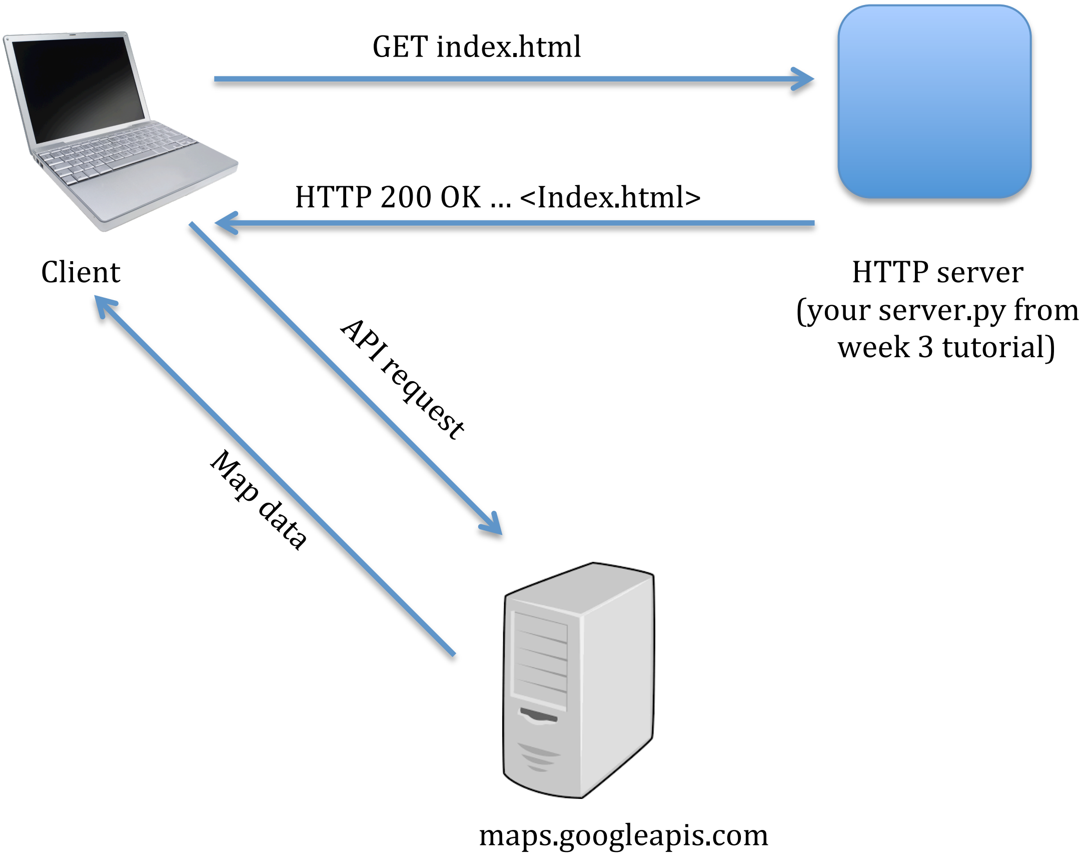
159352 Tutorial – Week 4

In this exercise we will explore how dynamic content is generated on the client as well as the server side.

Part A – Client side

JavaScript enables the client to generate/render dynamic content. The index.html has embedded JavaScript, which talks to the Google Maps API and dynamically renders a user requested location on the map.



Download the index.html file from Stream, and examine the JavaScript. Quite often you will require authentication (API key) to access web services.

Q1) Identify the line of code where you will enter the API key

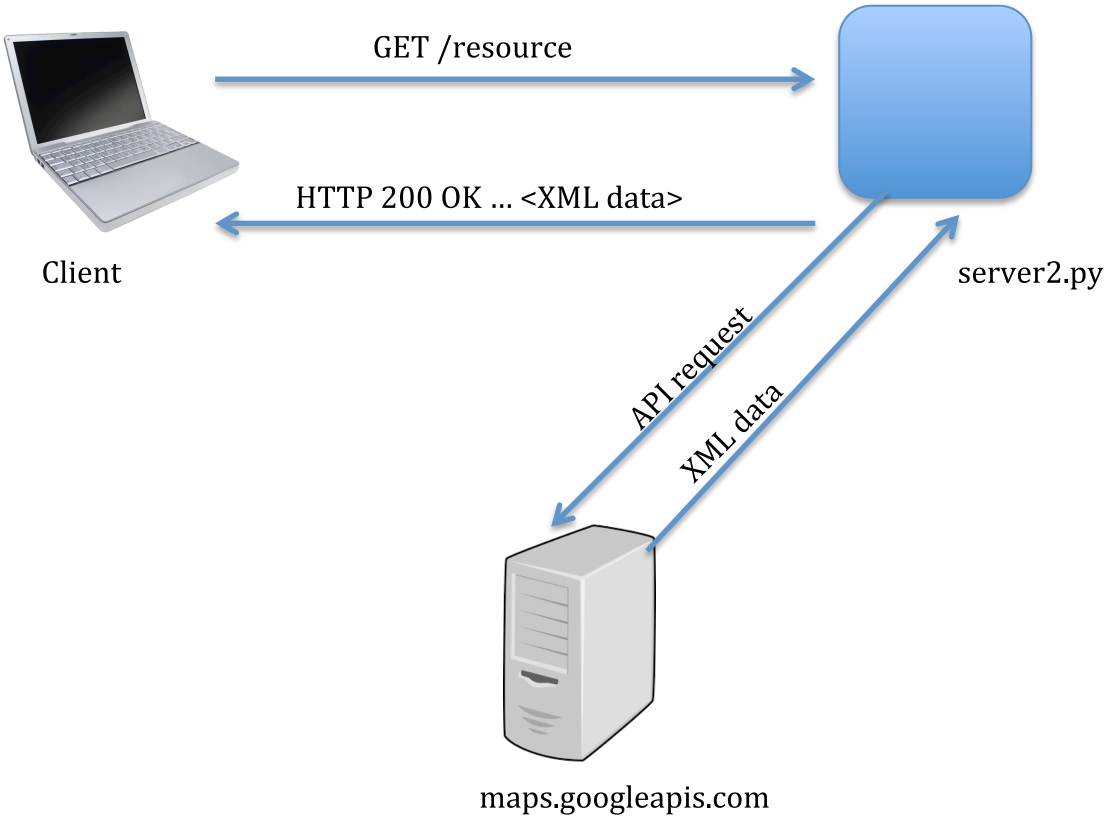
See the JavaScript in action by opening the index.html in any browser, or using your web server (from week 3 tutorial) to serve it to your browser.

Inspect the header information using developer tools in your browser.

Q2) Which HTTP method is used by the browser to pass parameters to the Google Maps API?

Part B – Server side

The minimalistic HTTP server (from tutorial 3) has been extended to serve dynamic content. The server2.py listens on port 8080, and passes the client request (location name) to Google maps API. The resulting XML data about location data is then relayed back to the client as shown below.



Download the server2.py from Stream and study the code. The server uses PycURL so you may need to install it if your system does not have it already. Curl is a powerful command line tool for transferring data with URLs over many protocols. Read the documentation about its usage. (https://curl.haxx.se)

Execute the server2.py program, open your browser (preferably chrome) and type http://servername:8080/location
e.g. http://localhost:8080/auckland

Q3) What is the root element in the XML response received by your browser?

Q4) Is there any element with an attribute?

Q5) Name any two elements that are considered siblings in the XML tree

Your task is to modify the server2.py code so that it can parse the XML data and send a simpler message to the client e.g.

Auckland is located at Latitude -36.8484597 and Longitude 174.7633315

Optional exercise – Create a mashup web service by extending the server functionality to talk to other API providers. Example you can use the latitude and longitude data to show pictures tagged at that location using Flickr API. https://www.programmableweb.com provides a comprehensive listing of APIs at your disposable. Generally the API providers require you to register (mostly for free) in order to use the services.

Submit the answers to the question (1-5) and your server code on Stream.