

Inspecting Elements

- Loading Webpages From Servers
- Modifying CSS In Browser
- Viewing Sources
- Local Host

Receiving a Request

When you load a webpage, you are sending a GET request with the url for the webpage.



A server will send to the client local copies of the files necessary to load the webpage.



Browsers will interpret these files into webpages that we can read and interact with.

Editing a Response

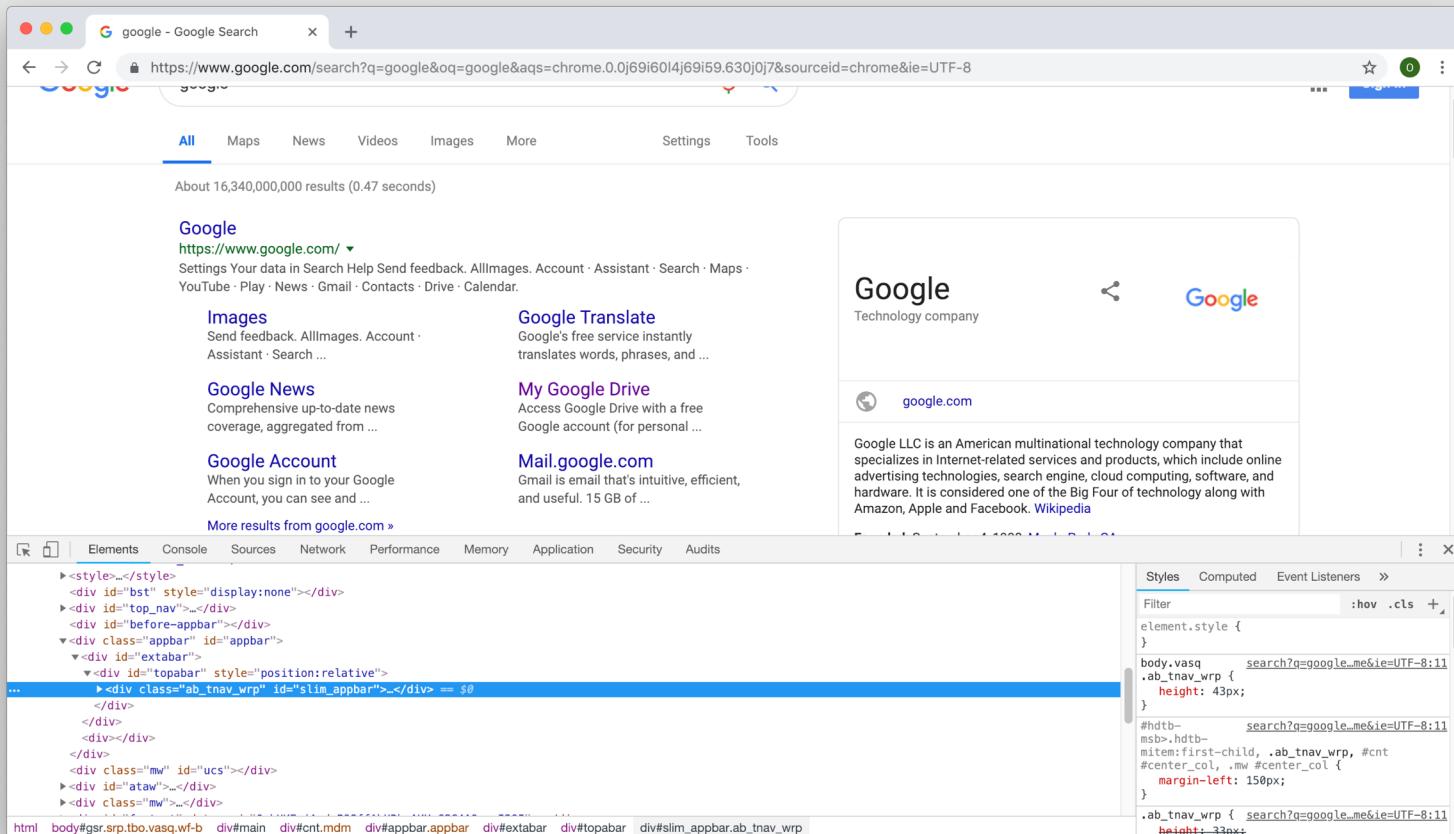
Sometimes we want to edit these files ourselves, so we use inspect elements to do so.



Since the server sent us local copies of the files, editing them won't edit the original files stored on the server.

Inspecting Elements

Inspect Elements is a prompt that allows you to show and edit the files used to load your webpage.



Inspecting Elements

Inspect Elements gives us direct access to these files to edit locally.

The screenshot shows the Chrome DevTools interface with the 'Elements' tab selected. The left panel displays the HTML structure of a page, with a red box highlighting the element `<div id="hdtbSum">`. The right panel shows the associated CSS styles for that element, with a blue box highlighting the rule `#hdtbSum { height: 58px; }`.

HTML side

```
<div id="subform_ctrl"></div>
▶<style>.</style>
<div id="bst" style="display:none"></div>
▼<div id="top_nav">
  ▼<div jscontroller="qik19b" jsdata="Z1JpA;;BB5JqM" jsaction="rcuQ6b:npT2md">
    <h1 class="bNg8Rb">Search Modes</h1>
    ▼<div class="hdtbna notl" id="hdtb" role="navigation" tabindex="0">
      ..<div id="hdtbSum">..</div> == $0
      ▶<div class="hdtb-td-c hdtb-td-h" aria-expanded="false" id="hdtbMenus" tabindex="0" data-ved="0ahUKEwjrobyu-8ffAhWLo1kKHX0vCGsQ3B8IIA">..</div>
      </div>
      </div>
    </div>
    <div id="before-appbar"></div>
    ▶<div class="appbar" id="appbar">..</div>
      <div class="mw" id="ucs"></div>
  html body#qsr.srp.tbo.vasq.wf-b div#main div#cnt.mdm div#top_nav div div#hdtb.hdtbna.notl div#hdtbSum
```

CSS side

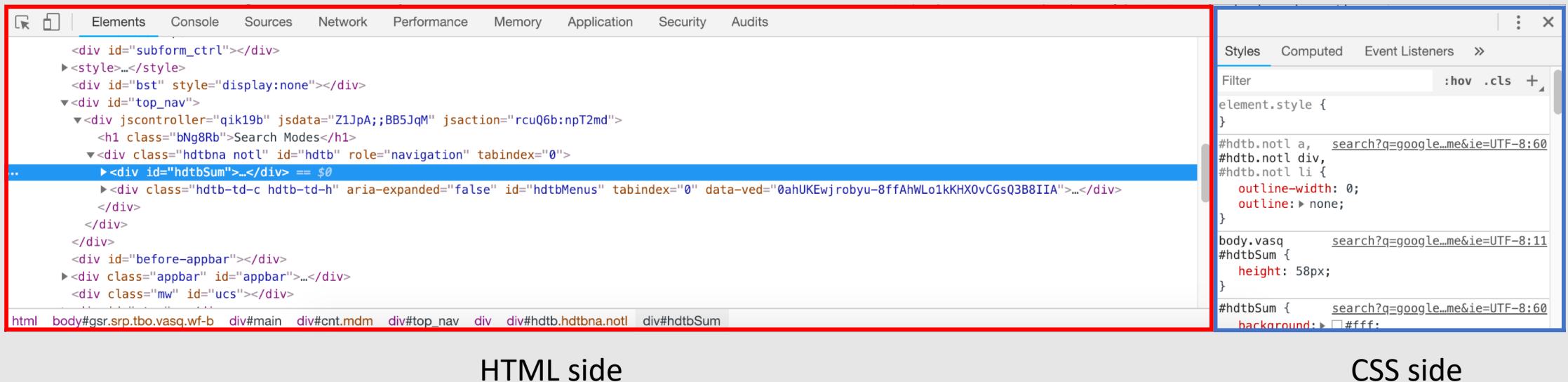
```
Styles Computed Event Listeners »
Filter :hov .cls +
element.style {
}
#hdtb.notl a, search?q=google...me&ie=UTF-8:60
#hdtb.notl div,
#hdtb.notl li {
  outline-width: 0;
  outline: ▶ none;
}
body.vasq search?q=google...me&ie=UTF-8:11
#hdtbSum {
  height: 58px;
}
#hdtbSum { search?q=google...me&ie=UTF-8:60
  background: □ #fff;
```

• ~~Loading Webpages From Servers~~

- Modifying CSS In Browser
- Viewing Sources
- Local Host

CSS In Browser

We can use the CSS tab to experiment and debug CSS properties directly in browser.



HTML side

CSS side

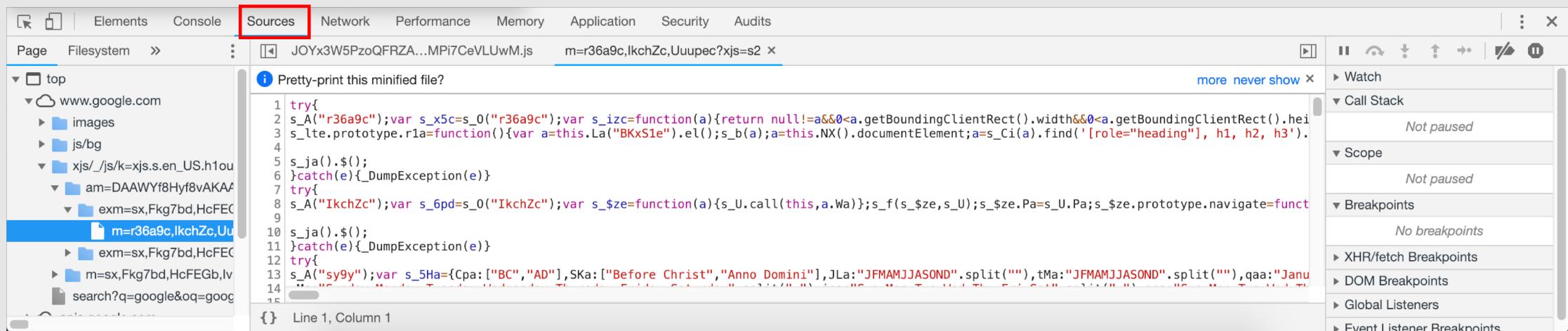
When an HTML element is selected, all relevant CSS classes and IDs will be displayed on the CSS side.

Depending on the browser, we can toggle between CSS properties in a given class or ID. Changing these properties will change it for all other elements that share the same class or ID.

- Loading Webpages From Servers
- Modifying CSS In Browser
- Viewing Sources
- Local Host

Viewing Sources

The Sources tab lists all the files retrieved from the server to load the webpage.

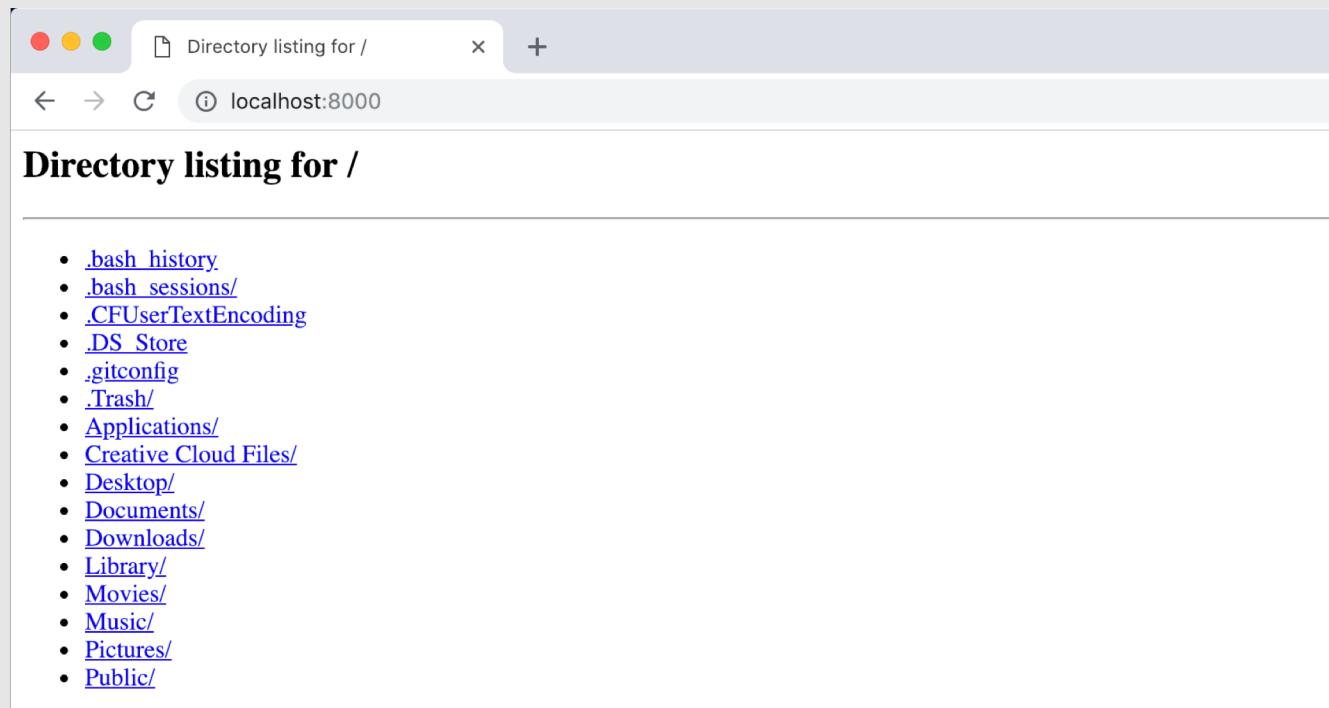


Allows users to view and edit original HTML, CSS, and JavaScript files along with images, texts, SQL, and other files.

- Loading Webpages From Servers
- Modifying CSS In Browser
- Viewing Sources
- Local Host

Local Host

As we will learn later with JSON Parsing, not everything can be loaded directly from clicking on index.html. Sometimes we need to create a local host to give our files and address so that our HTML and JavaScript files can refer to them.



Creating a Local Host

Make sure you have Python installed. In a command prompt, type the following:

Python 2.7

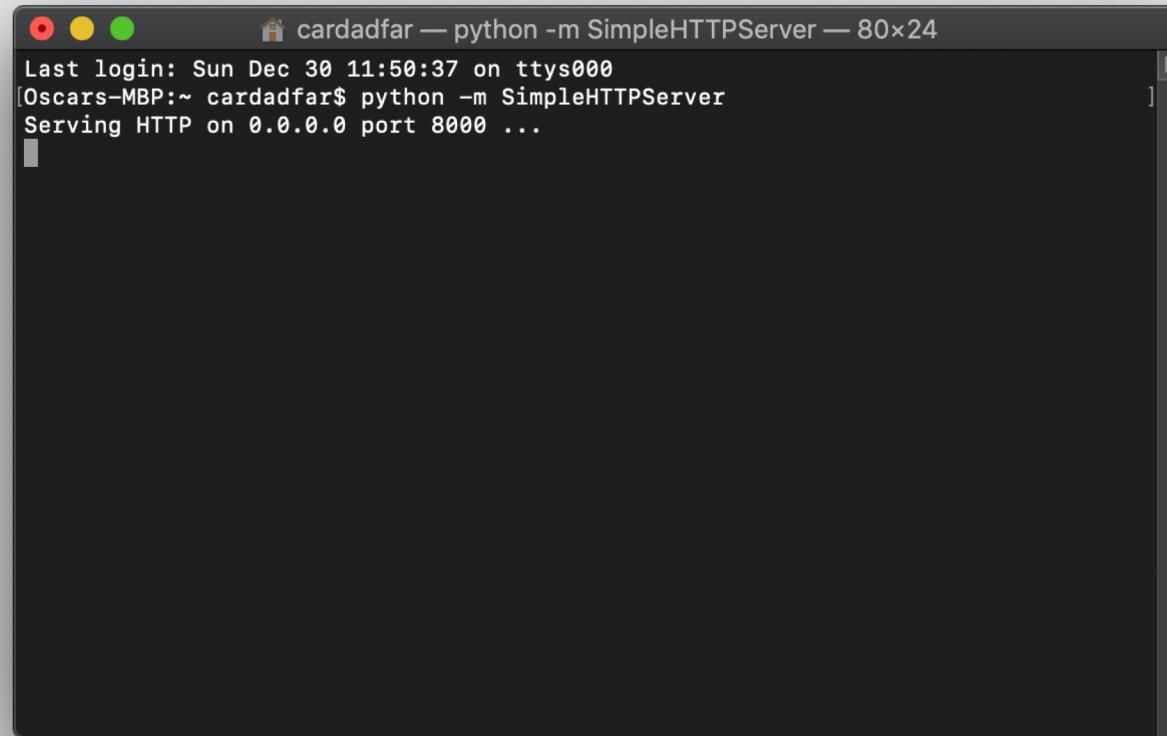
```
python -m SimpleHTTPServer
```

Python 3.6

```
python -m http.server
```

Creating a Local Host

You will get a response that a connection has been opened on a port (often 8000).

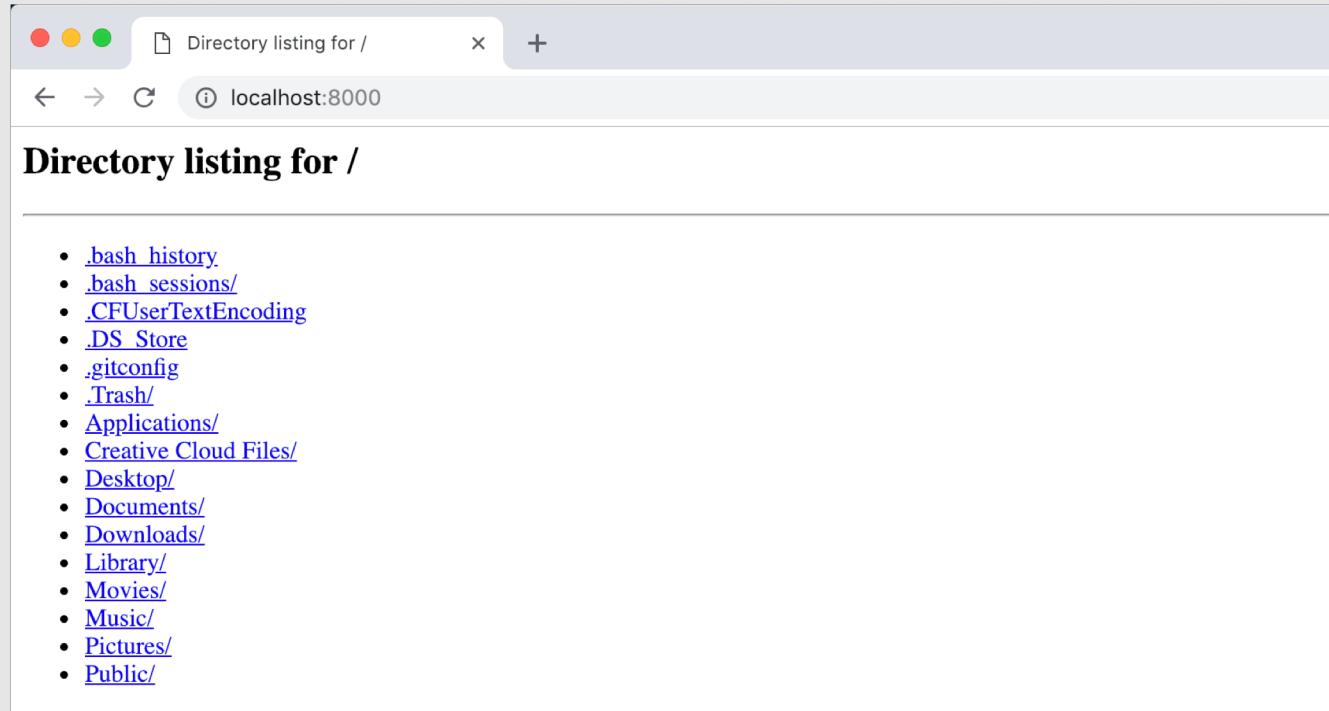
A screenshot of a terminal window titled "cardadfar — python -m SimpleHTTPServer — 80x24". The window shows the following text:

```
Last login: Sun Dec 30 11:50:37 on ttys000
[Oscars-MBP:~ cardadfar$ python -m SimpleHTTPServer
Serving HTTP on 0.0.0.0 port 8000 ...
```

The terminal has a dark background and light-colored text. The title bar includes the user name "cardadfar", the command "python -m SimpleHTTPServer", and the window size "80x24". The window has standard OS X-style red, yellow, and green close buttons in the top-left corner.

Creating a Local Host

Using a browser, go to `localhost:portnumber`



The directory you open your local host in will be the directory displayed by your local host

Homework Ideas

- ❑ Use Inspect Elements to debug any issues that you may be experiencing.
- ❑ Use Inspect Elements to quickly test out new CSS features live for faster results.
- ❑ Add additional content/styling to site.

Live Demo