

The HTTP Protocol

Monitoring HTTP Headers 1

I see the page index and the header of that page which are both get requests. It also contains the status of the get requests. Also, it builds and shows you the build of the data layer.

After changing the name of the index to index1 it now cannot get the right path because it does not match and thereby the first get request gives a 404 error code.

Monitoring HTTP Headers 2

The first request gets a status of 301 which indicates that it was redirected. While the next new one is getting a 200 status which means it got a successful response.

```
<div>
```

```
    
```

```
</div>
```

Monitoring HTTP Headers 3 (Response-codes 3xx)

The first request is with param /redirect and does get redirected which is also stated through the status of 302.

The second request requests to get the information from the r.html file.

The browser knew where to go because of the response of the first request that send it to the redirect servlet that says to respond with sending the r.html file back if the request has /redirect.

Redirecting to HTTPS instead of HTTP

First I request to go to http but the dns answers with that it have been permanently moved to another and therefore redirects us. The second request is where the site have been moved too which is the https through the dns.

The origin server did not find a current representation for the target resource or is not willing to disclose that one exists. From the 404 in the browser. The status of the request gives you a 404 status which is when there is an error in the client side. 500 and over in error status is when there is something wrong with the server.

Status Codes (5xx)

I get an 404 error and not the 5xx error.

Status Codes (4xx)

I get a 404 error status code that signals that something is wrong on the client side of things.

Status Codes – Ranges

The 2xx is signaled for when the response is successful.

The 3xx is signaled when the request gets redirected to somewhere else.

The 4xx signals when there is a client-side error.

The 5xx signal error in the server side of things.

Get HTTP Request Headers on the Server


Get/Post-parameters

With the method of “get” you can see the information of the hidden value, firstname and lastname within the url. And the values get send through a query string.

With the “post” method the information about the hidden value and such is not shown in the url. But can be seen still in the developer console. The value gets send by form data.

Sessions (Session Cookies)

It took the name of which was put in the form from the input field and placed it into a session which then in response uses that name to put into the url and giving it the parameter of name.

 localhost:8080/week2day3part2/SessionDemo?name=Red

At the same time it send the name to the jsp site that was then generated and the uses by taking the session and using it in the text.

Welcome Red !

The code for the input of the name and putting it into the session is seen below.

```
out.println("<h2>Please enter your name, and submit</h2>");
out.println("<form action='SessionDemo'>");
out.println("<input type='input' name='name'>");
out.println("<input type='submit'></form>");
```

It also sends the input with the attribute name and then sends it back to the servlet.

```
out.println("<form action='SessionDemo'>");
out.println("<input type='input' name='name'>");
out.println("<input type='submit'></form>");
```

First we start on the servlet and state that a string with the name of “name” is requesting to get the parameter that is “name” from the session. Then in a if else statement we say if name is not equal to null (empty) than we request to set that attribute from the session and put it on as “name”. Else it should get the value already stored in the session with the attribute “name”.

```
String name = request.getParameter("name");
if (name != null) {
    request.getSession().setAttribute("name", name);
} else {
    name = (String) request.getSession().getAttribute("name");
}
```

But before the that we have to check to make sure that no name is already in the session “name” if there is then we give them the welcome message with the attribute with got from the session.

Below is the code that checks if there exists an attribute in the session with name.

```
if (name != null) {
    name = (String) request.getSession().getAttribute("name");
    out.println("<p> Welcome " + name + " !</p>");
}
```