



<http://www.lycato.com>



Spring Boot

Spring MVC

Nguyễn Nghiệm



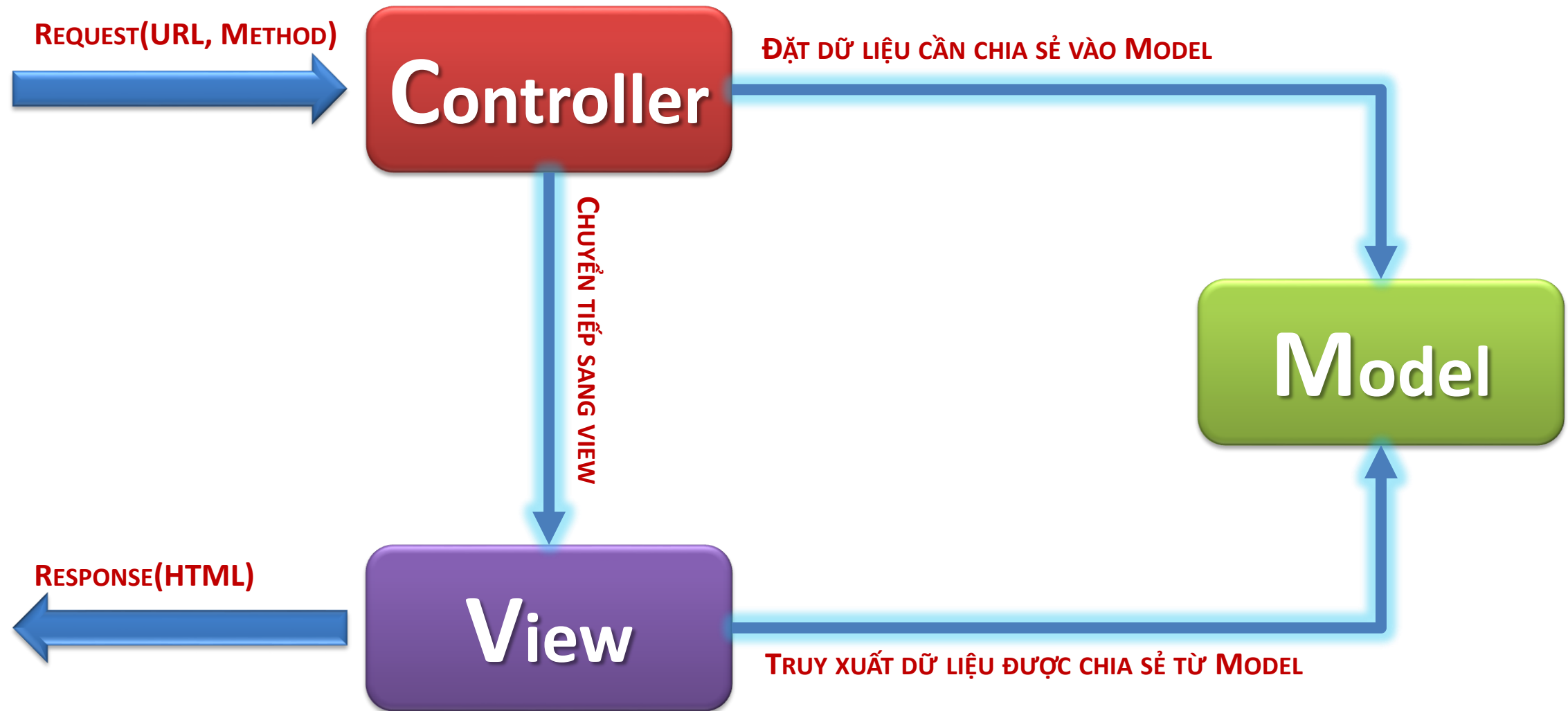


- ☐ MVC PATTERN
- ☐ REQUEST MAPPING
- ☐ READING DATA
 - ❖ PATH VARIABLE
 - ❖ READ PARAMETER
 - ❖ UPLOAD FILE
 - ❖ READ COOKIE
- ☐ MODEL
- ☐ RETURN OF CONTROL METHOD





MVC PATTERN



“view”
“forward:/url”
“redirect:/url”

Return

Mapping

@RequestMapping
@GetMapping
@PostMapping

Model

Parameter

@ModelAttribute
Model.addAttribute

@PathVariable
@RequestParam
@RequestPart
JavaBean



Requests — Mapping —> Method

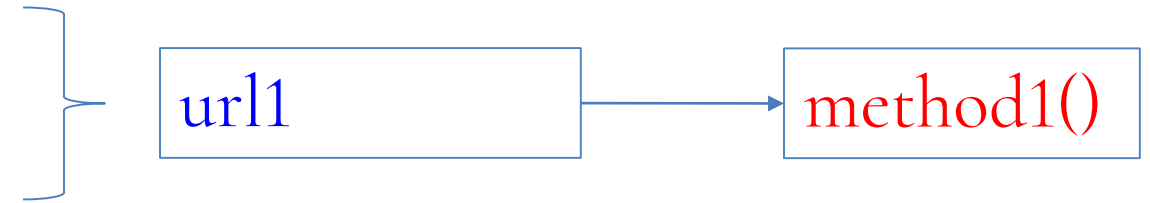
- ❑ Request Mapping (ánh xạ request) là sự kết hợp *các request* từ trình duyệt với một *phương thức trong Controller*
- ❑ Để thực hiện việc ánh xạ, hãy đặt *@RequestMapping()* ngay trên phương thức trong Controller
 - ❖ @RequestMapping(URL)
 - ❖ *@GetMapping*(URL) ~ @RequestMapping(URL, RequestMethod.GET)
 - ❖ *@PostMapping*(URL) ~ @RequestMapping(URL, RequestMethod.POST)



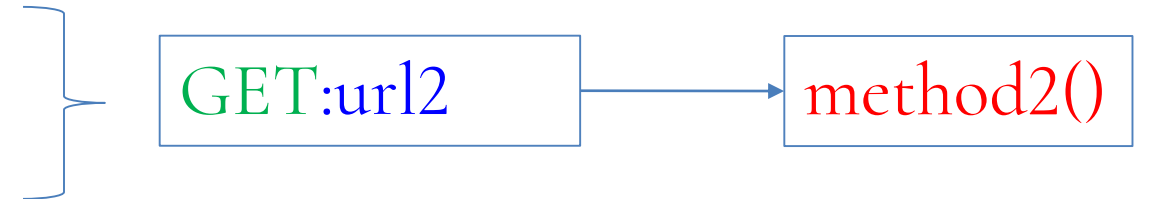
REQUEST MAPPING

@Controller

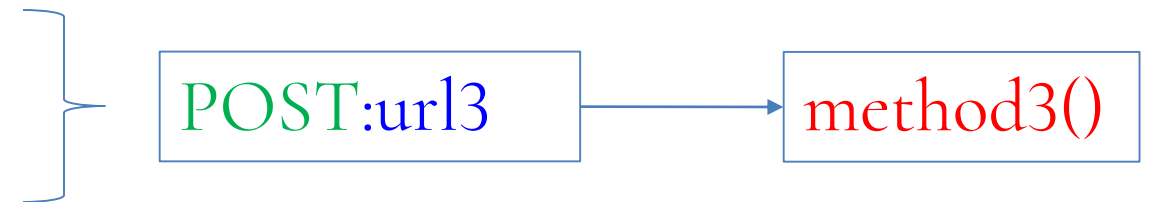
```
public class MyController {  
    @RequestMapping("url1")  
    public String method1() {...}
```



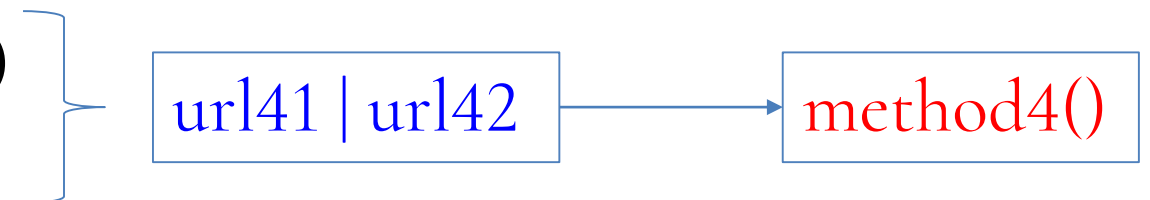
```
    @GetMapping("url2")  
    public String method2() {...}
```



```
    @PostMapping("url3")  
    public String method3() {...}
```



```
    @RequestMapping({"url41", "url42"})  
    public String method4() {...}
```



```
}
```

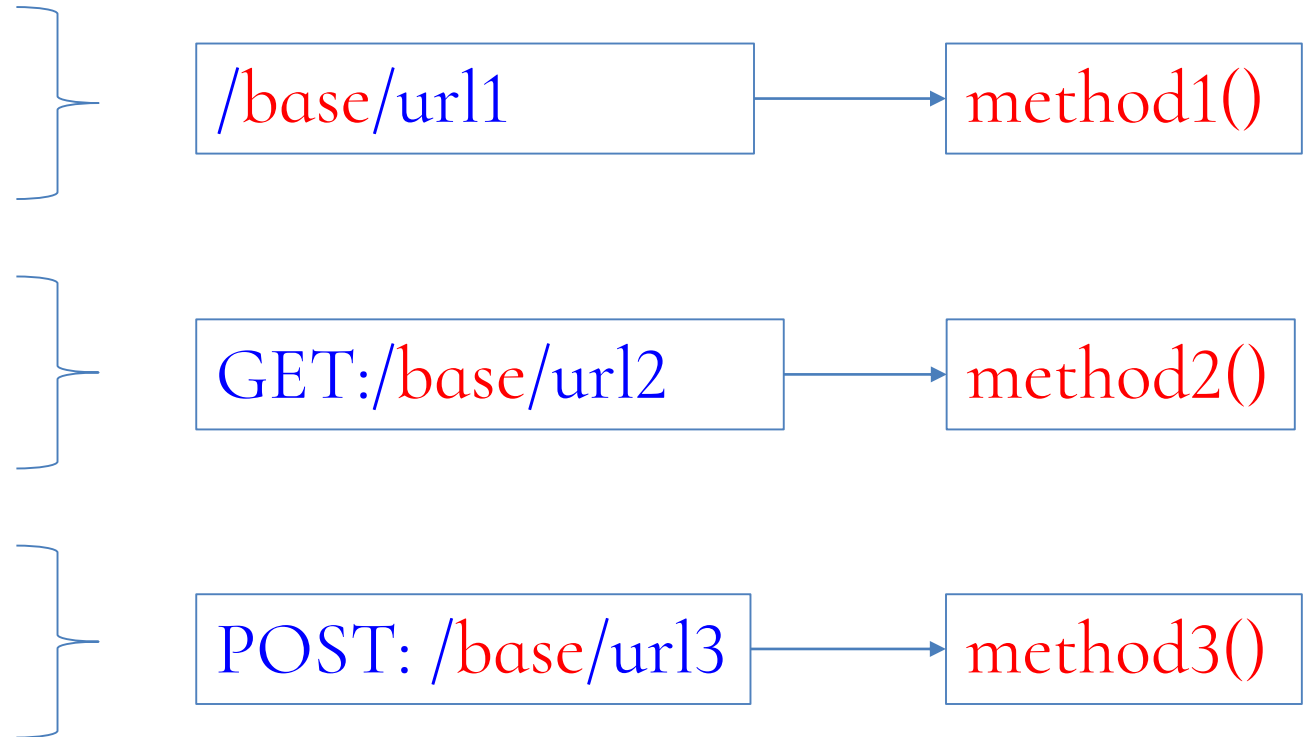


REQUEST MAPPING

```
@Controller
@RequestMapping("base")
public class MyController {
    @RequestMapping("url1")
    public String method1() {...}

    @GetMapping("url2")
    public String method2() {...}

    @PostMapping("url3")
    public String method3() {...}
}
```





@Controller

```
public class ProductController {  
    // /product/detail/1000, /product/detail/1234...  
    @RequestMapping("/product/detail/{id}")  
    public String detail(@PathVariable("id") Integer id) {...}  
  
    // /product/detail, /product/detail/1000, /product/detail/1234...  
    @RequestMapping({"/product/detail", "/product/detail/{id}"})  
    public String detail(@PathVariable("id") Optional<Integer> id) {...}  
  
    // /list-by-category/iphone 10 hot nhất hiện nay/1234...  
    @RequestMapping("/list-by-category/{name}/{id}")  
    public String listByCategory(@PathVariable("id") Integer id) {...}  
}
```




WHAT IS PARAMETER?

- ❑ Parameter (tham số) là dữ liệu được gửi từ form hoặc phần sau dấu ? của url.
- ❑ Ví dụ: Khi click vào link hoặc bấm nút [Submit] thì dữ liệu được gửi đến phương thức trong Controller được ánh xạ với url thông qua 2 parameter là p1 và p2.

```
<a href="url?p1=123&p2Hello">Link</a>
```

```
<form action="url">  
  <input name="p1">  
  <input name="p2">  
  <button>Submit</button>  
</form>
```



READING PARAMETERS

@Controller

```
public class MyController {  
    @RequestMapping("url")  
    public String method(@RequestParam("p1") Integer p1,  
        @RequestParam("p2") String p2) {...}
```

@RequestMapping("url")

```
public String method(HttpServletRequest request) {  
    String p2 = request.getParameter("p2");  
    Integer p1 = Integer.parseInt(request.getParameter("p1"));  
}
```

@RequestMapping("url")

```
public String method(MyClass bean) {...}  
}
```

@Data

```
public class MyClass{  
    Integer p1;  
    String p2;  
}
```



READING LIST CONTROL PARAMETERS

```
<input type="radio" name="p1" value="true">  
<input type="radio" name="p1" value="false">
```

} @RequestParam("p1") *Boolean* p1

```
<select name="p2">  
  <option value="VN">Việt Nam</option>  
  <option value="US">United States</option>  
</select>
```

} @RequestParam("p2") *String* p2

```
<input type="checkbox" name="p3" value="0">  
<input type="checkbox" name="p3" value="1">  
<input type="checkbox" name="p3" value="2">  
<input type="checkbox" name="p3" value="3">
```

} @RequestParam("p3") *List<Integer>* p3



```
<form action="url" method="post" enctype="multipart/form-data">  
  <input type="file" name="p1">  
  <button>Submit</button>  
</form>
```

```
@RequestMapping("url")  
public String method(@RequestPart("p1") MultipartFile p1) {  
    try {  
        String path = "c:/temp/" + p1.getOriginalFilename();  
        p1.transferTo(new File(path));  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
    return "path";  
}
```



vnexpress.net

SUN RIVERSIDE VILLAGE
Sống tiện
NGHỈ DƯỠNG
ĐẦU TƯ S
HOTLINE: 094

VNEXPRESS

Mới nhất International Đăng nhập

Elements Console Recorder Application

Local Storage
Session Storage
IndexedDB
Web SQL
Cookies
https://vnexpress.net
https://s.eclick.vn
https://g.eclick.vn
https://d7710603ea21b5638
https://static.eclick.vn
https://gum.criteo.com
https://tpc.google syndicatic
https://www.google.com
Trust Tokens

Filter Only show cookies with

Name	Value	[P	E	S	H	S	S	S	P	P
__Secure-1PS...	IQgRETg...	...	/	2...	8...	✓	✓		✓		H..
SIDCC	AJi4QfHf...	...	/	2...	8...						H..
HSID	Ah1jkO...	...	/	2...	2...	✓					H..
__Host-3PLSID	o.admin...	a..	/	2...	3...	✓	✓	N..			H..
SID	IQgRETg...	...	/	2...	7...						H..
__Host-1PLSID	o.admin...	a..	/	2...	3...	✓	✓		✓		H..
SMSV	ADHTE...	a..	/	2...	1...	✓	✓				H..
__Secure-3PA...	GG2EST5...	...	/	2...	5...		✓	N..			H..
__Secure-3PS...	IQgRETg...	...	/	2...	8...	✓	✓	N..			H..
__Secure-3PS...	AJi4QfFx...	...	/	2...	9...	✓	✓	N..			H..
SID	IQgRETg...	...	/	2...	7...						H..

@RequestMapping("url")
public String method(@CookieValue("SID") String sid) {
...
}

g web socket



- ❑ Model là nơi chứa dữ liệu chia sẻ giữa Controller và View (*Controller đặt data vào Model để truyền sang View*).
- ❑ Cấu trúc dữ liệu của Model tương tự như Map<String, Object>

```
@RequestMapping("/url")  
public String share(Model model) {  
    model.addAttribute("message", "Spring Boot 1");  
    model.addAttribute("now", new Date());  
    model.addAttribute("Spring Boot 2");  
    model.addAttribute(new Date());  
    return "path";  
}
```

Model

name	value
message	"Spring Boot 1"
now	Date object
string	"Spring Boot 2"
date	Date object

Tên của attribute là tên kiểu của value



@Controller

```
public class MyController {  
    @RequestMapping("url")  
    public String method(Model model) {  
        model.addAttribute("now1", new Date());  
        return "path";  
    }  
    @ModelAttribute("now2")  
    public Date getNow() {  
        return new Date();  
    }  
    @RequestMapping("url")  
    public String method(@ModelAttribute("now3") Date date) {  
        return "path";  
    }  
}
```




RETURN OF CONTROL METHOD

```
@RequestMapping("url1")  
public String method1() {  
    return "path";  
}
```

Chuyển tiếp sang **path.html**

```
@RequestMapping("url2")  
public void method2() {  
}
```

Chuyển tiếp sang **url2.html**

```
@RequestMapping("url3")  
public String method3() {  
    return "forward:/url1";  
}
```

Chuyển tiếp sang **/url1** (method1() sẽ chạy)

```
@RequestMapping("url4")  
public String method4() {  
    return "redirect:/url1";  
}
```

Tạo request mới đến **/url1**



- ✓ MVC PATTERN
- ✓ REQUEST MAPPING
 - ✓ READING DATA
 - ✓ PATH VARIABLE
 - ✓ READ PARAMETER
 - ✓ UPLOAD FILE
 - ✓ READ COOKIE
- ✓ MODEL
- ✓ RETURN OF CONTROL METHOD





Thank
you