Fundamental Skills - Inspecting a Webpage

Category	Experience Level
Web	Complete Beginner

Contents

Page Content

When you visit a webpage, your browser makes a HTTP request to that website's server (more on this later). The server sends a response back to your browser containing the contents of the page, and the browser renders it.

All browsers render this content slightly differently, but the core principle is the same - the server returns a mix of HTML, Javascript, and CSS, which the browser interprets according to internal rules.

HTML (Hypertext Markup Language) describes the elements of the page, including headings, paragraphs of text, forms, images, and more.
Javascript is a programming language that is executed within the browser; it is capable of many things, including making web requests, changing elements on the page, and accessing site data.

CSS (Cascading Style Sheets) is a way of describing the style of the page, such as text color, font size, and the size and positioning of elements.

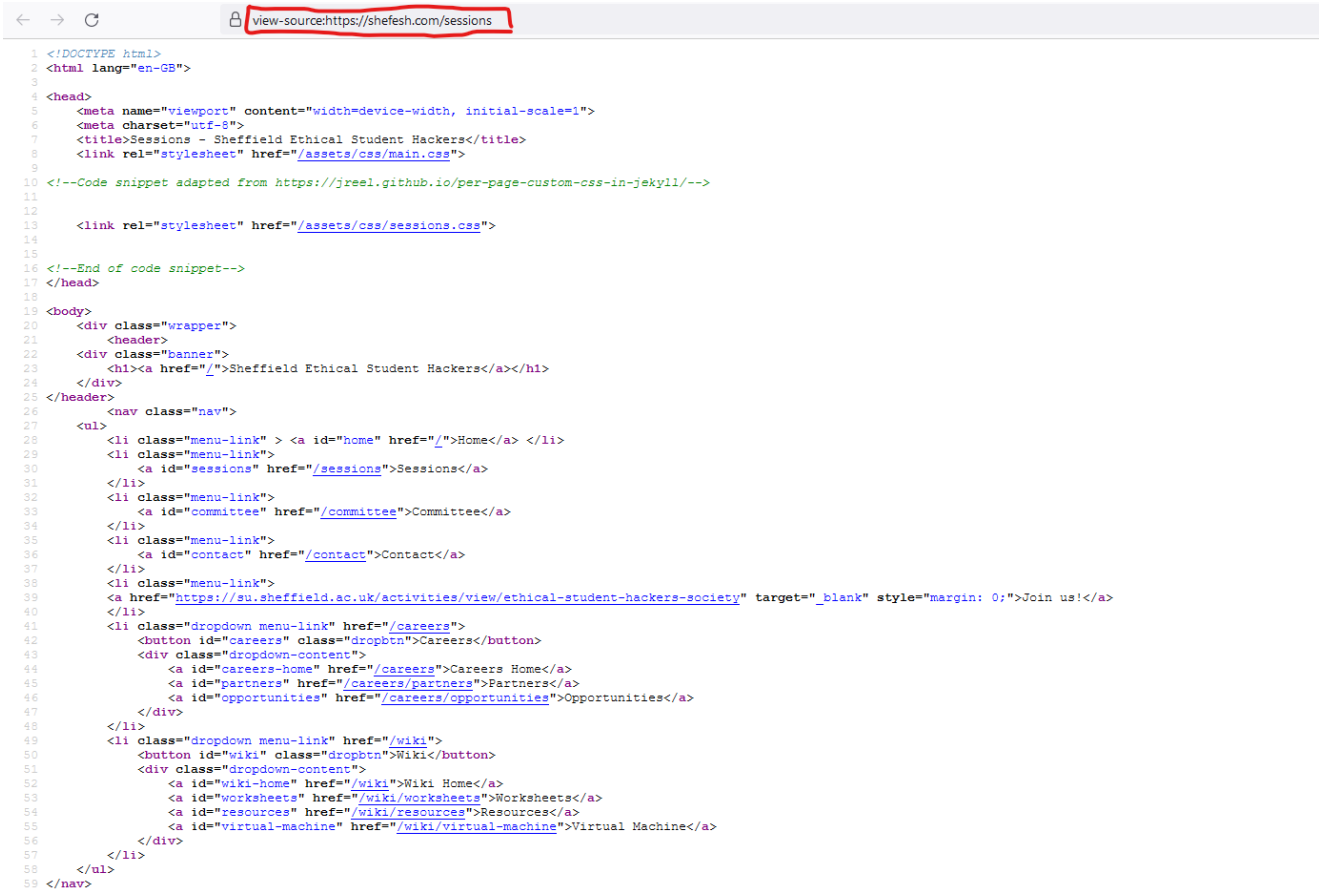
```
<!DOCTYPE html>
<html lang="en-GB">

<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta charset="utf-8">
    <title>Home - Sheffield Ethical Student Hackers</title>
    <link rel="stylesheet" href="/assets/css/main.css">

<!--Code snippet adapted from https://jreel.github.io/per-page-custom-css-in-jekyll/-->

<!--End of code snippet-->
</head>

<body>
    <div class="wrapper">
        <header>
    <div class="banner">
        <h1><a href="/">Sheffield Ethical Student Hackers</a></h1>
    </div>
</header>
        <nav class="nav">
    <ul>
        <li class="menu-link" > <a id="home" href="/">Home</a> </li>
        <li class="menu-link">
            <a id="sessions" href="/sessions">Sessions</a>
        </li>
        <li class="menu-link">
            <a id="committee" href="/committee">Committee</a>
        </li>
        <li class="menu-link">
            <a id="contact" href="/contact">Contact</a>
        </li>
        <li class="menu-link">
        <a href="https://su.sheffield.ac.uk/activities/view/ethical-student-hackers-society" target="_blank" style="margin: 0;">Join us!</a>
        </li>
        <li class="dropdown menu-link" href="/careers">
            <button id="careers" class="dropbtn">Careers</button>
            <div class="dropdown-content">
                <a id="careers-home" href="/careers">Careers Home</a>
                <a id="partners" href="/careers/partners">Partners</a>
                <a id="opportunities" href="/careers/opportunities">Opportunities</a>
            </div>
        </li>
        <li class="dropdown menu-link" href="/wiki">
            <button id="wiki" class="dropbtn">Wiki</button>
            <div class="dropdown-content">
                <a id="wiki-home" href="/wiki">Wiki Home</a>
                <a id="worksheets" href="/wiki/worksheets">Worksheets</a>
                <a id="resources" href="/wiki/resources">Resources</a>
                <a id="virtual-machine" href="/wiki/virtual-machine">Virtual Machine</a>
            </div>
        </li>
    </ul>
</nav>
```

> Above: a snippet of the HTML code on the shefesh.com site

Viewing the Site Source

In Chrome and Firefox you can press `Ctrl + U` to view the source code for a site.

Pressing this button will open the source code in a new tab:



This just shows the HTML code for the site - CSS and Javascript can be viewed separately via the Developer Tools.

Hidden Elements

A webpage can have hidden content. Sometimes this may tell us more about how the site works, or even be a pointer to sensitive information we shouldn't be able to see.

Comments are a way of writing HTML code that is ignored by the browser and not rendered - they can serve as annotations for the code, or 'secret' messages:

```
<!--Code snippet adapted from https://jreel.github.io/per-page-custom-css-in-jekyll/-->

    <link rel="stylesheet" href="/assets/css/sessions.css">

<!--End of code snippet-->
```

> Above: some comments in the SESH source code, highlighted in green

It's sometimes worth searching the site source for comments, by viewing the source then searching with `Ctrl + F` and typing `<!--` (the characters that indicate the beginning of a comment).

A page may also have elements that are hidden with styling, but not commented. For example, many forms have `<input>` elements with the attribute `type="hidden"` - this

might indicate a value that is to be submitted to the server but not modified by the user (such as a default value).

Furthermore, some elements may have the attribute `style="display: none"`, which makes them invisible on the page.

Inferring Back-end Functionality

Even if an element isn't hidden, it may contain useful information.

In dynamic websites (i.e. sites with a back-end programming language such as Ruby or PHP), elements sometimes have attributes that help them communicate with a server.

For example, a page that renders user comments may contain the user ID of the commenter as an attribute in the comment element (e.g. `userid="1234"`). We can sometimes use this information to enumerate users.

```
<div id="comments">
  <p userid="2079">This site is terrible</p>
  <p userid="1056">Love this content! Keep it up</p>
</div>
```

Similarly, elements such as forms may sometimes expose parts of a website that aren't well known or publicly available. A `<form>` element's `action=` attribute, for example, may point to a URL such as `/api/verify-user` - this shows us the existence of an API, which we can then try to interact with.

```
<form action="/api/v4/update-metrics" method="POST">
  <label for="name">Your Name:</label><br>
  <input type="text" id="name" name="name"><br>
  <input type="hidden" name="secret_input" value="secret_default_value">
  <input type="submit" value="Submit!">
</form>
```

> Above: a form with elements that expose an API endpoint, and a hidden input field

Developer Tools

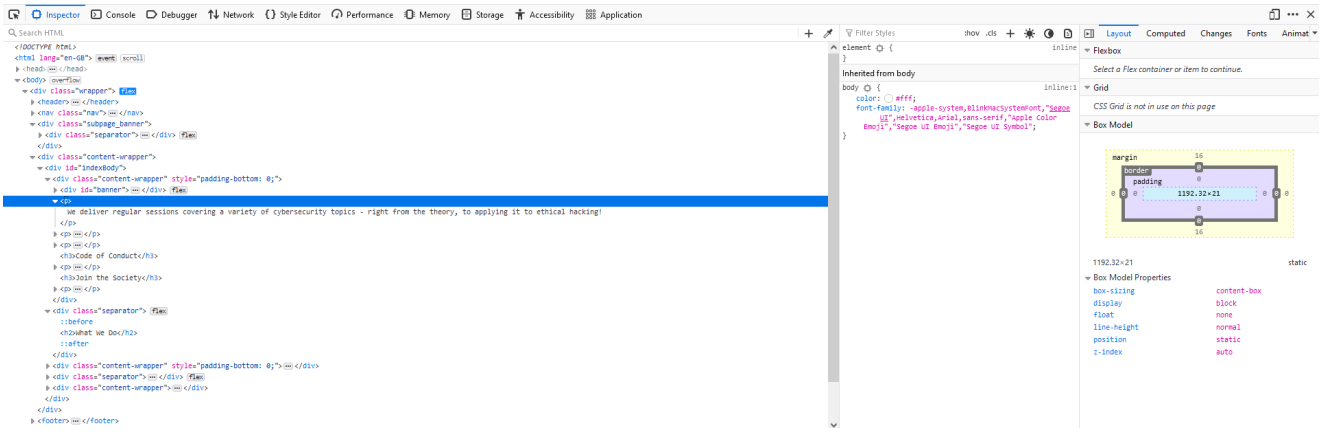
The Developer Tools are a powerful feature of most modern browsers used for inspecting and debugging site content.

They're useful for web developers and hackers alike, as they allow modification of the browser-side rendering of the site, and the viewing of Javascript code.

You can open the Developer Tools on most browsers by pressing `F12`.

Inspector

The first tab in the Developer Tools is the Inspector Tab. It allows you to view elements on the page, and their attributes, as HTML code.



> Above: viewing the site source using the Developer Tools

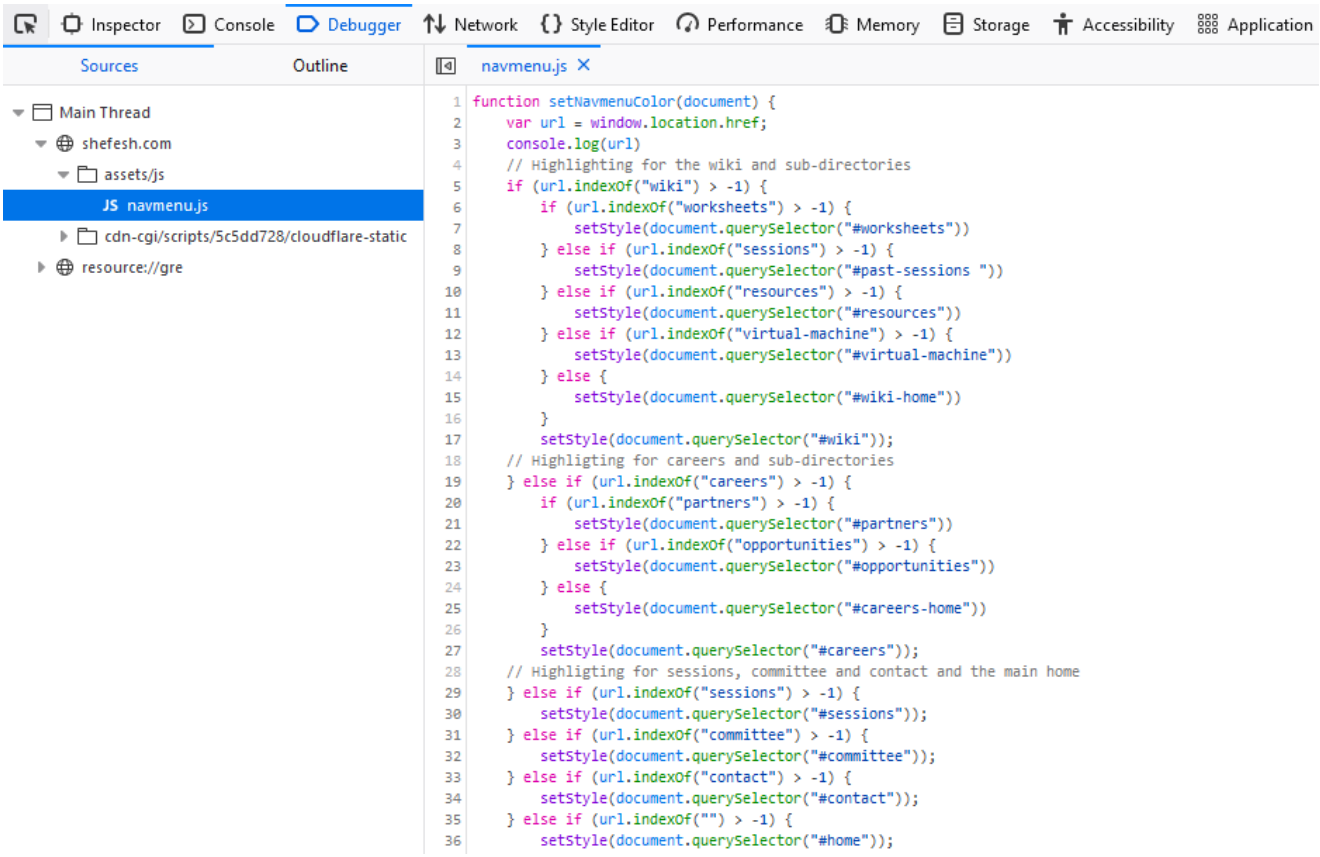
By default some elements are minimised - expand them to see what's inside by clicking the ellipsis button.

On large sites, it may be hard to find the element you're looking for. To jump to a specific part of the page, you can press `Ctrl + Shift + C` to view the source for any element that you hover over.

Not only can this be used to view elements, but it can change them too. Double clicking an attribute allows it to be modified arbitrarily. This can be used to bypass features such as the `type="hidden"` attribute on form fields, by simply deleting the attribute.

Debugger

The Debugger allows us to view client-side code running on the site, such as Javascript:

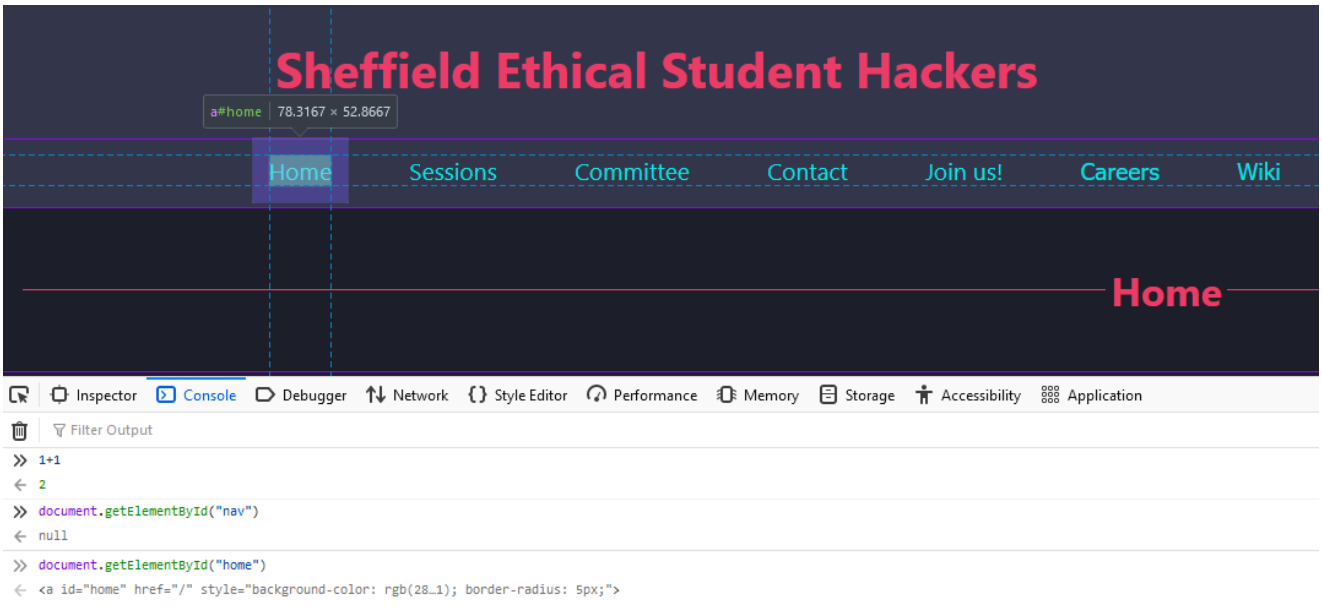


This can help us understand how certain dynamic functionality of the page works.

Console

The Console tab will show us the output of any Javascript on the page, including logs and error messages.

We can also run arbitrary javascript in the console:



This is useful when we may want to access Javascript variables defined in the code, to see how they behave.

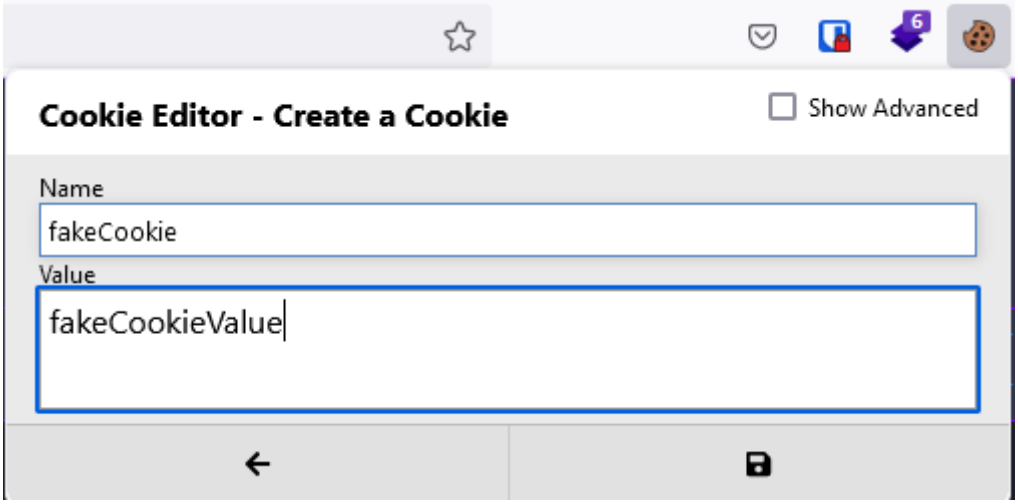
Storage

The storage tab show us any cookies we might have on the site we are viewing. This is good for understanding many aspects of site functionality.

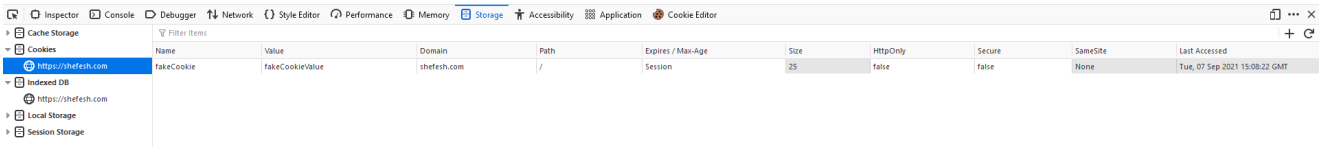
A cookie's name and value might give us clues about how the site handles logins, what we are authorised to do, and what technology the site is using.

Modifying Cookies

We can add and modify cookies arbtirarily in our browser. A good way to do this is with a Cookie Editor plugin.

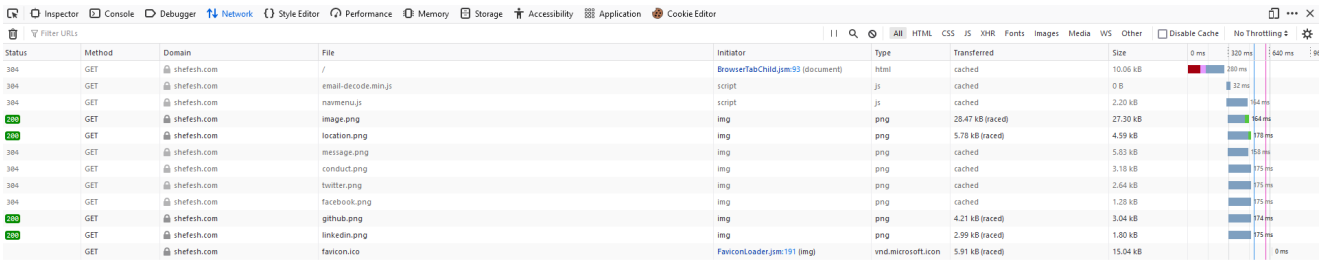


The cookie will then appear in the Storage tab:



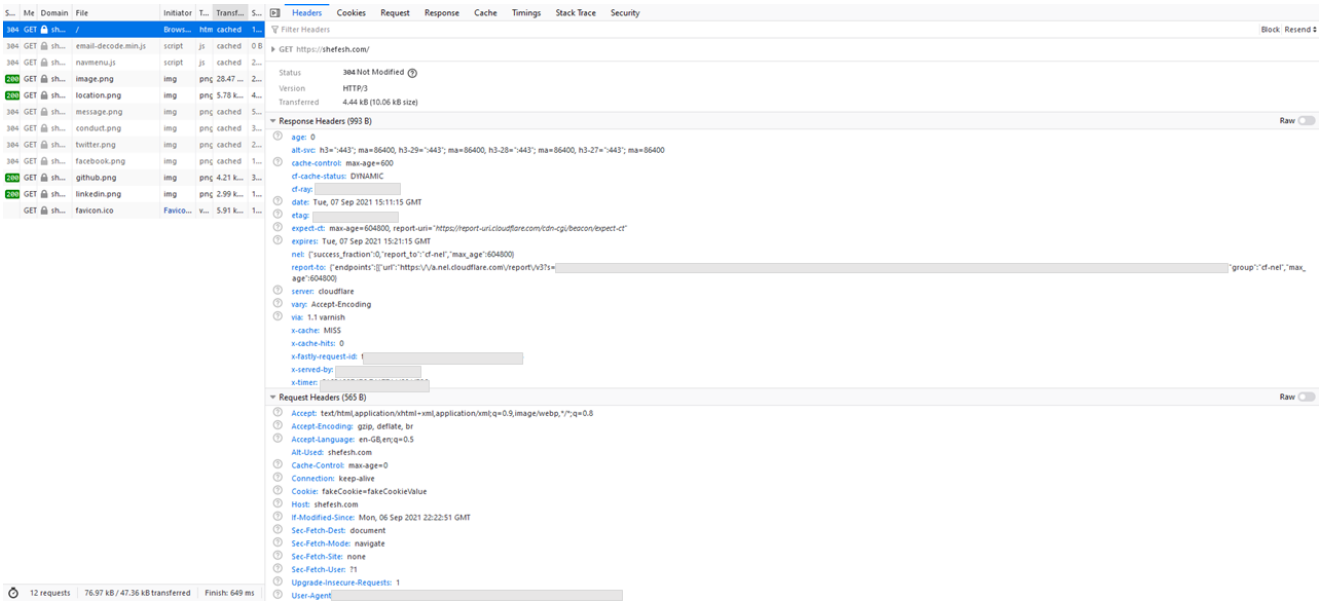
Network

The Network tab is one of the most powerful tools in the Developer tools, and allows us to view, edit, and resend HTTP requests associated with the page.

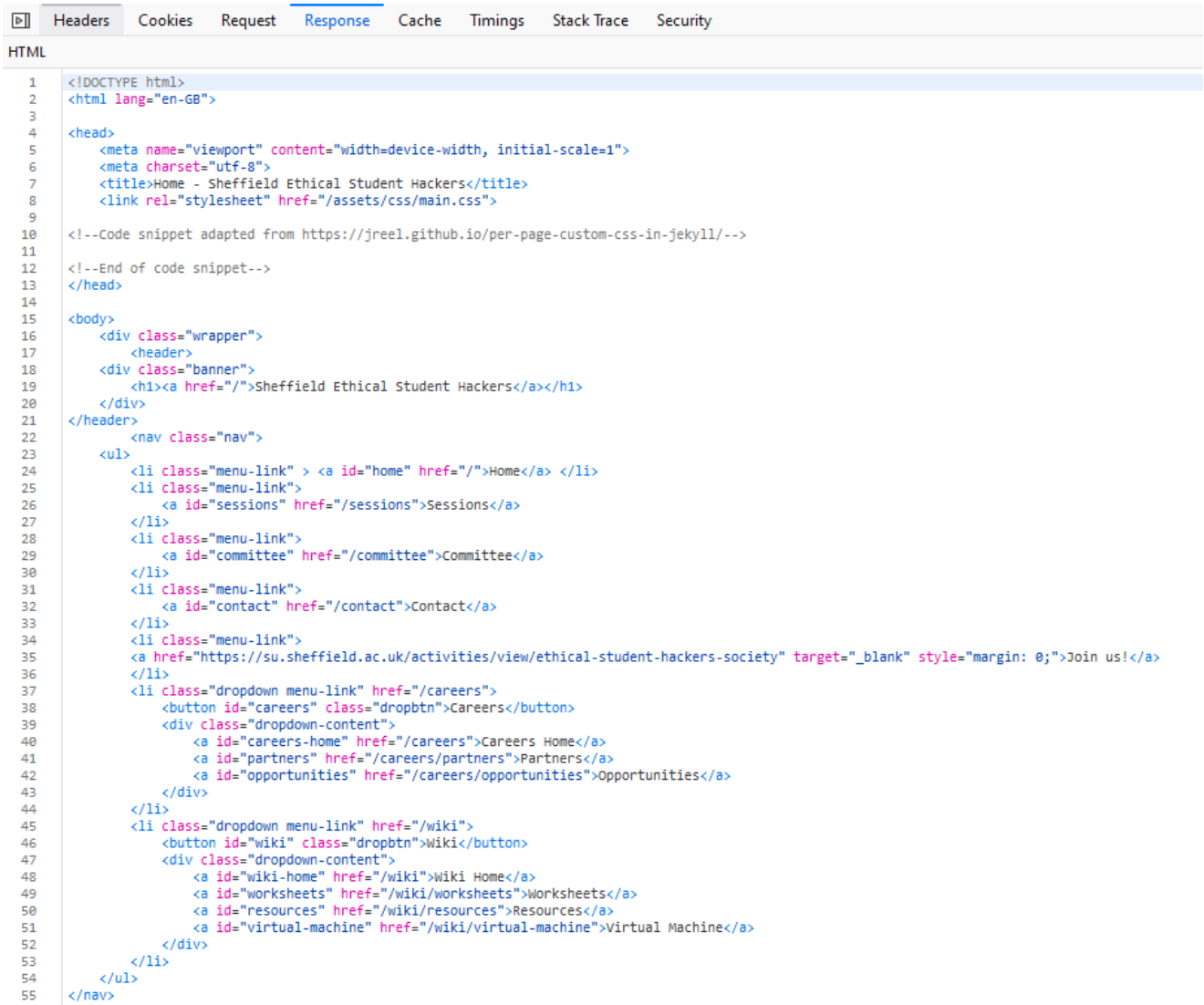


By default it shows the Request Status, Method, Domain, File (i.e. path), and some other information.

Clicking an individual request will show the request and response headers for that request:

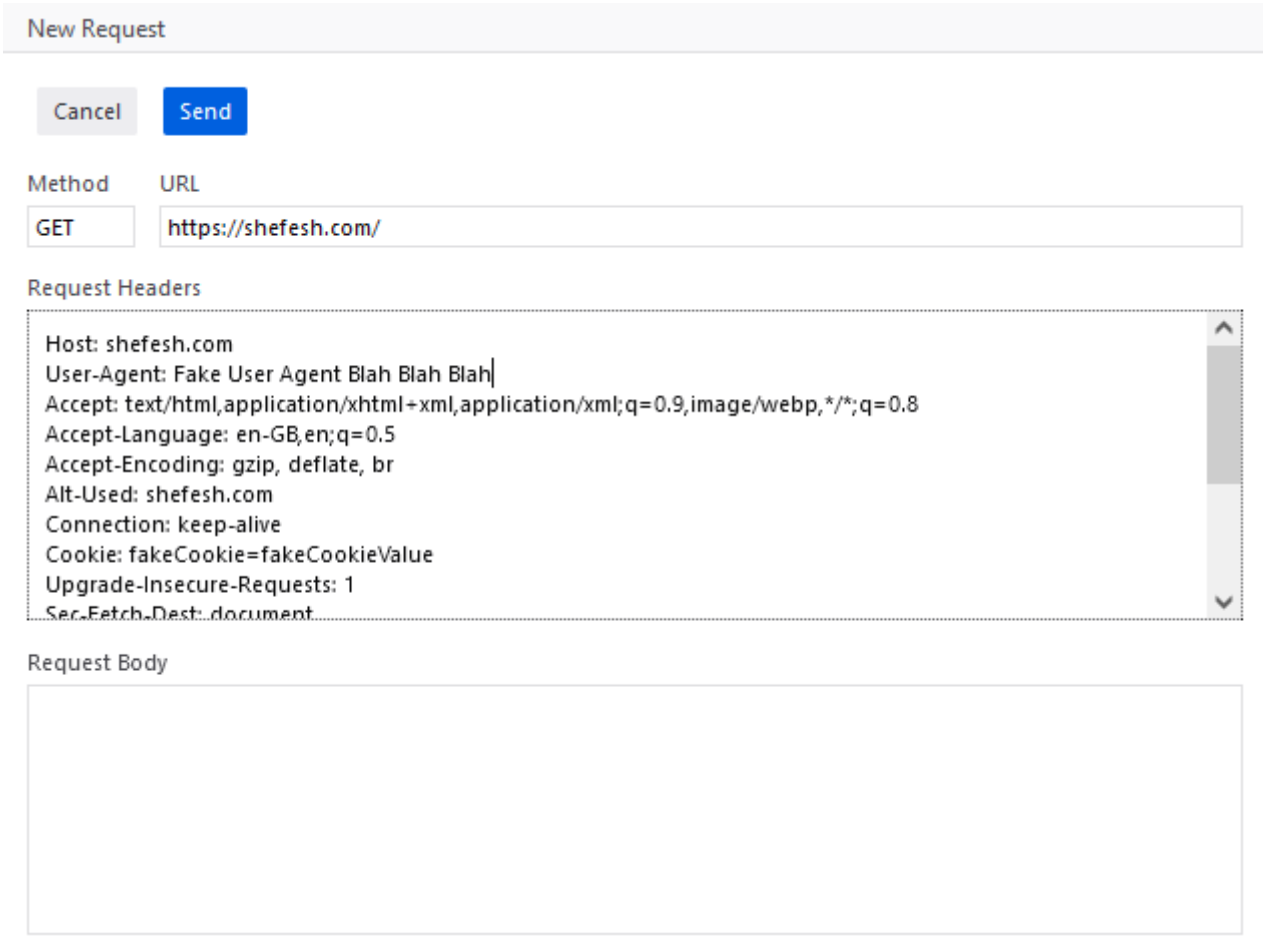


The response can also be viewed:



```
<!DOCTYPE html>
<html lang="en-GB">

<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta charset="utf-8">
    <title>Home - Sheffield Ethical Student Hackers</title>
    <link rel="stylesheet" href="/assets/css/main.css">

<!--Code snippet adapted from https://jreel.github.io/per-page-custom-css-in-jekyll/-->

<!--End of code snippet-->
</head>

<body>
    <div class="wrapper">
        <header>
    <div class="banner">
        <h1><a href="/">Sheffield Ethical Student Hackers</a></h1>
    </div>
</header>
        <nav class="nav">
    <ul>
        <li class="menu-link" > <a id="home" href="/">Home</a> </li>
        <li class="menu-link">
            <a id="sessions" href="/sessions">Sessions</a>
        </li>
        <li class="menu-link">
            <a id="committee" href="/committee">Committee</a>
        </li>
        <li class="menu-link">
            <a id="contact" href="/contact">Contact</a>
        </li>
        <li class="menu-link">
        <a href="https://su.sheffield.ac.uk/activities/view/ethical-student-hackers-society" target="_blank" style="margin: 0;">Join us!</a>
        </li>
        <li class="dropdown menu-link" href="/careers">
            <button id="careers" class="dropbtn">Careers</button>
            <div class="dropdown-content">
                <a id="careers-home" href="/careers">Careers Home</a>
                <a id="partners" href="/careers/partners">Partners</a>
                <a id="opportunities" href="/careers/opportunities">Opportunities</a>
            </div>
        </li>
        <li class="dropdown menu-link" href="/wiki">
            <button id="wiki" class="dropbtn">Wiki</button>
            <div class="dropdown-content">
                <a id="wiki-home" href="/wiki">Wiki Home</a>
                <a id="worksheets" href="/wiki/worksheets">Worksheets</a>
                <a id="resources" href="/wiki/resources">Resources</a>
                <a id="virtual-machine" href="/wiki/virtual-machine">Virtual Machine</a>
            </div>
        </li>
    </ul>
</nav>
```

Finally, clicking `Resend > Edit and Resend` in the top-right allows the request to be modified and resent:



This lets us arbitrarily change data such as the `User-Agent` header.

Going Further

To learn about how to use the Developer Tools, read the docs for each browser:

- https://developer.mozilla.org/en-US/docs/Tools
- https://developer.chrome.com/docs/devtools/

For more about how HTTP requests work, including responses, headers, and request bodies, read the HTTP Requests Fundamental Skill page.

Cheatsheet

View Site Source: `Ctrl + U`

Open Developer Tools: `F12`

Inspect a Specific Element: `Ctrl + Shift + C`

Worksheet

1. Visit https://shefesh.com. Using the techniques we've looked at, what can you find?

2. Visit https://juice-shop.herokuapp.com/#/ and open the Developer Tools. What sort of Javascript Files does it have? What about cookies?