+}/{name}", method = RequestMethod.GET)

String getDynamicUriValueRegex(@PathVariable("name") String name) {

System.out.println("Name is " + name);

return "Dynamic URI parameter fetched using regex";

}

}

//------ Using @RequestMapping With Dynamic URIs-------------------//

The @RequestMapping annotation is used in combination with the @PathVaraible annotation to handle dynamic URIs. In this use case, the URI values can act as the parameter of the handler methods in the controller. You can also use regular expressions to only accept the dynamic URI values that match the regular expression.

@RestController

@RequestMapping("/home")

public class IndexController {

@RequestMapping(value = "/fetch/{id}", method = RequestMethod.GET)

String getDynamicUriValue(@PathVariable String id) {

System.out.println("ID is " + id);

return "Dynamic URI parameter fetched";

}

@RequestMapping(value = "/fetch/{id:[a-z]+}/{name}", method = RequestMethod.GET)

String getDynamicUriValueRegex(@PathVariable("name") String name) {

System.out.println("Name is " + name);

return "Dynamic URI parameter fetched using regex";

}

}

//----------------------------------------------------------------//

2:

@RequestMapping With @RequestParam

@RestController

@RequestMapping("/home")

public class IndexController {

@RequestMapping(value = "/id")

String getIdByValue(@RequestParam("id") String personId) {

System.out.println("ID is " + personId);

return "Get ID from query string of URL with value element";

}

@RequestMapping(value = "/personId")

String getId(@RequestParam String personId) {

System.out.println("ID is " + personId);

return "Get ID from query string of URL without value element";

}

}

//-@RequestMapping With @RequestParam and required--------//

@RestController

@RequestMapping("/home")

public class IndexController {

@RequestMapping(value = "/name")

String getName(@RequestParam(value = "person", required = false) String personName) {

return "Required element of request param";

}

}

//---------The @RequestMapping Default Handler Method----------//

@RestController

@RequestMapping("/home")

public class IndexController {

@RequestMapping()

String

default () {

return "This is a default method for the class";

}

}

//--------------------------------------------------------------//