LING 408/508: Computational Techniques for Linguists

Lecture 22

Today's Topics

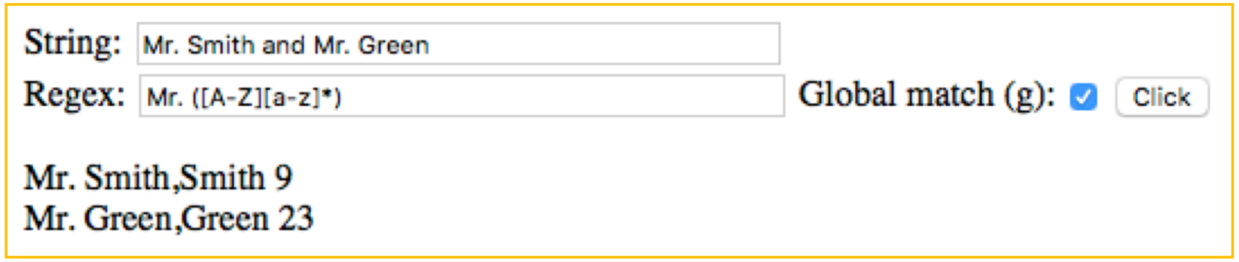
- Some remaining notes on Javascript regex

This week's milestone: a webserver

- Homework 8: set up your own webserver

Javascript Regexp Tester

- Useful property
  - `regex.lastIndex`

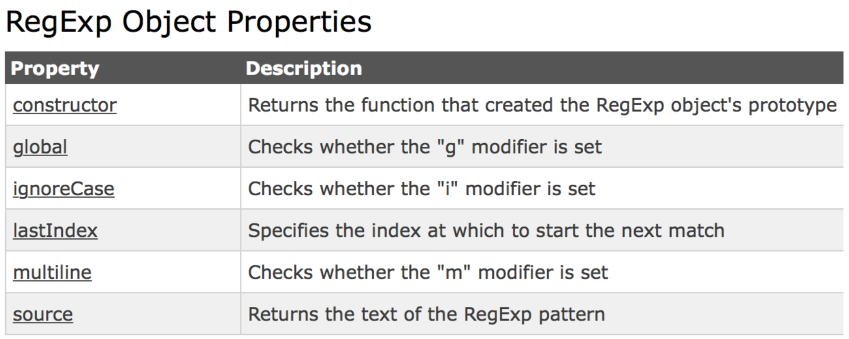


String: Mr. Smith and Mr. Green

Regex: Mr. ([A-Z][a-z]*)

Global match (g): ✓

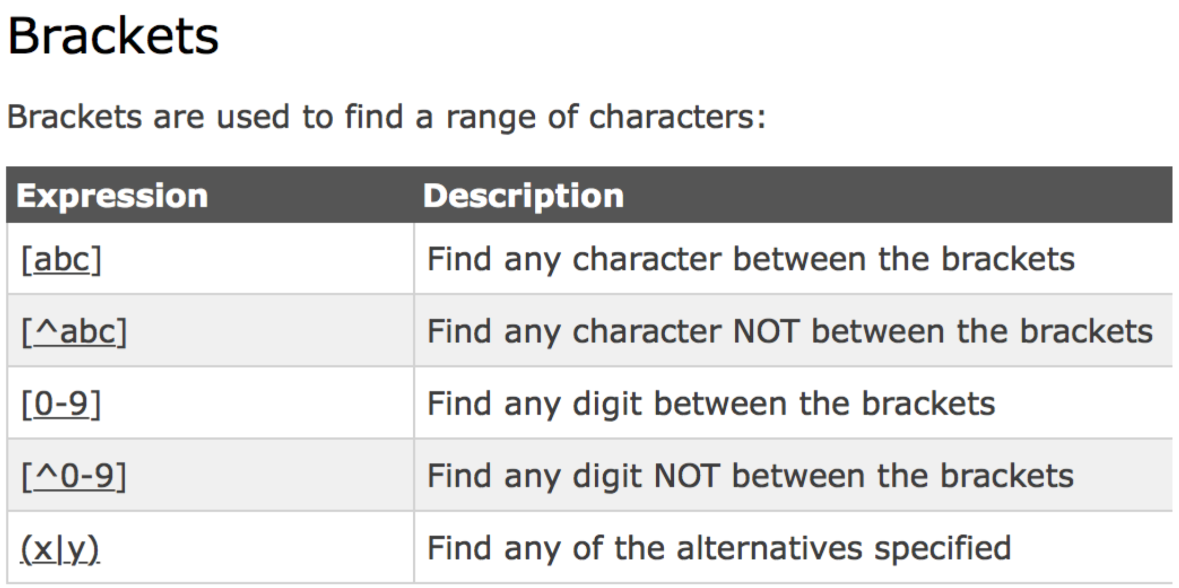
```
Mr. Smith,Smith 9
Mr. Green,Green 23
```



RegExp Object Properties

Property	Description
constructor	Returns the function that created the RegExp object's prototype
global	Checks whether the "g" modifier is set
ignoreCase	Checks whether the "i" modifier is set
lastIndex	Specifies the index at which to start the next match
multiline	Checks whether the "m" modifier is set
source	Returns the text of the RegExp pattern

Regular expression syntax



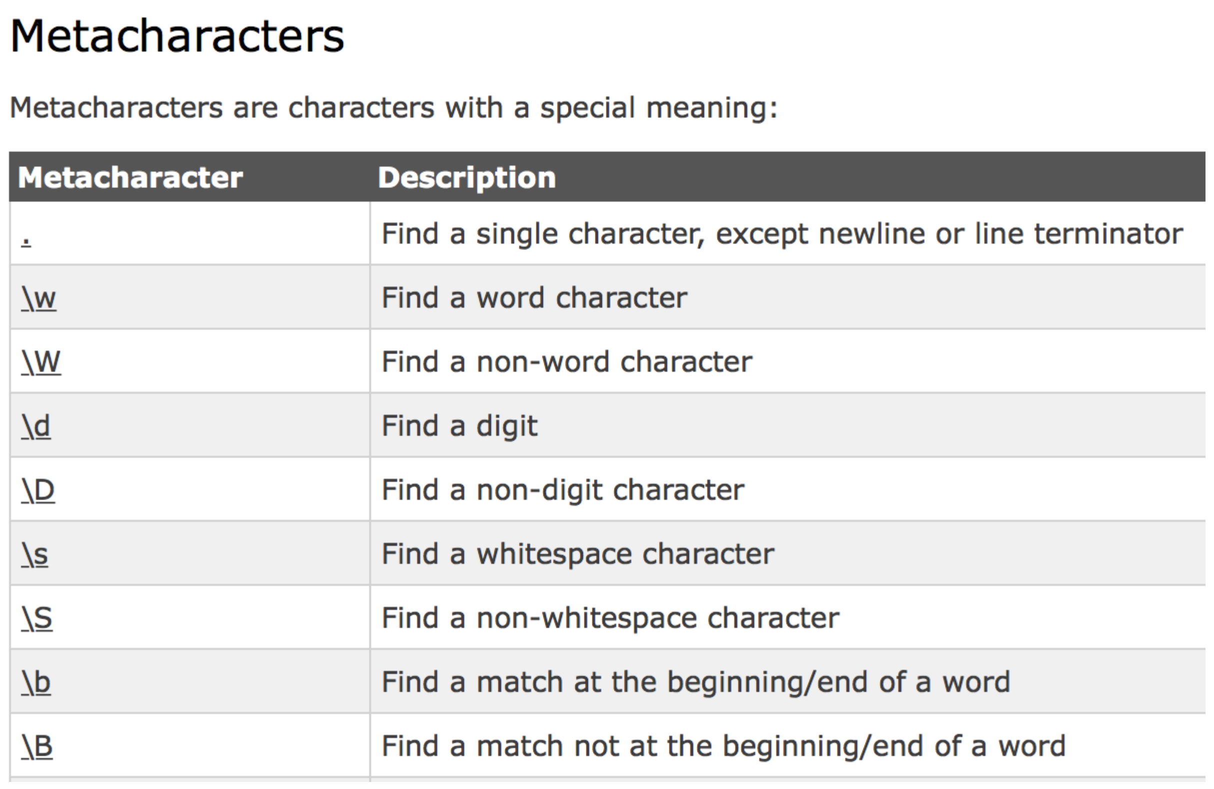
Brackets

Brackets are used to find a range of characters:

Expression	Description
[abc]	Find any character between the brackets
[^abc]	Find any character NOT between the brackets
[0-9]	Find any digit between the brackets
[^0-9]	Find any digit NOT between the brackets
(x|y)	Find any of the alternatives specified

http://www.w3schools.com/jsref/jsref_obj_regexp.asp

Regular expression syntax

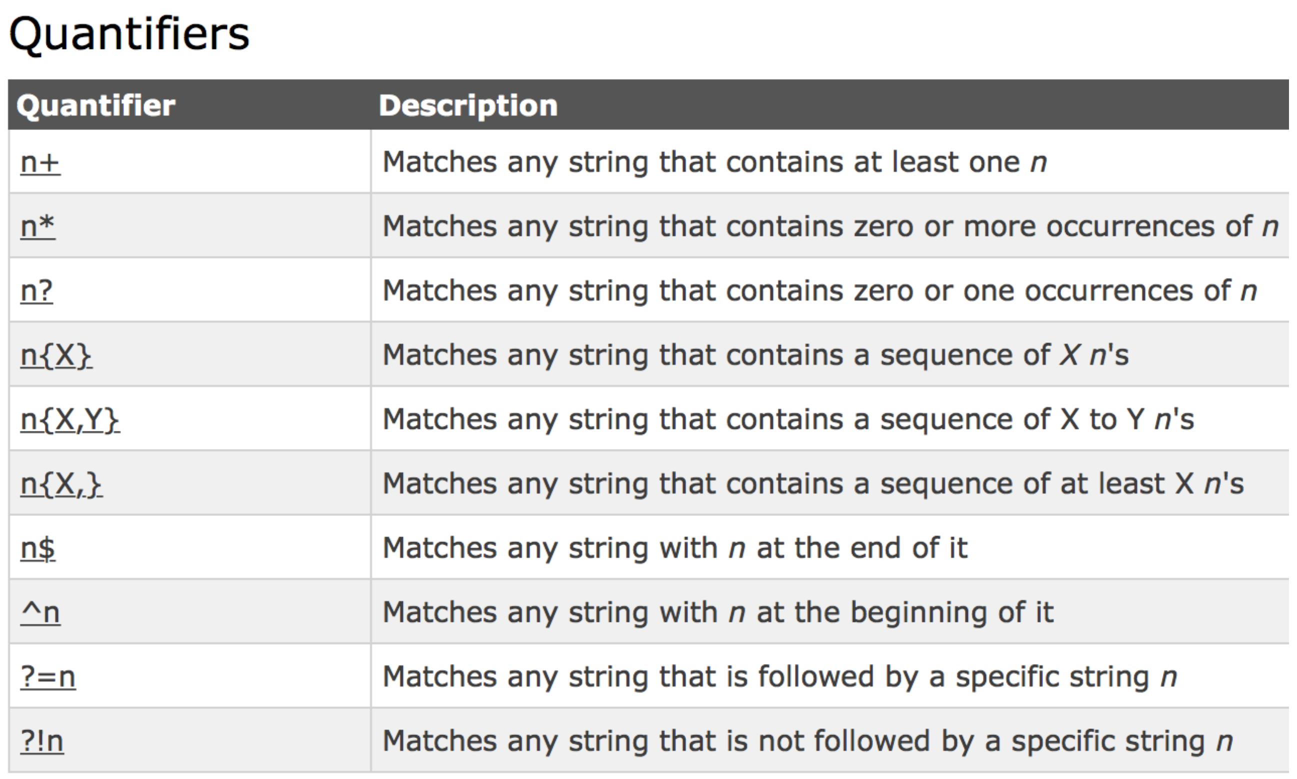


Metacharacters

Metacharacters are characters with a special meaning:

Metacharacter	Description
.	Find a single character, except newline or line terminator
\w	Find a word character
\W	Find a non-word character
\d	Find a digit
\D	Find a non-digit character
\s	Find a whitespace character
\S	Find a non-whitespace character
\b	Find a match at the beginning/end of a word
\B	Find a match not at the beginning/end of a word

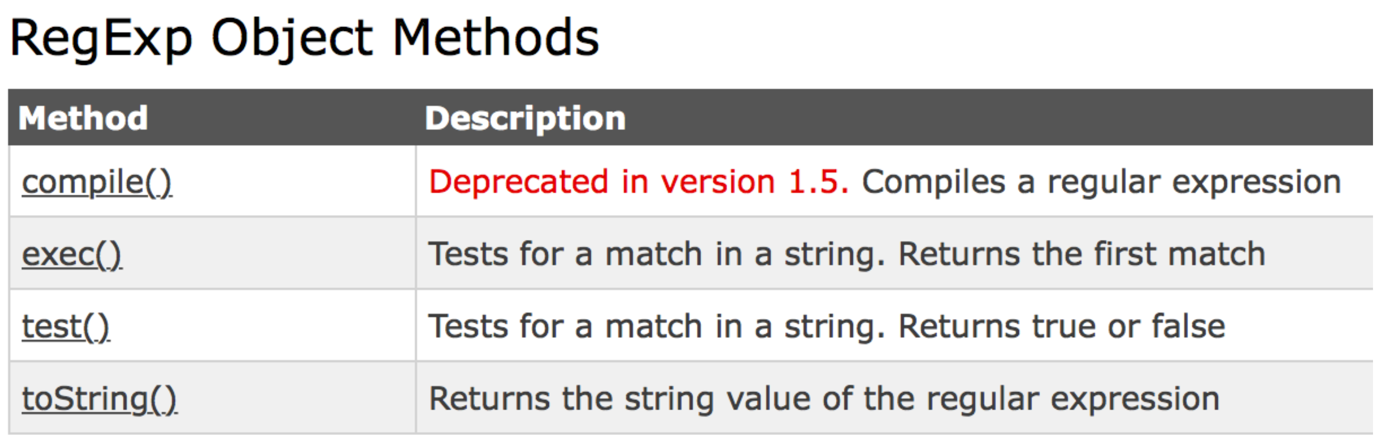
Regular expression syntax



Quantifiers

Quantifier	Description
n+	Matches any string that contains at least one *n*
n*	Matches any string that contains zero or more occurrences of *n*
n?	Matches any string that contains zero or one occurrences of *n*
n{X}	Matches any string that contains a sequence of *X* *n*'s
n{X,Y}	Matches any string that contains a sequence of X to Y *n*'s
n{X,}	Matches any string that contains a sequence of at least X *n*'s
n$	Matches any string with *n* at the end of it
^n	Matches any string with *n* at the beginning of it
?=n	Matches any string that is followed by a specific string *n*
?!n	Matches any string that is not followed by a specific string *n*

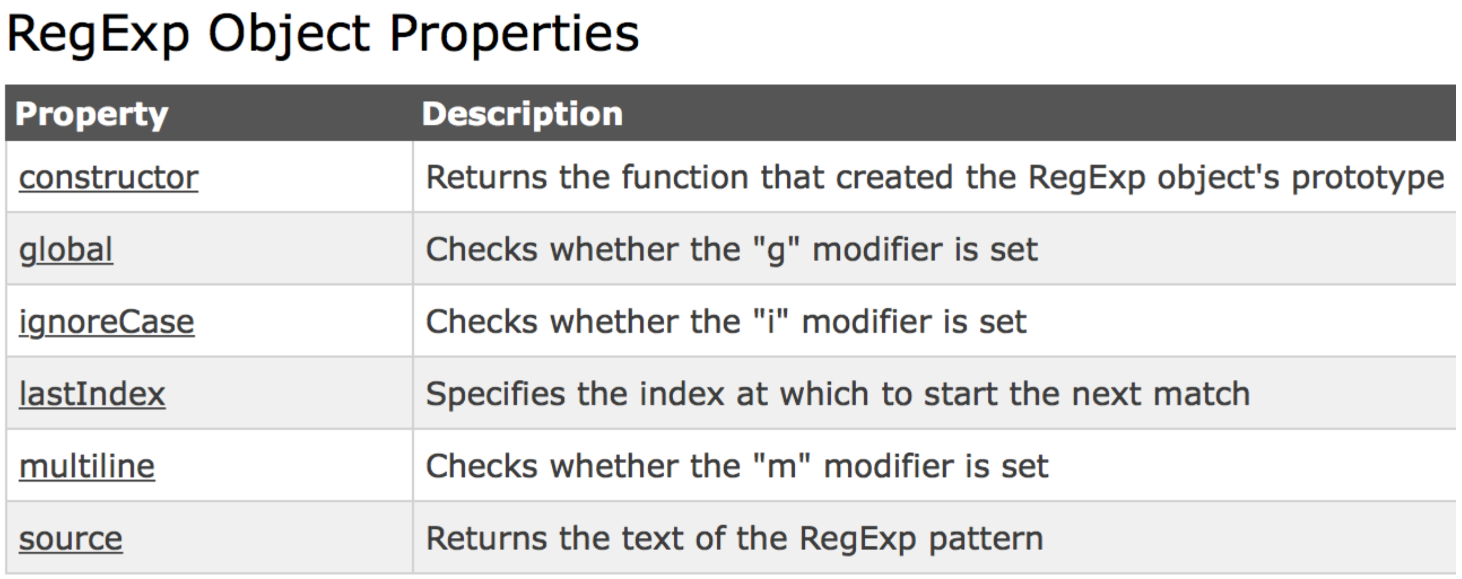
Regular expression syntax



RegExp Object Methods

Method	Description
compile()	Deprecated in version 1.5. Compiles a regular expression
exec()	Tests for a match in a string. Returns the first match
test()	Tests for a match in a string. Returns true or false
toString()	Returns the string value of the regular expression

Regular expression syntax

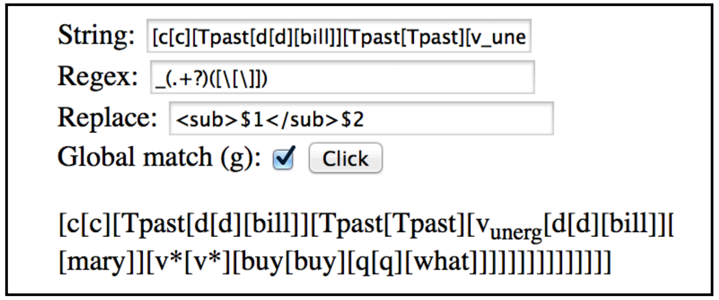


RegExp Object Properties

Property	Description
constructor	Returns the function that created the RegExp object's prototype
global	Checks whether the "g" modifier is set
ignoreCase	Checks whether the "i" modifier is set
lastIndex	Specifies the index at which to start the next match
multiline	Checks whether the "m" modifier is set
source	Returns the text of the RegExp pattern

Regex Replace

- We'll also need the method replace():
    - var regex = new RegExp(*re_s*,*flag_s*);
    - modified_string = string.replace(regex,*replacement*)
  - replacement string can contain $*n* (NOT \n)
  - (*n* = group number)



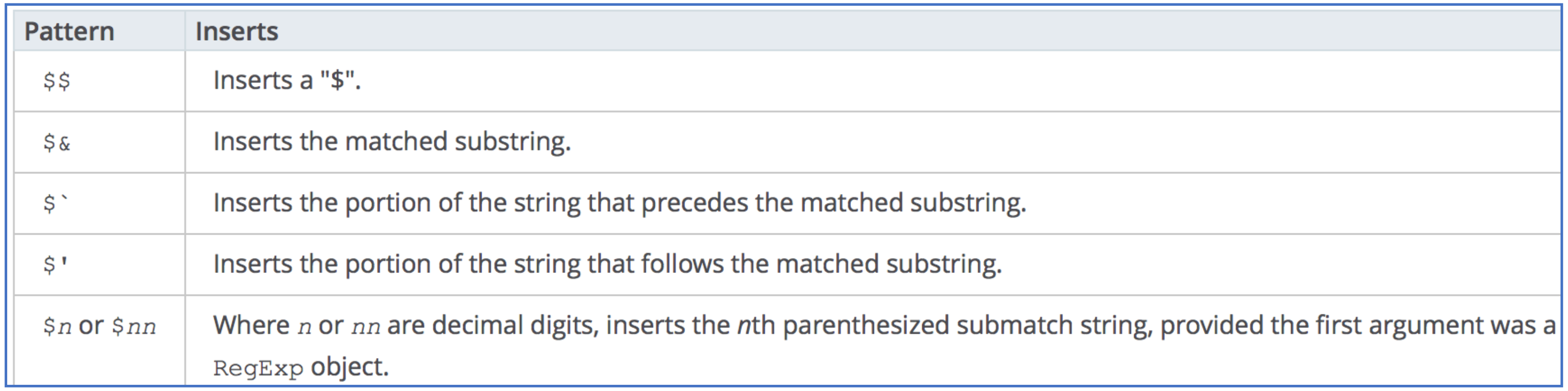
String: `[c[c][Tpast[d[d][bill]][Tpast[Tpast][v_une`

Regex: `_(.+?)([\[\]])`

Replace: `<sub>$1</sub>$2`

Global match (g): ✓ Click

[c[c][Tpast[d[d][bill]][Tpast[Tpast][v$_{unerg}$[d[d][bill]][
[mary]][v*[v*][buy[buy][q[q][what]]]]]]]]]]]]]



Pattern	Inserts
`$$`	Inserts a "$".
`$&`	Inserts the matched substring.
`` $` ``	Inserts the portion of the string that precedes the matched substring.
`$'`	Inserts the portion of the string that follows the matched substring.
`$n` or `$nn`	Where *n* or *nn* are decimal digits, inserts the *n*th parenthesized submatch string, provided the first argument was a `RegExp` object.

developer.mozilla.org

Regex Replace

- Let's try it out:
    - http://elmo.sbs.arizona.edu/~sandiway/ling508-20/rep-test.html

The server side

- So far, all the web programming has been **client-side** only
  - i.e. the Javascript code is running on the browser
- Let's build a webserver
  - the client-side will send form information to the **server-side** to be processed

Building a Webserver

- We'll use `cgi-bin` and `bash` scripts initially …

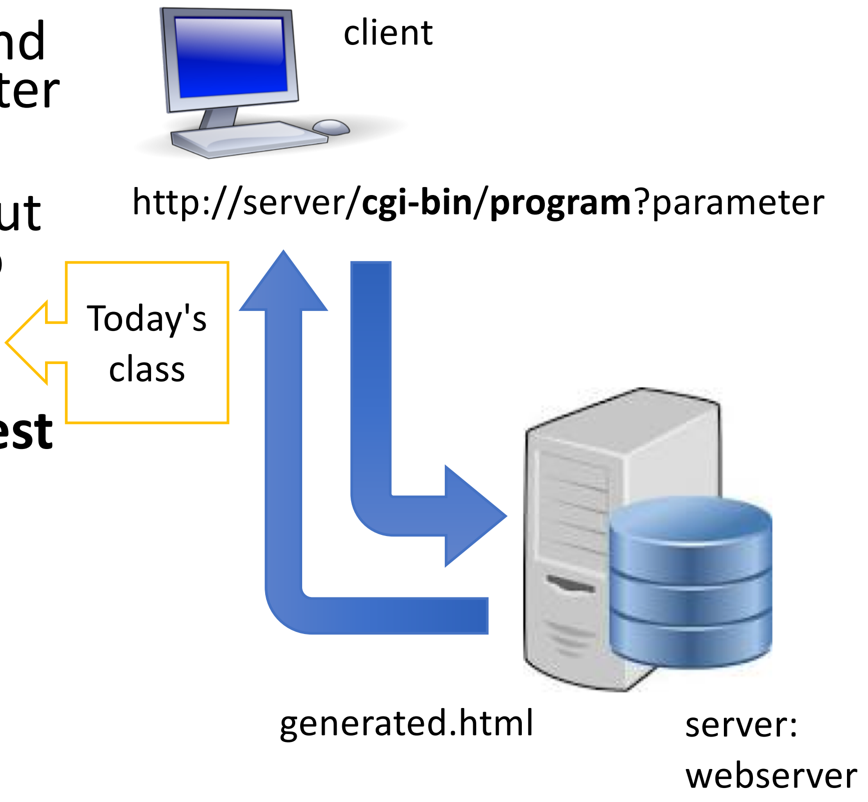
- Apache2 is the most common webserver software
  - *unfortunately, configuration are similar but different on OSX and Ubuntu*

  (we will cover both today)

Common Gateway Interface (CGI)

- The glue between a webserver and programs that run on the computer (= server) hosting the webserver

1. Normally, a webserver sends out **static webpages** in response to (URL) requests from a client (your web browser).
2. Sometimes, we want the **request to run a program** (a script or binary) on the server that does some computation and generates some result to be displayed on the client (as a webpage).



Apache Webserver on OSX

Commands to be entered at a Terminal

- Apache version (OSX 10.13 *High Sierra*):

```
• ~$ httpd –v
• Server version: Apache/2.4.33 (Unix)
• Server built:   Apr  3 2018 17:54:07
• ~$ which httpd
• /usr/sbin/httpd
```

Apache 2.4

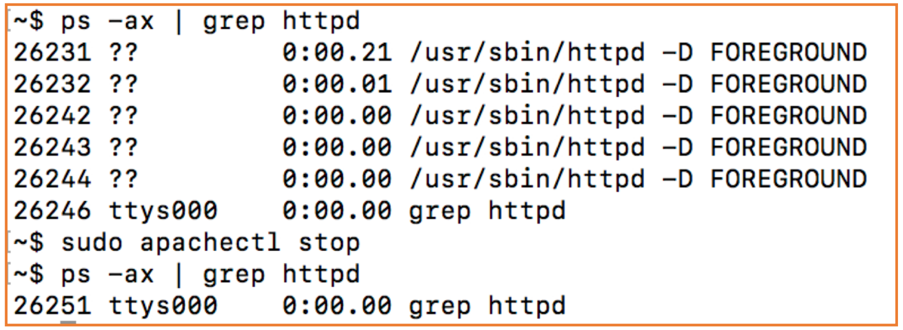
- Apache version (OSX 10.15 *Catalina*):

```
• ~$ httpd –v
• Server version: Apache/2.4.41 (Unix)
• Server built:   Jun  5 2020 23:42:06
```

Apache Webserver on OSX

Commands to be entered at a Terminal

- Apache webserver control:
  - **`~$ which apachectl`**
  - `/usr/sbin/apachectl`
  - **`sudo apachectl start`**
  - **`sudo apachectl stop`**
  - **`sudo apachectl –k restart`** (after configuration change)
  - **`apachectl configtest`** (check configuration)
  - `Syntax OK`
  - **`ps –ax | grep httpd`**
  - **`sudo apachectl stop`**
  - **`ps –ax | grep httpd`**



```
~$ ps -ax | grep httpd
26231 ??         0:00.21 /usr/sbin/httpd -D FOREGROUND
26232 ??         0:00.01 /usr/sbin/httpd -D FOREGROUND
26242 ??         0:00.00 /usr/sbin/httpd -D FOREGROUND
26243 ??         0:00.00 /usr/sbin/httpd -D FOREGROUND
26244 ??         0:00.00 /usr/sbin/httpd -D FOREGROUND
26246 ttys000    0:00.00 grep httpd
~$ sudo apachectl stop
~$ ps -ax | grep httpd
26251 ttys000    0:00.00 grep httpd
```

Apache Webserver on OSX

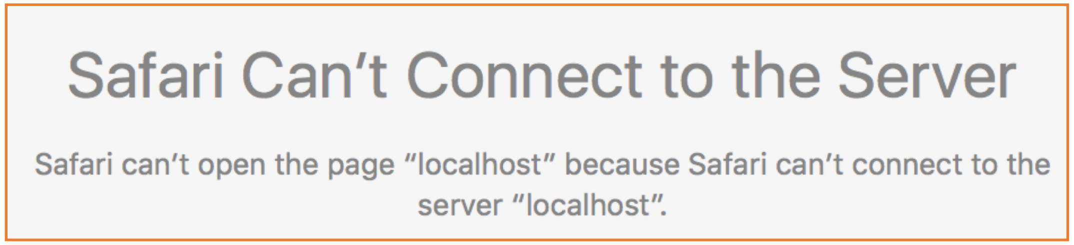
```
~$ apachectl configtest
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using Sandiways-MacBook-4.local. Set the 'ServerName' directive globally to suppress this message
Syntax OK
```

Apache Webserver on OSX

- **`sudo apachectl start`**
- On a browser, enter: http://localhost/

```
~$ ps -ax | grep httpd
52404 ??       0:00.40 /usr/sbin/httpd -D FOREGROUND
52420 ??       0:00.00 /usr/sbin/httpd -D FOREGROUND
52422 ttys000    0:00.00 grep httpd
```

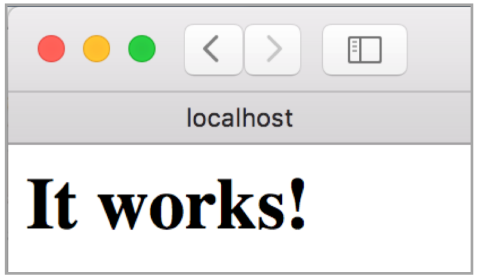
not running…



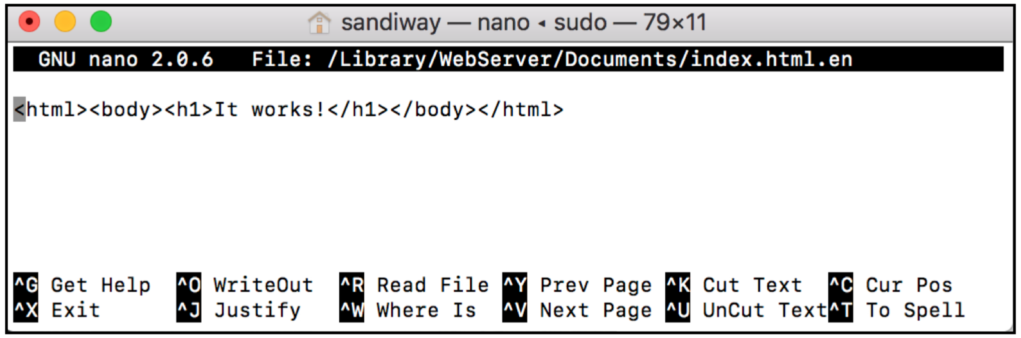
Safari Can't Connect to the Server

Safari can't open the page "localhost" because Safari can't connect to the server "localhost".

running…



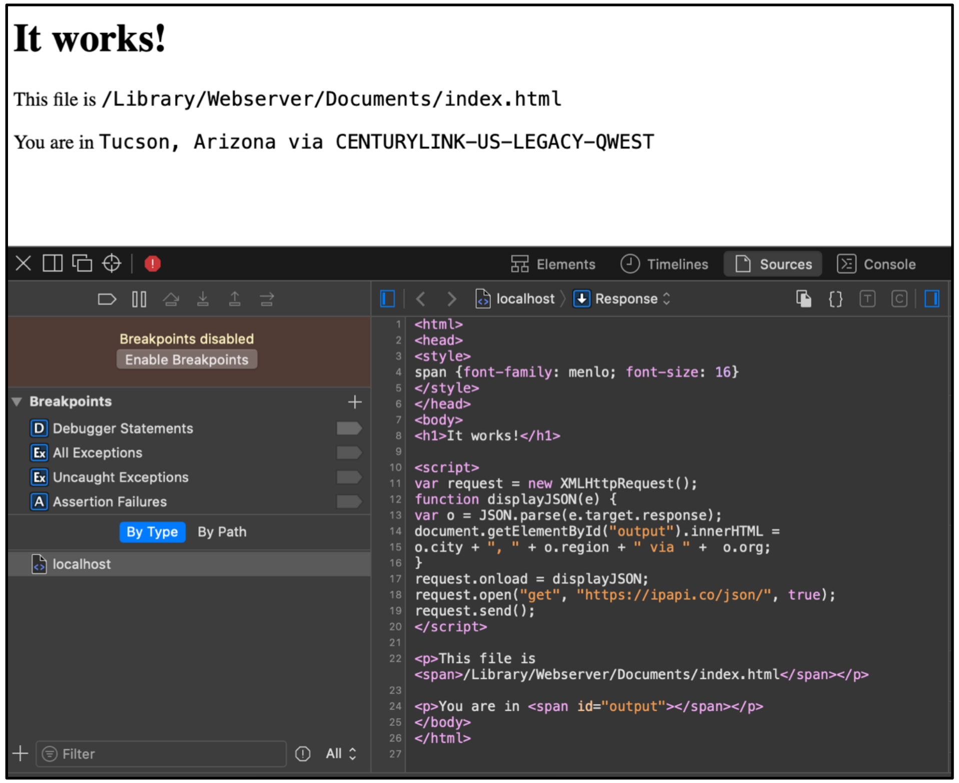
It works!



```
  GNU nano 2.0.6    File: /Library/WebServer/Documents/index.html.en

<html><body><h1>It works!</h1></body></html>





^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

Apache Webserver on OSX

- On MacOS Catalina:



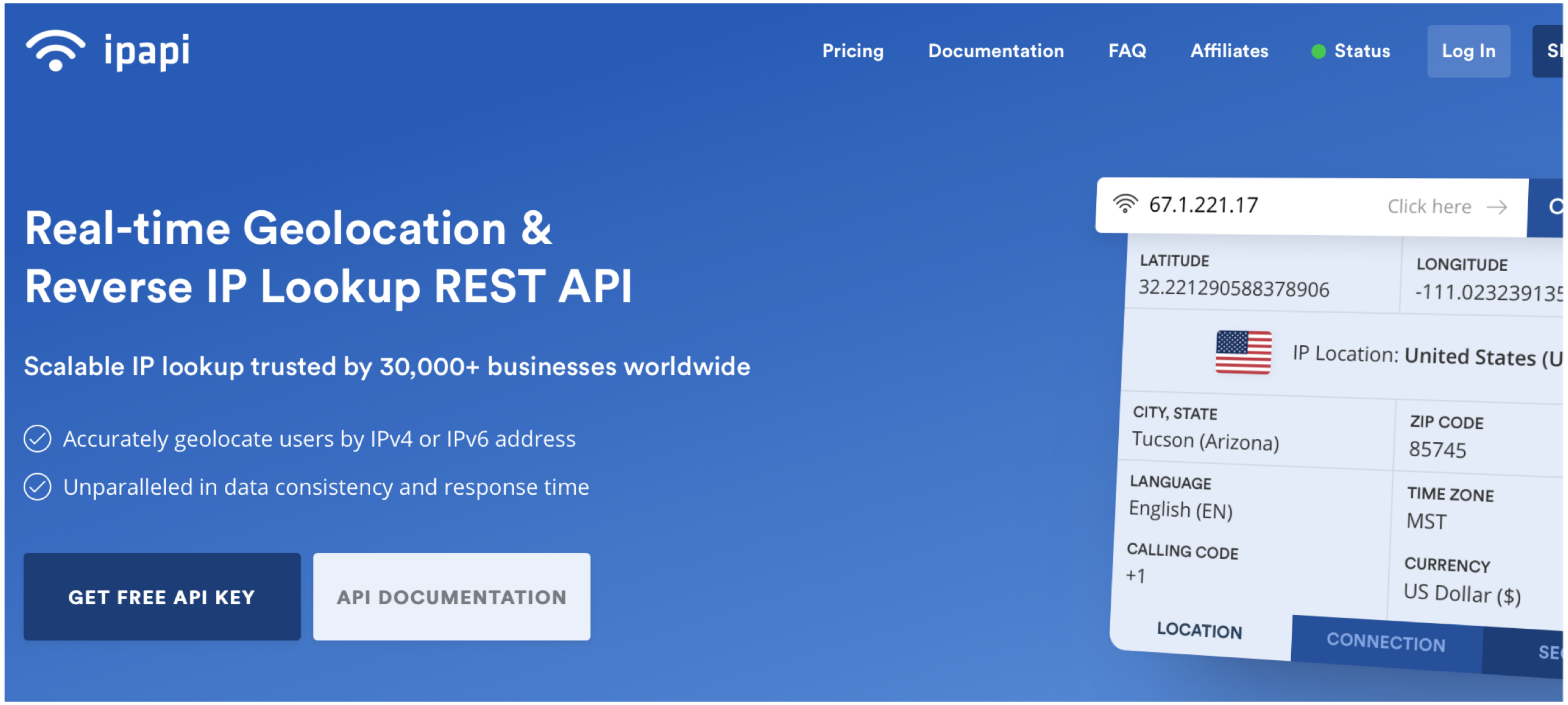
It works!

This file is `/Library/Webserver/Documents/index.html`

You are in `Tucson, Arizona via CENTURYLINK-US-LEGACY-QWEST`

```
<html>
<head>
<style>
span {font-family: menlo; font-size: 16}
</style>
</head>
<body>
<h1>It works!</h1>

<script>
var request = new XMLHttpRequest();
function displayJSON(e) {
var o = JSON.parse(e.target.response);
document.getElementById("output").innerHTML =
o.city + ", " + o.region + " via " +  o.org;
}
request.onload = displayJSON;
request.open("get", "https://ipapi.co/json/", true);
request.send();
</script>

<p>This file is <span>/Library/Webserver/Documents/index.html</span></p>

<p>You are in <span id="output"></span></p>
</body>
</html>
```

Apache Webserver on OSX



Real-time Geolocation & Reverse IP Lookup REST API

Scalable IP lookup trusted by 30,000+ businesses worldwide

- Accurately geolocate users by IPv4 or IPv6 address
- Unparalleled in data consistency and response time

67.1.221.17

LATITUDE 32.221290588378906	LONGITUDE -111.02323913
CITY, STATE Tucson (Arizona)	ZIP CODE 85745
LANGUAGE English (EN)	TIME ZONE MST
CALLING CODE +1	CURRENCY US Dollar ($)

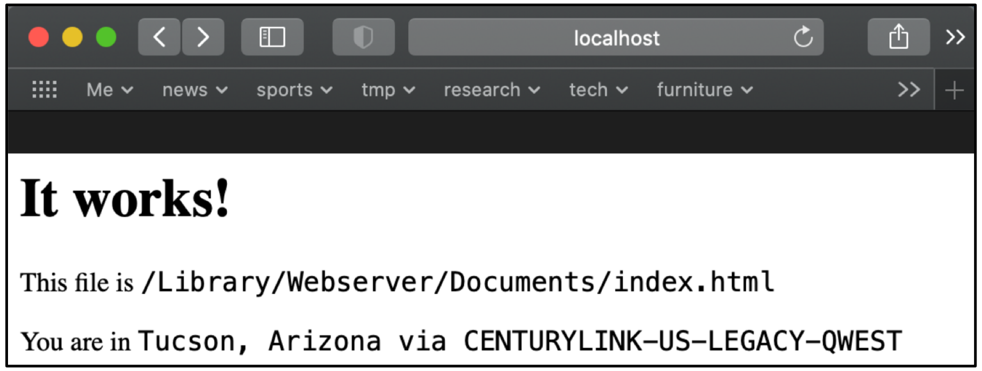
IP Location: United States (U

Sample Site webpage

Normally, javascript is sandboxed for security.

It has no access to your machine details, e.g. IP address or filesystem

So how did we do this?

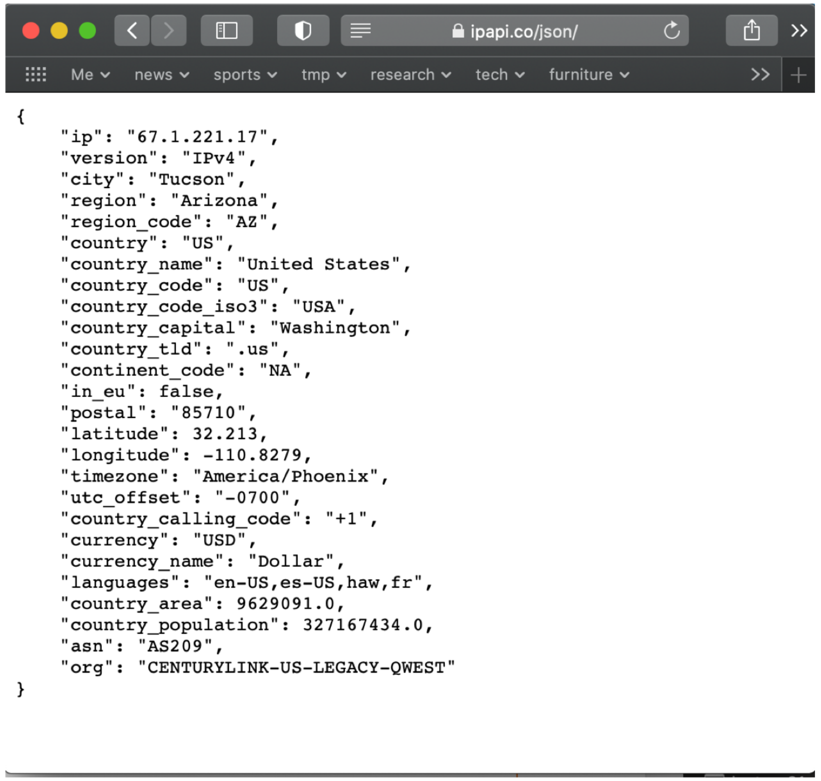


It works!

This file is `/Library/Webserver/Documents/index.html`

You are in `Tucson, Arizona via CENTURYLINK-US-LEGACY-QWEST`

JSON = Javascript Object Notation



```
{
    "ip": "67.1.221.17",
    "version": "IPv4",
    "city": "Tucson",
    "region": "Arizona",
    "region_code": "AZ",
    "country": "US",
    "country_name": "United States",
    "country_code": "US",
    "country_code_iso3": "USA",
    "country_capital": "Washington",
    "country_tld": ".us",
    "continent_code": "NA",
    "in_eu": false,
    "postal": "85710",
    "latitude": 32.213,
    "longitude": -110.8279,
    "timezone": "America/Phoenix",
    "utc_offset": "-0700",
    "country_calling_code": "+1",
    "currency": "USD",
    "currency_name": "Dollar",
    "languages": "en-US,es-US,haw,fr",
    "country_area": 9629091.0,
    "country_population": 327167434.0,
    "asn": "AS209",
    "org": "CENTURYLINK-US-LEGACY-QWEST"
}
```

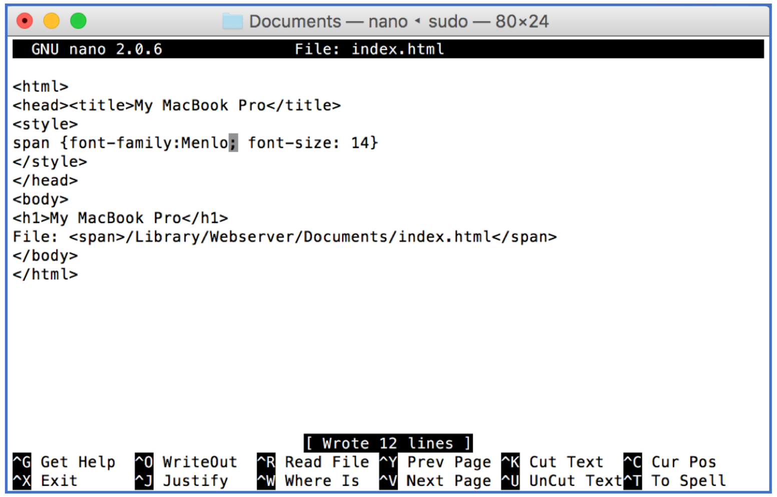
Sample Site webpage

```
<html>
<head>
<style>
span {font-family: menlo; font-size: 16}
</style>
</head>
<body>
<h1>It works!</h1>

<script>
var request = new XMLHttpRequest();
function displayJSON(e) {
var o = JSON.parse(e.target.response);
document.getElementById("output").innerHTML =
o.city + ", " + o.region + " via " +  o.org;
}
request.onload = displayJSON;
request.open("get", "https://ipapi.co/json/",
true);
request.send();
</script>


<p>This file is
<span>/Library/Webserver/Documents/index.html
</span></p>

<p>You are in <span id="output"></span></p>
</body>
</html>
```

Apache Webserver on OSX

Default static webpage storage location:

- http://localhost/
- `/Library/WebServer/Documents/index.html.en~orig`
- Let's create `index.html` ourselves!
- `sudo nano /Library/Webserver/Documents/index.html.en~orig`

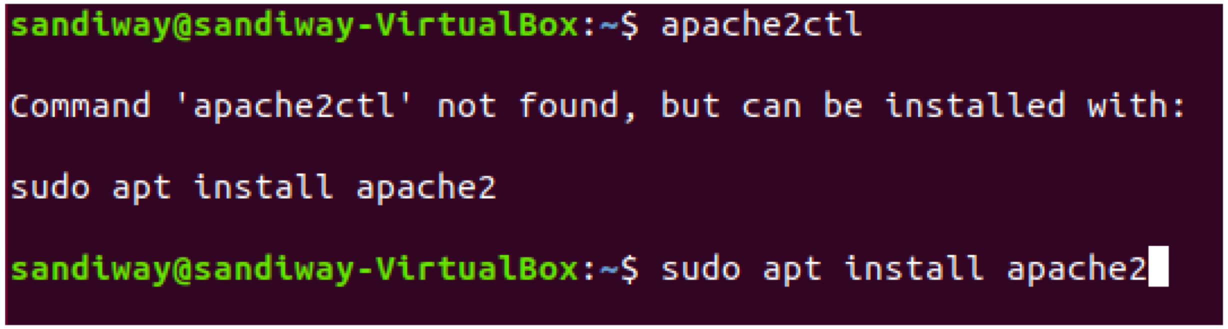


```
  GNU nano 2.0.6                File: index.html

<html>
<head><title>My MacBook Pro</title>
<style>
span {font-family:Menlo; font-size: 14}
</style>
</head>
<body>
<h1>My MacBook Pro</h1>
File: <span>/Library/Webserver/Documents/index.html</span>
</body>
</html>

                              [ Wrote 12 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

nano is a simple text editor
^ means use the Control key
save file as
`/Library/Webserver/Documents/index.html`

Apache Webserver on Ubuntu

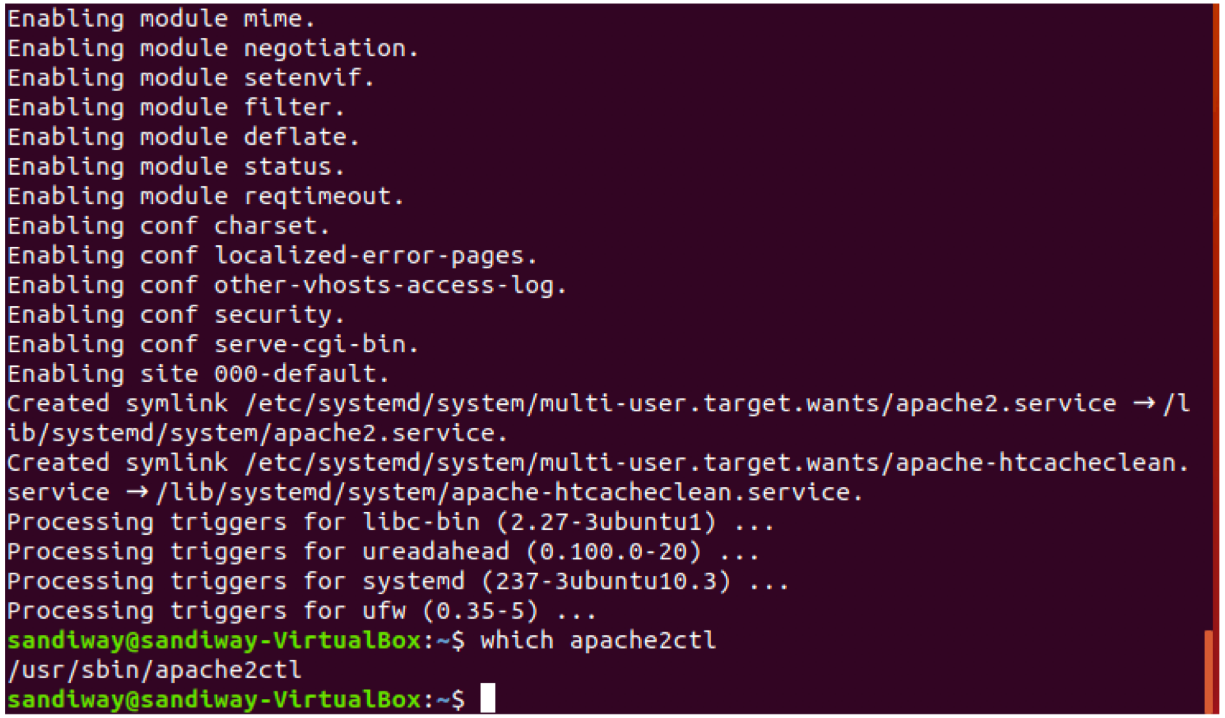
- Ubuntu:
  - **`sudo apt-get update`**



```
sandiway@sandiway-VirtualBox:~$ apache2ctl

Command 'apache2ctl' not found, but can be installed with:

sudo apt install apache2

sandiway@sandiway-VirtualBox:~$ sudo apt install apache2
```

Apache Webserver on Ubuntu

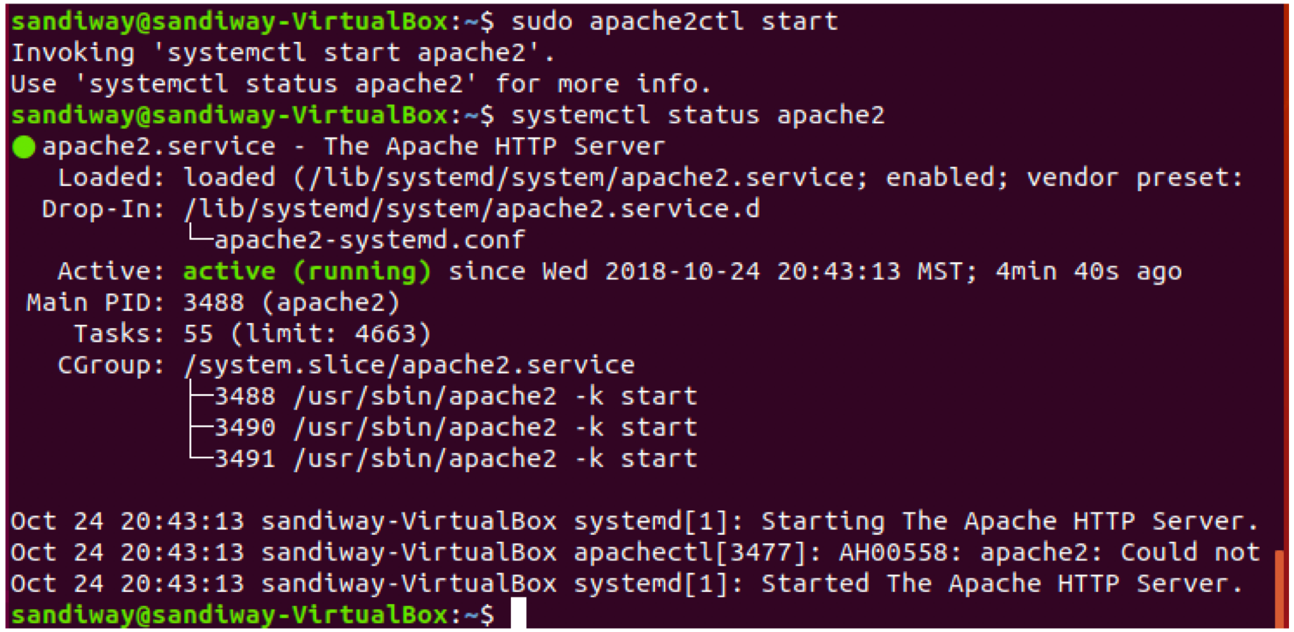
- Ubuntu:
  - **`sudo apt install apache2`** or **`sudo apt-get install apache2`**



```
Enabling module mime.
Enabling module negotiation.
Enabling module setenvif.
Enabling module filter.
Enabling module deflate.
Enabling module status.
Enabling module reqtimeout.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /l
ib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.
service → /lib/systemd/system/apache-htcacheclean.service.
Processing triggers for libc-bin (2.27-3ubuntu1) ...
Processing triggers for ureadahead (0.100.0-20) ...
Processing triggers for systemd (237-3ubuntu10.3) ...
Processing triggers for ufw (0.35-5) ...
sandiway@sandiway-VirtualBox:~$ which apache2ctl
/usr/sbin/apache2ctl
sandiway@sandiway-VirtualBox:~$
```

Apache2 on Ubuntu

- Apache webserver:
  - **`sudo apache2ctl start`**
  - **`sudo apache2ctl stop`**
  - **`sudo apache2ctl restart`**

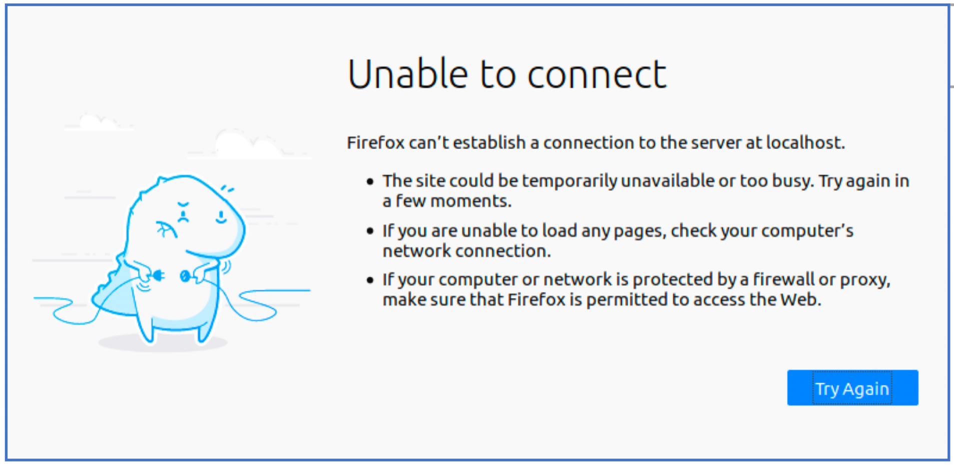


```
sandiway@sandiway-VirtualBox:~$ sudo apache2ctl start
Invoking 'systemctl start apache2'.
Use 'systemctl status apache2' for more info.
sandiway@sandiway-VirtualBox:~$ systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset:
  Drop-In: /lib/systemd/system/apache2.service.d
           └─apache2-systemd.conf
   Active: active (running) since Wed 2018-10-24 20:43:13 MST; 4min 40s ago
 Main PID: 3488 (apache2)
    Tasks: 55 (limit: 4663)
   CGroup: /system.slice/apache2.service
           ├─3488 /usr/sbin/apache2 -k start
           ├─3490 /usr/sbin/apache2 -k start
           └─3491 /usr/sbin/apache2 -k start

Oct 24 20:43:13 sandiway-VirtualBox systemd[1]: Starting The Apache HTTP Server.
Oct 24 20:43:13 sandiway-VirtualBox apachectl[3477]: AH00558: apache2: Could not
Oct 24 20:43:13 sandiway-VirtualBox systemd[1]: Started The Apache HTTP Server.
sandiway@sandiway-VirtualBox:~$
```

Apache2 on Ubuntu

- Apache webserver:
  - **`sudo systemctl start apache2`**
  - **`sudo systemctl stop apache2`**
  - **`sudo systemctl restart apache2`**

not running…



Unable to connect

Firefox can't establish a connection to the server at localhost.

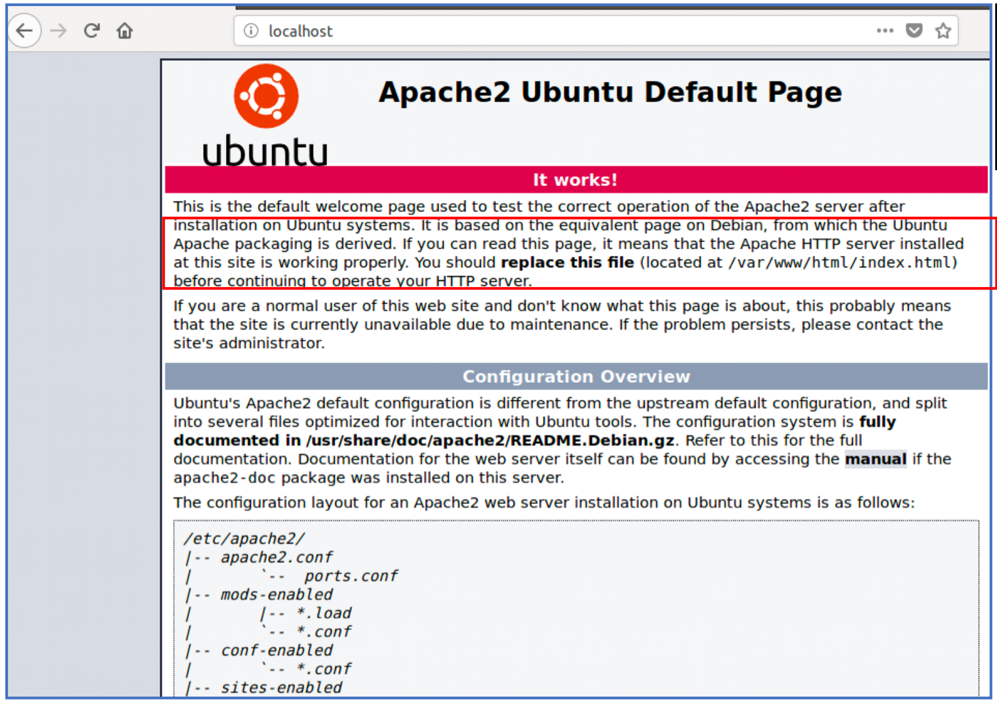
- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the Web.

http://localhost/

Apache2 on Ubuntu

- Apache webserver:
  - **`sudo systemctl start apache2`**

by default it should be running…



Apache2 Ubuntu Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the apache2-doc package was installed on this server.

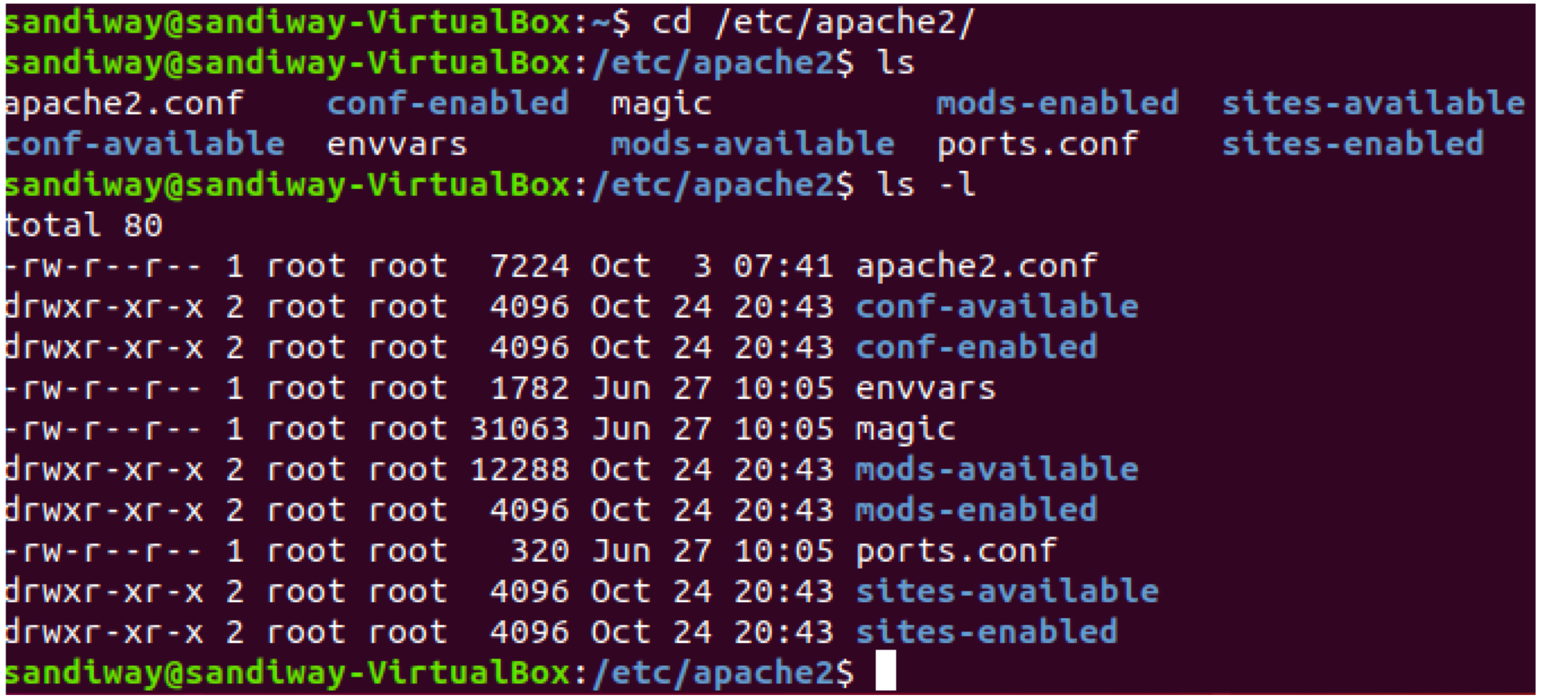
The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|       `--  ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
|       `-- *.conf
|-- sites-enabled
```

DocumentRoot
`/var/www/html`
`/var/www/html/index.html`

Apache2 on Ubuntu

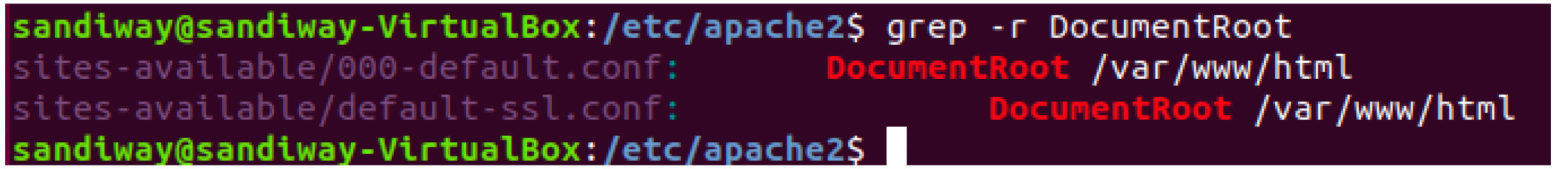
- Master configuration file:
  - **`/etc/apache2/httpd.conf`**



```
sandiway@sandiway-VirtualBox:~$ cd /etc/apache2/
sandiway@sandiway-VirtualBox:/etc/apache2$ ls
apache2.conf    conf-enabled  magic           mods-enabled  sites-available
conf-available  envvars       mods-available  ports.conf    sites-enabled
sandiway@sandiway-VirtualBox:/etc/apache2$ ls -l
total 80
-rw-r--r-- 1 root root  7224 Oct  3 07:41 apache2.conf
drwxr-xr-x 2 root root  4096 Oct 24 20:43 conf-available
drwxr-xr-x 2 root root  4096 Oct 24 20:43 conf-enabled
-rw-r--r-- 1 root root  1782 Jun 27 10:05 envvars
-rw-r--r-- 1 root root 31063 Jun 27 10:05 magic
drwxr-xr-x 2 root root 12288 Oct 24 20:43 mods-available
drwxr-xr-x 2 root root  4096 Oct 24 20:43 mods-enabled
-rw-r--r-- 1 root root   320 Jun 27 10:05 ports.conf
drwxr-xr-x 2 root root  4096 Oct 24 20:43 sites-available
drwxr-xr-x 2 root root  4096 Oct 24 20:43 sites-enabled
sandiway@sandiway-VirtualBox:/etc/apache2$
```

Apache2 on Ubuntu

- **`cd /etc/apache2/`**
- **`grep –r DocumentRoot`**

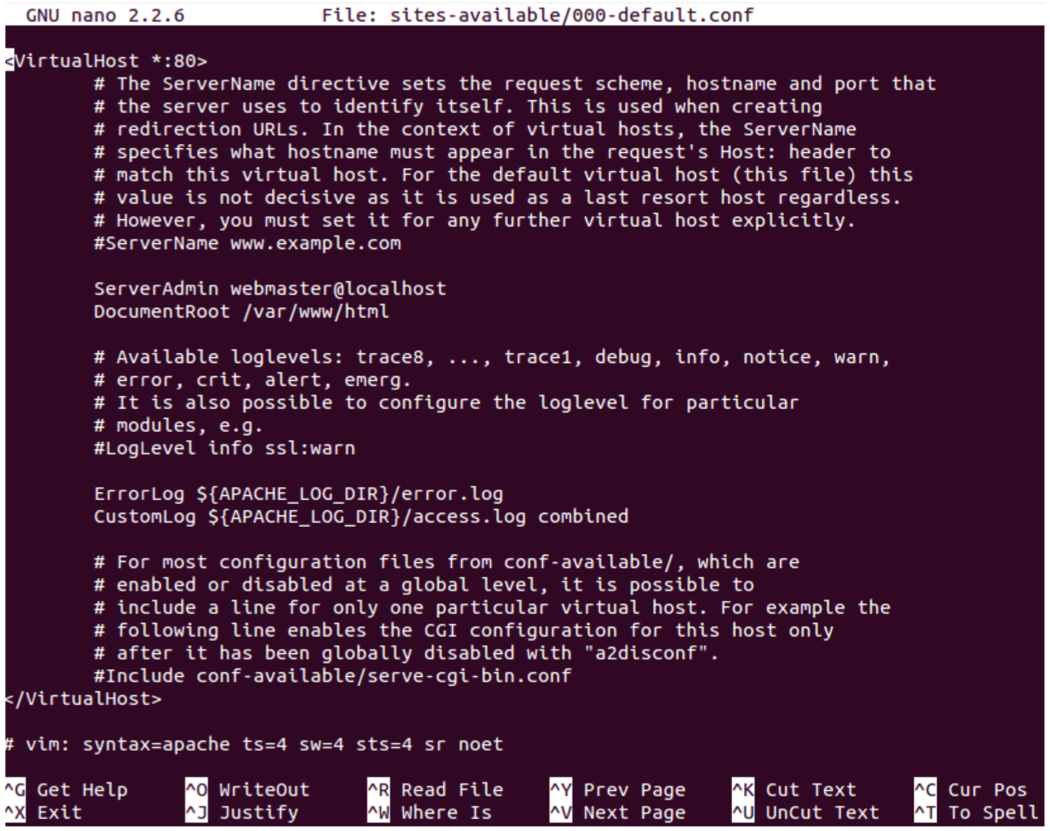


```
sandiway@sandiway-VirtualBox:/etc/apache2$ grep -r DocumentRoot
sites-available/000-default.conf:       DocumentRoot /var/www/html
sites-available/default-ssl.conf:               DocumentRoot /var/www/html
sandiway@sandiway-VirtualBox:/etc/apache2$ 
```

`/etc/apache2/sites-enabled/000-default.conf`

Apache2 on Ubuntu

- **`/etc/apache2/sites-enabled/000-default.conf`**



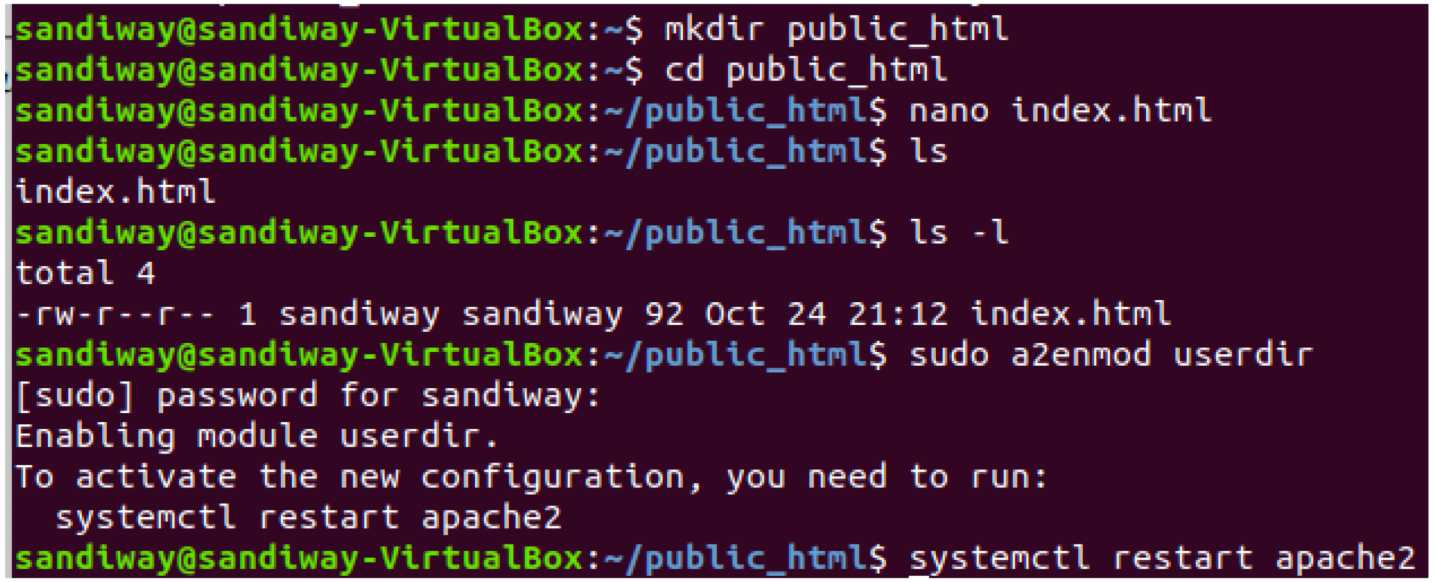
```
  GNU nano 2.2.6             File: sites-available/000-default.conf

<VirtualHost *:80>
        # The ServerName directive sets the request scheme, hostname and port that
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        #ServerName www.example.com

        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        # For most configuration files from conf-available/, which are
        # enabled or disabled at a global level, it is possible to
        # include a line for only one particular virtual host. For example the
        # following line enables the CGI configuration for this host only
        # after it has been globally disabled with "a2disconf".
        #Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

^G Get Help    ^O WriteOut    ^R Read File   ^Y Prev Page   ^K Cut Text    ^C Cur Pos
^X Exit        ^J Justify     ^W Where Is    ^V Next Page   ^U UnCut Text  ^T To Spell
```

Apache2 on Ubuntu

- Logs are in directory: **`/var/log/apache2/`**
  - **`access.log`**
  - **`error.log`**
- User web files in **`~/public_html`**
  - **`mkdir public_html`**
  - **`nano public_html/index.html`**

Apache2 on Ubuntu

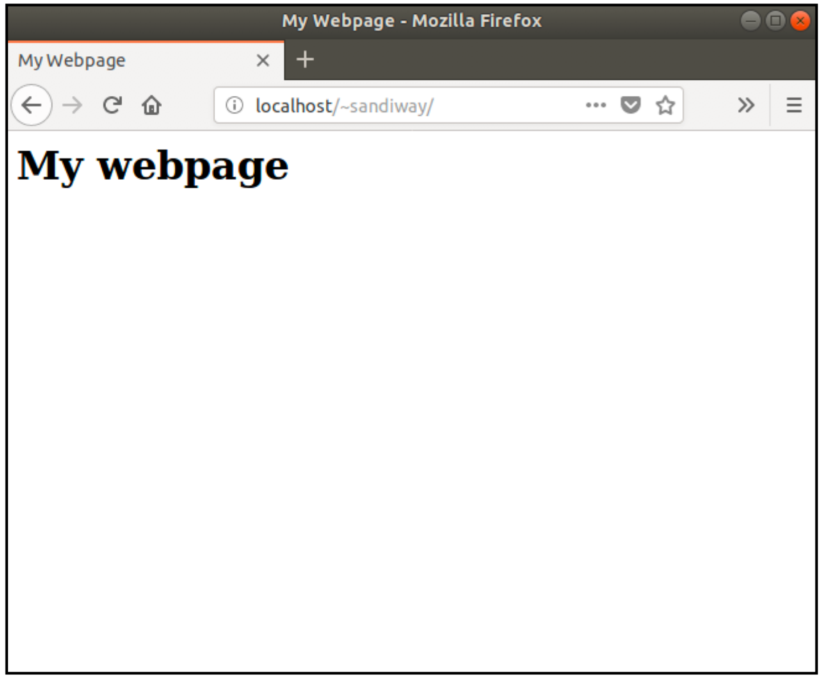
- To enable user web files in **`~/public_html`**
  - **`sudo a2enmod userdir`**
  - **`sudo systemctl restart apache2`**
  - http://localhost/~sandiway/



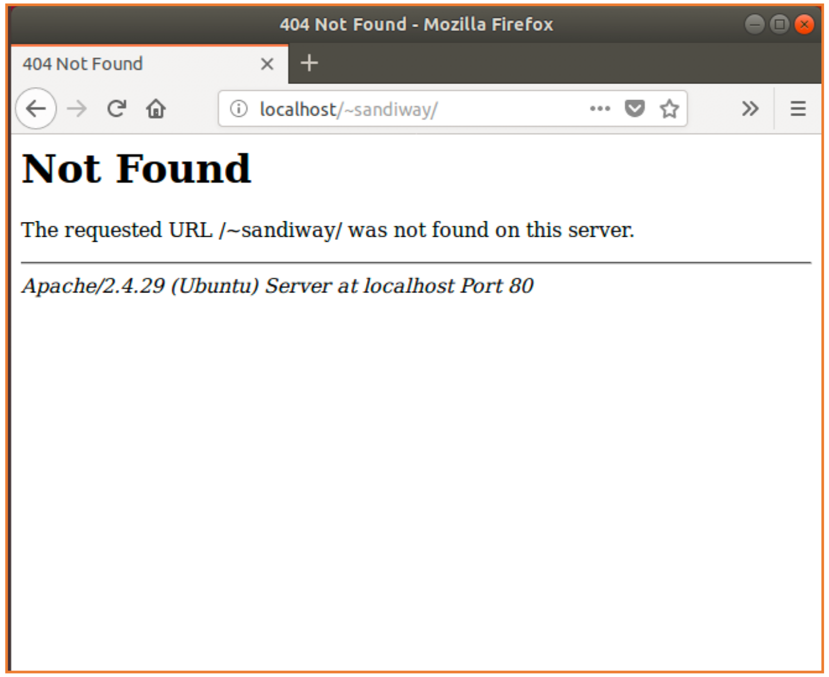
```
sandiway@sandiway-VirtualBox:~$ mkdir public_html
sandiway@sandiway-VirtualBox:~$ cd public_html
sandiway@sandiway-VirtualBox:~/public_html$ nano index.html
sandiway@sandiway-VirtualBox:~/public_html$ ls
index.html
sandiway@sandiway-VirtualBox:~/public_html$ ls -l
total 4
-rw-r--r-- 1 sandiway sandiway 92 Oct 24 21:12 index.html
sandiway@sandiway-VirtualBox:~/public_html$ sudo a2enmod userdir
[sudo] password for sandiway:
Enabling module userdir.
To activate the new configuration, you need to run:
  systemctl restart apache2
sandiway@sandiway-VirtualBox:~/public_html$ systemctl restart apache2
```

Apache2 on Ubuntu

- To enable user web files in **`~/public_html`**
  - **`sudo a2enmod userdir`** **`(a2dismod)`**
  - **`sudo systemctl restart apache2`**
  - http://localhost/~sandiway/



My webpage



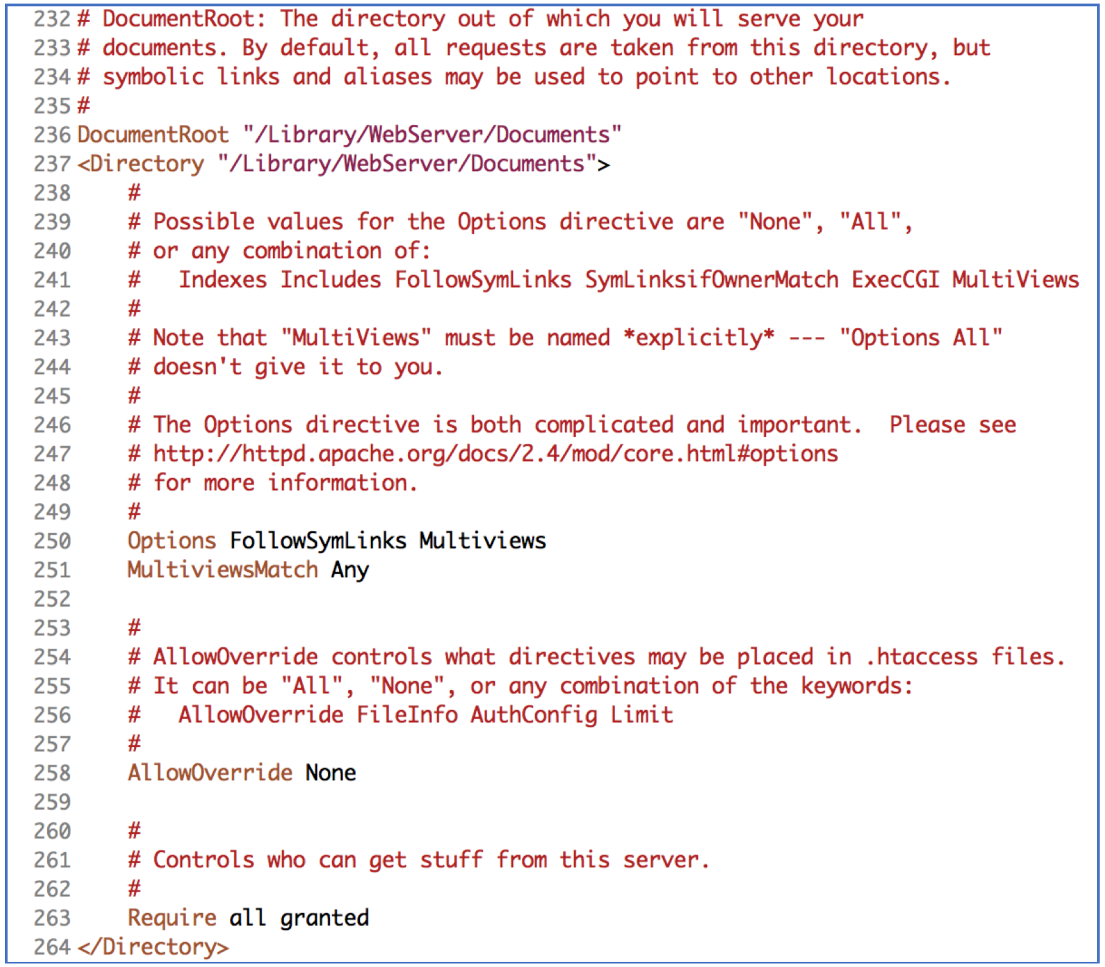
Not Found

The requested URL /~sandiway/ was not found on this server.

Apache/2.4.29 (Ubuntu) Server at localhost Port 80

Apache Webserver on OSX

- Configuration file: **`/etc/apache2/httpd.conf`**



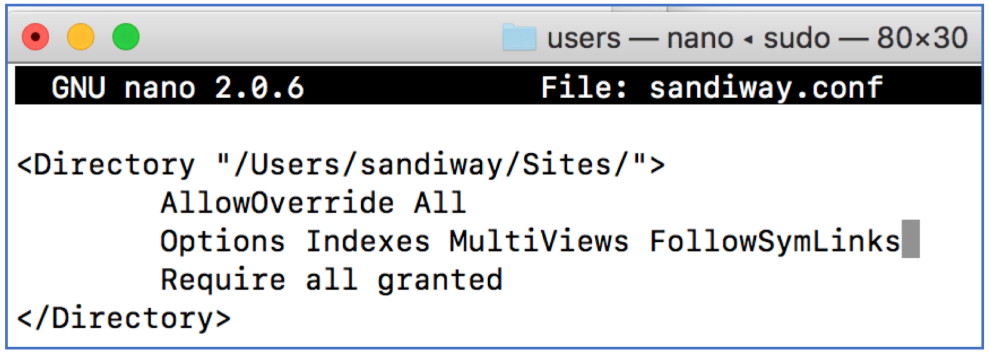
```
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "/Library/WebServer/Documents"
<Directory "/Library/WebServer/Documents">
    #
    # Possible values for the Options directive are "None", "All",
    # or any combination of:
    #   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
    #
    # Note that "MultiViews" must be named *explicitly* --- "Options All"
    # doesn't give it to you.
    #
    # The Options directive is both complicated and important.  Please see
    # http://httpd.apache.org/docs/2.4/mod/core.html#options
    # for more information.
    #
    Options FollowSymLinks Multiviews
    MultiviewsMatch Any

    #
    # AllowOverride controls what directives may be placed in .htaccess files.
    # It can be "All", "None", or any combination of the keywords:
    #   AllowOverride FileInfo AuthConfig Limit
    #
    AllowOverride None

    #
    # Controls who can get stuff from this server.
    #
    Require all granted
</Directory>
```

Apache Webserver on OSX

Static webpages

- storage locations:
  - http://localhost/~sandiway/ **(no need to be superuser)**
  - **`mkdir ~sandiway/Sites`** (**`/Users/`*`username`*`/Sites`**)
  - **`~/Sites/index.html`** (create this file!)
  - **`sudo nano /etc/apache2/users/sandiway.conf`**

create this file …

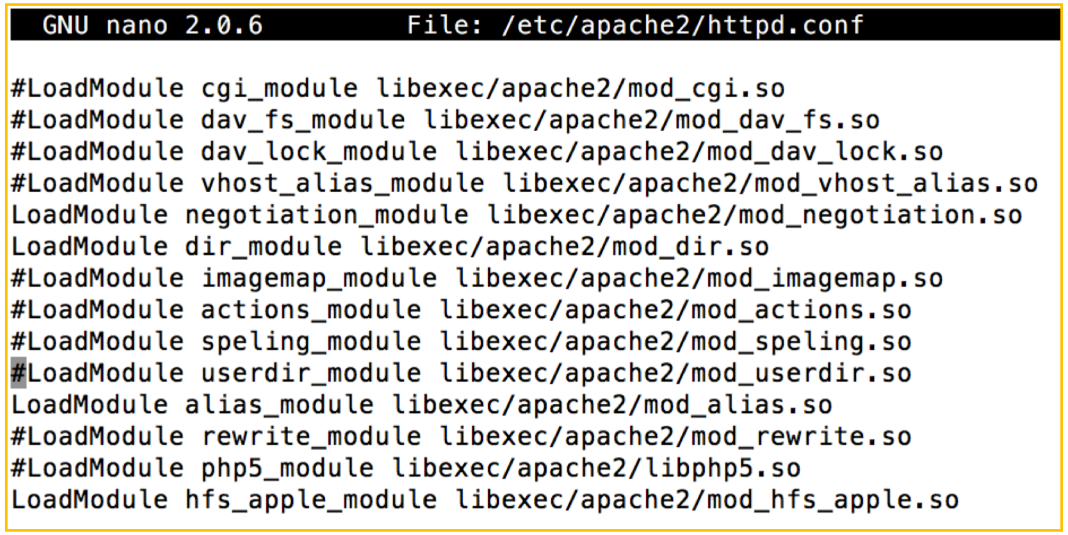


```
  GNU nano 2.0.6                    File: sandiway.conf

<Directory "/Users/sandiway/Sites/">
        AllowOverride All
        Options Indexes MultiViews FollowSymLinks
        Require all granted
</Directory>
```

Apache Webserver on OSX

Static webpages

- storage locations:
  - http://localhost/~sandiway/
  - **`sudo nano /etc/apache2/httpd.conf`**



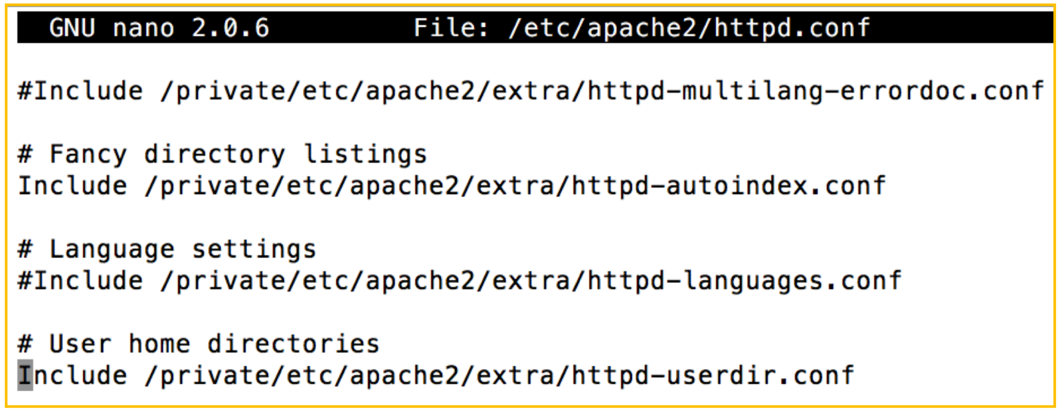
```
  GNU nano 2.0.6            File: /etc/apache2/httpd.conf

#LoadModule cgi_module libexec/apache2/mod_cgi.so
#LoadModule dav_fs_module libexec/apache2/mod_dav_fs.so
#LoadModule dav_lock_module libexec/apache2/mod_dav_lock.so
#LoadModule vhost_alias_module libexec/apache2/mod_vhost_alias.so
LoadModule negotiation_module libexec/apache2/mod_negotiation.so
LoadModule dir_module libexec/apache2/mod_dir.so
#LoadModule imagemap_module libexec/apache2/mod_imagemap.so
#LoadModule actions_module libexec/apache2/mod_actions.so
#LoadModule speling_module libexec/apache2/mod_speling.so
#LoadModule userdir_module libexec/apache2/mod_userdir.so
LoadModule alias_module libexec/apache2/mod_alias.so
#LoadModule rewrite_module libexec/apache2/mod_rewrite.so
#LoadModule php5_module libexec/apache2/libphp5.so
LoadModule hfs_apple_module libexec/apache2/mod_hfs_apple.so
```

uncomment `mod_userdir.so` line (*remove the comment char #*)

Apache Webserver on OSX

Static webpages

- storage locations:
  - http://localhost/~sandiway/
  - **`sudo nano /etc/apache2/httpd.conf`**



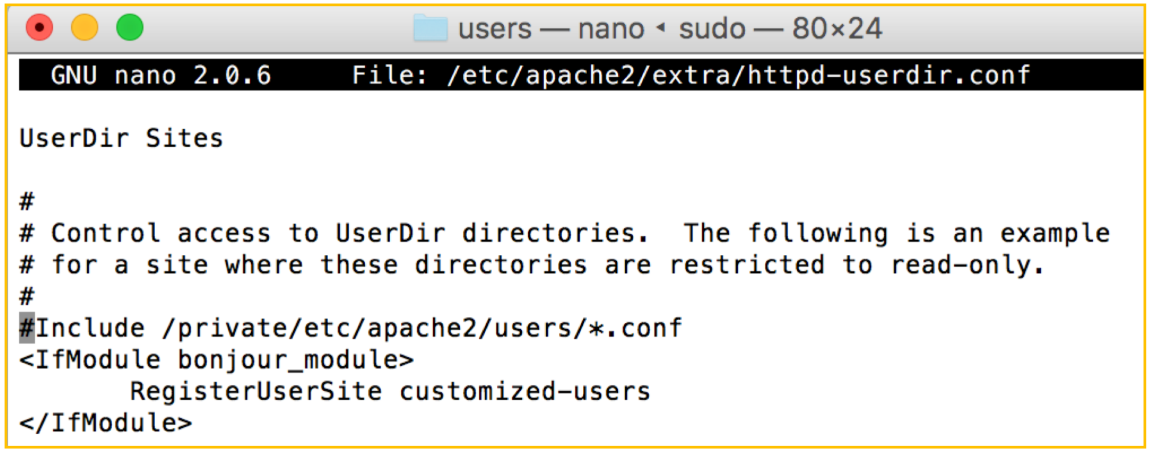
```
  GNU nano 2.0.6             File: /etc/apache2/httpd.conf

#Include /private/etc/apache2/extra/httpd-multilang-errordoc.conf

# Fancy directory listings
Include /private/etc/apache2/extra/httpd-autoindex.conf

# Language settings
#Include /private/etc/apache2/extra/httpd-languages.conf

# User home directories
Include /private/etc/apache2/extra/httpd-userdir.conf
```

uncomment `httpd_userdir.conf` line
(*remove the #*)

Apache Webserver on OSX

Static webpages

- storage locations:
  - http://localhost/~sandiway/
  - **`sudo nano /etc/apache2/extra/httpd-userdir.conf`**



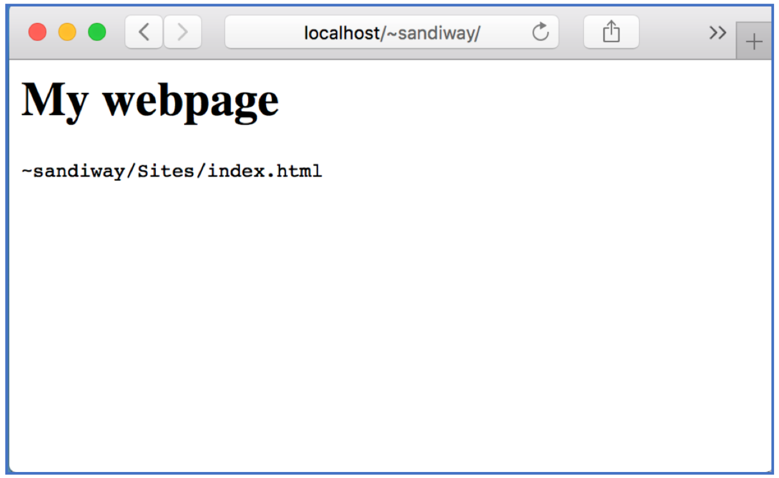
```
  GNU nano 2.0.6      File: /etc/apache2/extra/httpd-userdir.conf

UserDir Sites

#
# Control access to UserDir directories.  The following is an example
# for a site where these directories are restricted to read-only.
#
#Include /private/etc/apache2/users/*.conf
<IfModule bonjour_module>
       RegisterUserSite customized-users
</IfModule>
```

uncomment this include (*remove the #*)

Apache Webserver on OSX

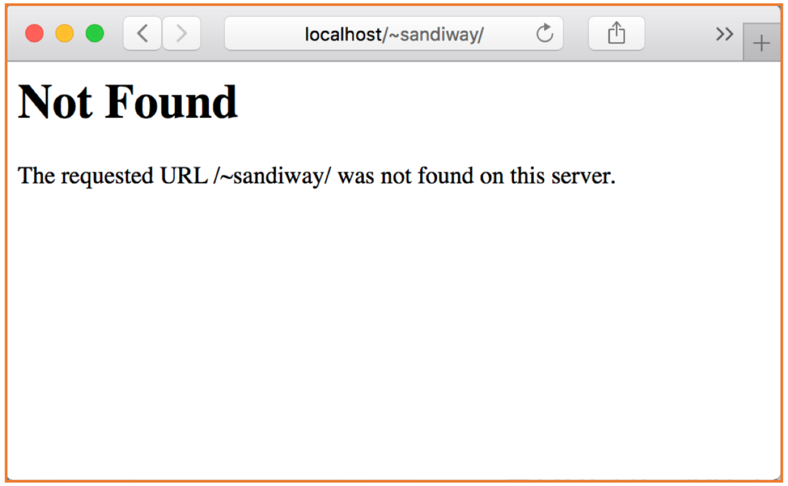
Static webpages

- storage locations:
  - http://localhost/~sandiway/
  - **`sudo apachectl –k restart`**
  - create a file **`~sandiway/Sites/index.html`**



My webpage

`~sandiway/Sites/index.html`

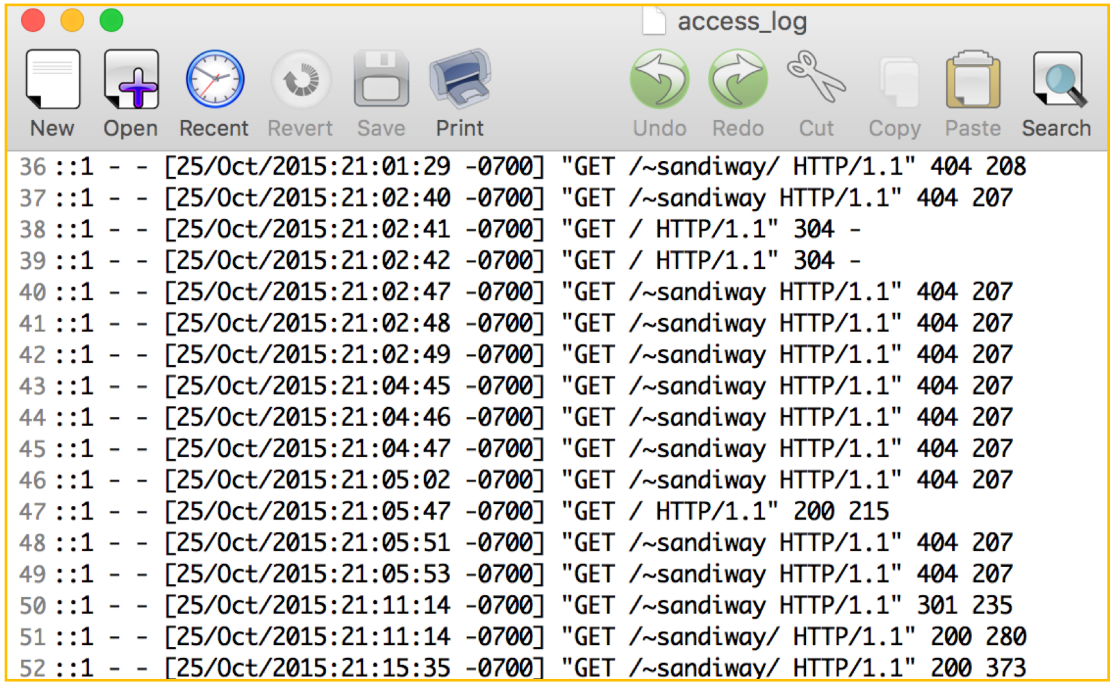


Not Found

The requested URL /~sandiway/ was not found on this server.

Apache Webserver on OSX

- **`/var/log/apache2/access_log`**



```
::1 - - [25/Oct/2015:21:01:29 -0700] "GET /~sandiway/ HTTP/1.1" 404 208
::1 - - [25/Oct/2015:21:02:40 -0700] "GET /~sandiway HTTP/1.1" 404 207
::1 - - [25/Oct/2015:21:02:41 -0700] "GET / HTTP/1.1" 304 -
::1 - - [25/Oct/2015:21:02:42 -0700] "GET / HTTP/1.1" 304 -
::1 - - [25/Oct/2015:21:02:47 -0700] "GET /~sandiway HTTP/1.1" 404 207
::1 - - [25/Oct/2015:21:02:48 -0700] "GET /~sandiway HTTP/1.1" 404 207
::1 - - [25/Oct/2015:21:02:49 -0700] "GET /~sandiway HTTP/1.1" 404 207
::1 - - [25/Oct/2015:21:04:45 -0700] "GET /~sandiway HTTP/1.1" 404 207
::1 - - [25/Oct/2015:21:04:46 -0700] "GET /~sandiway HTTP/1.1" 404 207
::1 - - [25/Oct/2015:21:04:47 -0700] "GET /~sandiway HTTP/1.1" 404 207
::1 - - [25/Oct/2015:21:05:02 -0700] "GET /~sandiway HTTP/1.1" 404 207
::1 - - [25/Oct/2015:21:05:47 -0700] "GET / HTTP/1.1" 200 215
::1 - - [25/Oct/2015:21:05:51 -0700] "GET /~sandiway HTTP/1.1" 404 207
::1 - - [25/Oct/2015:21:05:53 -0700] "GET /~sandiway HTTP/1.1" 404 207
::1 - - [25/Oct/2015:21:11:14 -0700] "GET /~sandiway HTTP/1.1" 301 235
::1 - - [25/Oct/2015:21:11:14 -0700] "GET /~sandiway/ HTTP/1.1" 200 280
::1 - - [25/Oct/2015:21:15:35 -0700] "GET /~sandiway/ HTTP/1.1" 200 373
```

Homework 8

- For Mac owners
  - set up Apache2 on your mac
- For Ubuntu owners
  - set up Apache2 in VirtualBox
- In each case:
  - http://localhost/
  - http://localhost/~yourusername/
  - Create two different index.html webpages at these locations, e.g. add your photo on your user homepage
  - Show your system works! (snapshots)
  - Submit one PDF file (by next Monday midnight)