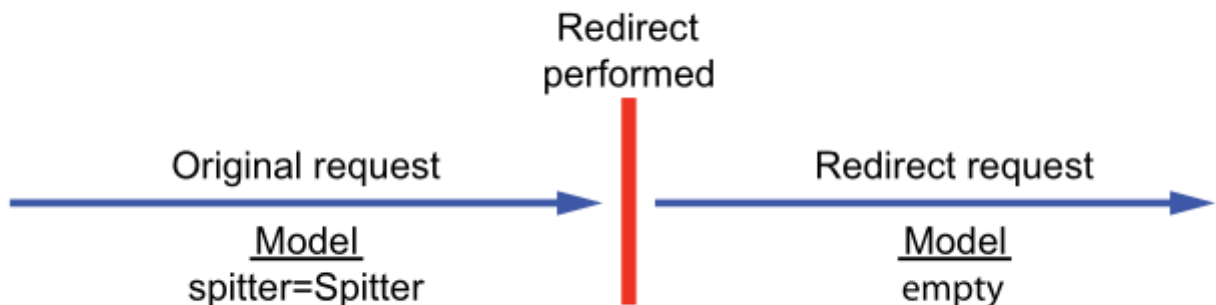


redirect와 forward

스프링 컨테이너 메소드의 리턴값으로 `redirect:` 또는 `forward:` 라는 prefix를 붙여주면 `UrlBasedViewResolver`가 이를 해석하여 해당 URL 주소로 리다이렉트 또는 포워드 요청이 가게 됩니다. 그런데 `forward`와 달리 `redirect`의 경우, 지정된 주소로 새로운 HTTP GET 요청이 시작되기 때문에 실행 이전에 수행된 모델 데이터가 소멸합니다. 따라서 `redirect`를 할 때는 Request 객체나 Model 객체에 데이터를 담아도 전달되지 않습니다.



다만 기본적으로 리다이렉트시 Model 객체의 속성은 URL 주소의 쿼리 스트링으로 추가되어 전달됩니다.

```
@RequestMapping("/main")
public String main() {
    return "main";
}

@RequestMapping("/redirect")
public String redirect(Model model) {
    model.addAttribute("mesg", "hello");
    return "redirect:main";
}
```

```
<h1>main.jsp</h1>
```

```
Model 객체 : ${ mesg } <br>
```

```
Model 객체 : <%= request.getAttribute("mesg") %> <br>
```

```
<hr>
```

```
파라미터 : ${ param.mesg } <br>
```

```
파라미터 : <%= request.getParameter("mesg") %> <br>
```

← → ↻ ⓘ localhost:8092/redirect/main?mesg=hello

main.jsp

Model 객체 :

Model 객체 : null

파라미터 : hello

파라미터 : hello

/redirect 주소 요청시

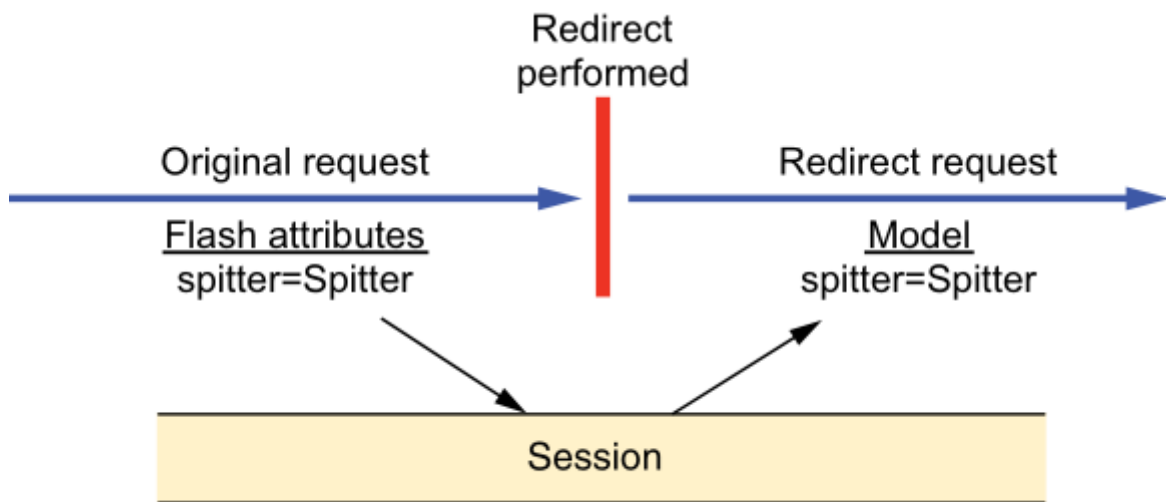
파라미터로 전달된 것을 확인할 수 있습니다. 그것이 원래의 의도였다면 괜찮지만, 이 경우 단순히 GET 방식으로 직접 URL을 지정해주는 방법이 있기 때문에 이를 의도하지는 않았을 겁니다. 더구나 데이터가 모두 URL에 노출되는 단점이 있습니다. 이때 사용할 수 있는 객체가 RedirectAttributes 입니다.

RedirectAttributes

@RequestMapping 메소드의 파라미터 타입으로 RedirectAttributes 객체를 지정하면 특정 속성을 리다이렉트되는 뷰 페이지에서 사용할 수 있도록 전달할 수 있습니다. 메소드가 리다이렉트되면 RedirectAttributes 객체의 내용이 사용되고, 리다이렉트되지 않는 경우 Model 객체의 내용이 사용됩니다.

Flash Attributes

RedirectAttributes를 사용하여 리다이렉트되는 주소로 데이터를 전달하기 위해 플래시 속성(Flash Attributes)을 사용합니다. 플래시 속성은 하나의 요청에서 속성을 저장하고 다른 곳에서 사용하기 위한 방법을 제공합니다. 대표적인 경우는 Post-Redirect-Get 패턴의 리다이렉트입니다. 플래시 속성을 사용하여 데이터를 전달하는 과정은 다음과 같습니다.



1. @RequestMapping 메소드의 파라미터로 RedirectAttributes를 받아서 RedirectAttributes.addFlashAttribute 메소드로 플래시 속성을 추가합니다.
2. RedirectAttributes를 통해 추가된 플래시 속성은 리다이렉트가 되기 전에 임시로 세션에 저장됩니다.
3. 리다이렉트가 이루어진 후 세션에 저장되었던 플래시 속성은 타겟 URL을 제공하는 컨트롤러의 Model 객체에 자동으로 추가됩니다.
4. 이후 즉시 제거됩니다.

이제 다시 RedirectAttributes 객체를 이용해 리다이렉트되는 페이지로 데이터를 제대로 전달해 보겠습니다.

```
@RequestMapping("/redirect2")
public String redirect(RedirectAttributes rttr) {
    rttr.addFlashAttribute("mesg", "hello");
```

```
    return "redirect:main";  
}
```

<h1>main.jsp</h1>

Model 객체 : \${ mesg }

Model 객체 : <%= request.getAttribute("mesg") %>

<hr>

파라미터 : \${ param.mesg }

파라미터 : <%= request.getParameter("mesg") %>

← → ↻ ⓘ localhost:8092/redirect/main

main.jsp

Model 객체 : hello

Model 객체 : hello

파라미터 :

파라미터 : null

/redirect2 주소 요청시

파라미터가 아닌 Model 객체로부터 데이터를 가져오는 것을 확인할 수 있습니다.