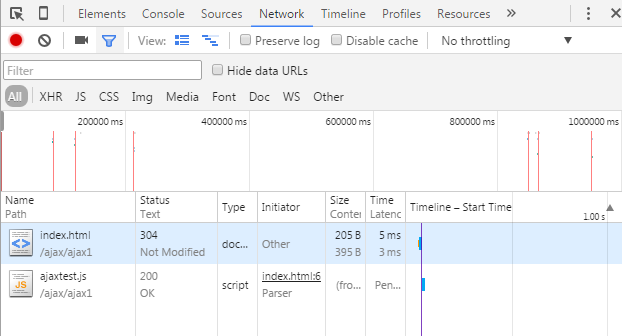
# Web Development 2.1 - Lab6 – Intro to Ajax

**Part 1**

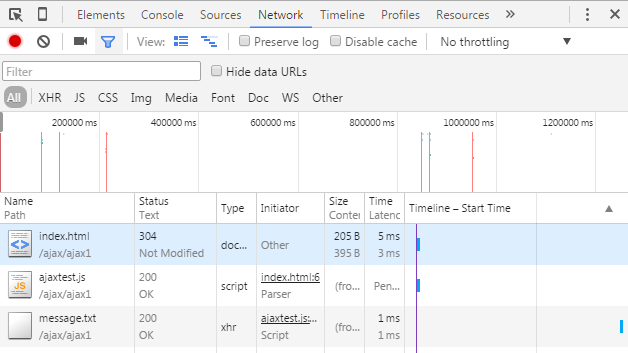
1. Preview the 3 files in /ajax1/
   1. Index.html
   2. Ajaxtest.js
   3. Message.txt
2. Start ‘Visual Studio Code’ (VS Code). VS Code has a built-in web server extension that we’ll use to run the Ajax examples. We need a webserver for the XMLHttpRequest to work.
3. Search for and download the ‘Live Server’ extension for VS Code and then install it.
4. Open the /ajax1/ folder in VS Code (Open Folder)
5. Right-click in the index.html file and choose ‘Open with Live Server’. You can view the webpage at <http://127.0.0.1:5500/index.html>. It’s best to use Chrome to view the file.
6. Click on the link to display the contents of the file
7. If Ajax is working correctly, you will see a **prompt** which displays the text in message.txt
8. Review the files to understand how it’s working.

**Part 2**

1. Open the web-inspector and reload the page.
2. Use the inspector (Network Tab) to view the http requests and responses that are happening
   1. After the page first loads

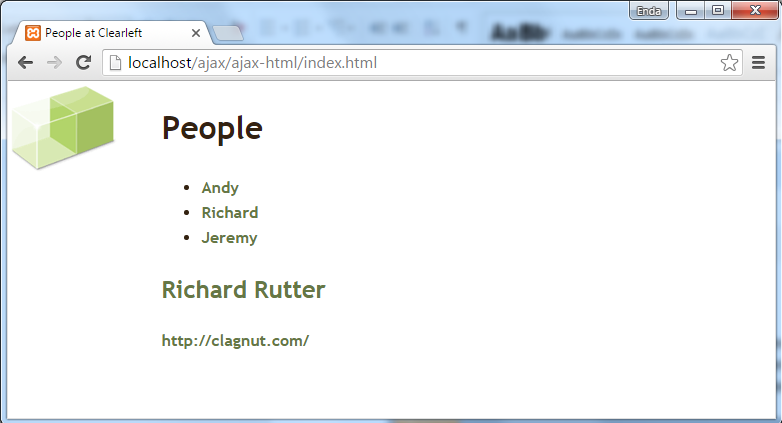


* 1. After you click on the link

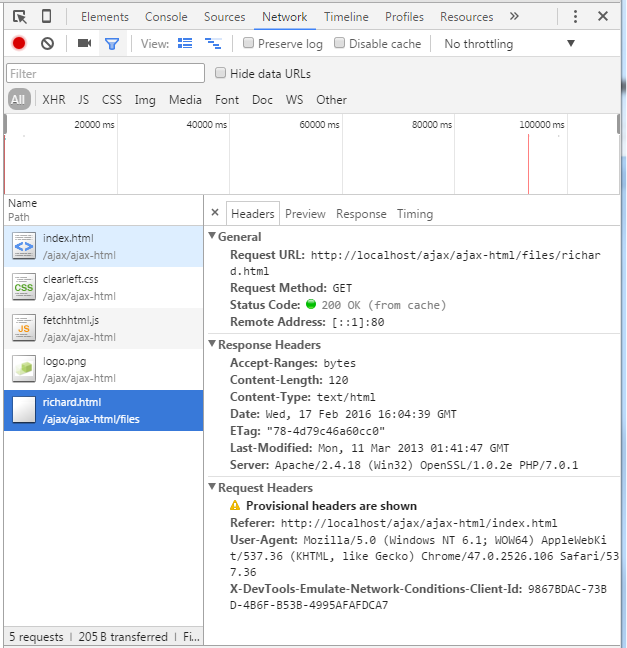


**Part 3**

1. Preview all files in the /ajax-html/ folder, including the html files in the /files/ folder.
2. Open the /ajax-html/ folder in VS Code and open with the live server as before. Use Chrome to test the links. The page should update with the appropriate links if everything is working properly:



1. Using the web-inspector, view what’s happening at each stage in the communication



**Part 3**

1. Write an identical application to Part 2 that uses XML files for each person (Andy, Richard and Jeremy).

e.g.

<?xml version="1.0" encoding="utf-8"?>

<details>

<name>….</name>

<website>…</website>

<email>…</email>

</details>

1. Save the application in \ajax-xml\ folder. Most of the html and css are the same as the previous example. The JavaScript file will need more changes.
2. Use the notes to figure out how the JavaScript code will parse the information from the xml files.
3. Use DOM methods to create all the elements you need to add to the page (i.e. document.createElement(), document.createTextNode() etc.)

Note: You should ***not*** use the innerHTML property for this lab.

**Part 4**

1. Write an identical application to Part 2 that uses JSON files for each person (Andy, Richard and Jeremy).

e.g.

{"person": {

"name":"Andy Budd",

"website":"http://andybudd.com/",

"email":"andy@clearleft.com"

}

}

Save the application in \ajax-json\

1. Use the notes to figure out how the JavaScript code will parse the information from the JSON files.
2. In this part, use the innerHTML property (***not*** DOM methods) to add the data to the page