포워딩(Forwarding)

포워딩 이란

어떠한 요청에 대해 해당 페이지에서 처리하지않고 다른 페이지로 권한을 위임하는 것을 포워딩이라함

URL은 바뀌지않으며, 내부에서만 접근이 가능하다.

다른 페이지로의 이동과 같은 페이지의 흐름을 제어한다.

서블릿에서의 포워딩은 두가지

하나는 RequestDispatcher 클래스 이고, 또하나는 HttpServletResponse 클래스

RequestDispatcher는 권한을 위임하여 기존 객체를 전송

HttpServletResponse는 새로운 request객체를 생성하여 전송

현재의 jsp 페이지에서 다른 페이지로 이동하는 태그. jsp컨테이너는 현재 페이지에서 forward 를

만나면 그전까지 출력 버퍼에 저장되어 있던 내용을 모두 삭제하고 forward액션태그에 설정된 페이지로

프로그램제어와 응답처리권등을 영구적으로 넘긴다.

컨트롤러는 외부 jsp로 이동하지만, 웹 브라우저의 주소(url)는 처음에 요청한 jsp 파일이다.

sendRedirect()와 차이점

리다이렉트는 sendRedirect() 함수가 호출되는 시점에 페이지를 바로 이동하는게 아니라 서블릿 클래스 내에 코드가 모두 실행되고 반환될 때, 웹서버의 서블릿 클래스가 클라이언트의 요청을 바꿔서 다시 요청받는거다.

독립적인 재요청을 보낸다

1. 리다이렉트는 url 주소가 달라지고, 포워드는 url 주소가 달라지지 않는다.

리다이렉트는 클라이언트가 서버에게 요청을 보내면 서버는 어떠한 일을 처리하고 클라이언트에게 새롭게 요청할 곳을 알려준다. 그래서 리다이렉트의 결과는 실제 실행한 뒤에 url 주소가 달라진다.

반면, 포워드는 클라이언트가 요청을 보냈을 때 서버 쪽에서 혼자 처리하는 것이 아니라 또 다른 back에게 일을 넘기는 것이다. 포워드의 경우 클라이언트는 서버에서 서블릿이 혼자 일을 처리했는지, 다른 서블릿에게 일을 위임했는지 알 필요가 없다. 그래서 포워드가 실행되면 url주소는 바뀌지 않는다.

2. 리다이렉트는 request, response 객체가 여러 번 생성되고, 포워드는 한 번만 생성된다.

실제 클라이언트가 서버에게 요청을 하면 반드시 생성되는 객체가 request와 response이다. 요청이 들어와서 응답을 할 때까지 계속 유지가 되어야 한다. 포워드 같은 경우는 이 객체들이 한 번만 만들어진다. 반면, 리다이렉트는 요청과 응답이 여러번 실행되기 때문에 request, response 객체가 여러 번 만들어진다.

forward()는 제어를 넘기기 이전에 출력 버퍼를 비우기 때문에 a.jsp -> b.jsp로 호출시 a.jsp에서 어떤 내용을 버퍼에 출력했더라도 무시되며 제어가 넘어간 b.jsp의 출력 내용만 브라우저에게 전달되므로 주의하도록 합니다.

간단히 설명하면 sendRedirect() 메서드를 사용하면 서버가 request에 대해 response하고 난 뒤

메서드에 입력된 경로로 이동한다는 뜻이다.

이때 forward와 차이점은 sendRedirect()를 쓰면 클라이언트와 서버 간 통신이 끝난다는 것이다.

RequestDispatcher객체의 forward는 일부러 response하지 않고 응답을 끌면서 다른 리소스로 전달한다.

반면 response객체의 sendRedirect()는 응답을 끝내는 역할을 한다.

이때 Http 프로토콜의 통신 방식을 알면 두 메서드가 어떤 차이가 있는지 명확히 알 수 있다.

forward는 다른 리소스로 패스하고, sendRedirect는 response 객체와 함께 응답하며 통신을 끝내는 것이다.

두 메서드는 모양새는 다르지만 결과적으론 각자의 객체를 다른 리소스로 전달하는 역할을 한다.

Core 라이브러리

Choose 문