HTML and Media

More than just images

HTML

- Originally designed for sharing scientific data
- Now, used for everything!
  - Mostly things it was never meant to do…
  - Tables, frames, layout (CSS), embedded media…

Media Types

- Text
- Tabular data
- Images
- Animation (GIFs, then…?)
- Sound
- Video
- 3D?

Video in HTML5

```
<video src="filename">
        Text to show if the browser doesn't support video.
</video>
```

http://www.w3schools.com/tags/tag_video.asp

Supported Formats

Browser	MP4	WebM	Ogg
Internet Explorer	YES	NO	NO
Edge	YES	yes from Anniversary Edition on	*UNDER CONSIDERATION*
Chrome	YES	YES	YES
Firefox	YES from Firefox21 (Windows), Firefox 30 (Linux)	YES	YES
Safari	YES	Add-in support (not native EXCEPT IOS)	Add-in support (not native)
Opera	YES from Opera 25	YES	YES

Video tag attributes

Attribute	Value	Description
audio	muted	Defining the default state of the the audio. Currently, only "muted" is allowed
autoplay	autoplay	If present, then the video will start playing as soon as it is ready
controls	controls	If present, controls will be displayed, such as a play button
height	*pixels*	Sets the height of the video player
loop	loop	If present, the video will start over again, every time it is finished
poster	*url*	Specifies the URL of an image representing the video
preload	preload	If present, the video will be loaded at page load, and ready to run. Ignored if "autoplay" is present
src	*url*	The URL of the video to play
width	*pixels*	Sets the width of the video player

The Formats…

- OGG/WebM
  - Open-source
  - Free
- MP4
  - Proprietary/licensed
  - "Better"
    - Significantly higher quality, better file size
- WebM
  - Based on Matroska container
  - Licensed to Google as a perpetual, royalty free (essentially free)
  - Better than OGG for compression and > browser support

https://developer.mozilla.org/en-US/docs/Web/HTML/Supported_media_formats

For Compatibility…

```
<video width="320" height="240" controls="controls">
    <source src="movie.ogg" type="video/ogg" />
    <source src="movie.mp4" type="video/mp4" />
    <source src="movie.webm" type="video/webm" />
   Your browser does not support the video tag.

</video>
```

Those Pesky Codecs…

- Ogg (Theora/Vorbis):
  - .ogg
  - VLC
- MPEG4 (H.264/AAC):
  - .mp4
  - Media Encoder
- WebM (VP8/Vorbis):
  - .webm
  - ???
- If you don't use just the right codecs and wrappers, it might not work. So…

Those Pesky Codecs…

- http://www.mirovideoconverter.com/
- Miro is a free, cross-platform tool for converting media, and it can handle all three HTML5 configurations. Yay!
- And no, this isn't an advertisement…
- Also try:
  - http://www.webmsoft.com/free-webm-encoder.html
  - http://www.mediasoftzone.com/video-converter/index.html?gclid=CJv84MDhq6gCFeJ95Qodv B56IQ

R·I·T

Audio

- We can also play audio directly in the browser with the <audio> tag

- But do we want to? Hmm…

Audio tag

```
<audio controls>
        <source src="horse.mp3" type="audio/mpeg">
        Your browser does not support the audio tag.
</audio>
```

http://www.w3schools.com/tags/tag_audio.asp

Supported Formats

Browser	MP3	WAV	OGG
Internet Explorer	YES	NO	NO
Edge	YES	YES	*UNDER CONSIDERATION*
Chrome	YES	YES	YES
Firefox	YES	YES	YES
Safari	YES	YES	Add-in support (not native)
Opera	YES	YES	YES

Website to test HTML 5 Audio Formats:
http://hpr.dogphilosophy.net/test/

More info on HTML5 video

- http://www.w3schools.com/html/html5_video.asp
- http://dev.w3.org/html5/spec/Overview.html
- http://www.html5video.org/
- http://www.w3schools.com/html5/html5_video.asp
- http://caniuse.com/

Other browser media…

- <canvas> and JavaScript
  - 2D raster-based drawing environment
  - Supports bitmap images, animation, video…
- WebGL
  - 3D right in the browser!
- Want to know more?
  - 330 Rich Media Web App Dev. I
  - Google it!

R·I·T

Putting Your HTML Files On-Line

An introduction to SFTP

HTML and the WWW meet…

- The HTML in your web page (in conjunction with CSS) tells a web browser what it needs to know to properly display the page.
- Once you put your HTML files on a web server, any browser connected to the Internet can retrieve and view your files.

Client-Server

- What does the web **server** do?
  - Waits for requests for HTML documents, image and sound files and so on. It then sends (*serves*) them back to the requestor (usually a web browser)
  - RIT has a web server for student use called **banjo.rit.edu**
- What does a web browser (**client**) do?
  - Requests HTML pages (and related files).
  - Retrieves HTML pages (and related files).
  - Displays ("renders") the page for the viewer.

Getting Files online using the FTP protocol

- For your pages to be visible on the web, you will need to transfer your HTML and related files (images and so on) to a *web server* that is connected to the Internet.
- To do this you will use an FTP (File Transfer Protocol) program
- Here in the labs, you might use a program such as WinSCP or *Filezilla* to transfer you files.
- The Mac only ***Fetch*** program is a nice option too.

FTP Commands

- **All FTP clients (programs) have these commands, but they can be implemented in slightly different ways.**
  - **PUT -** take files from your local computer and *transfer* a copy of them to the server. Also called *uploading*.
  - **GET -** retrieve copies of files from the server and place them on your local computer's hard drive. Also called *downloading*.
  - **MKDIR -** create a folder (or directory) on the server. Absolutely required to keep your site organized.

Information the FTP program needs

1) The name (location) of the **web server** that will host your files. Ex. *banjo.rit.edu*

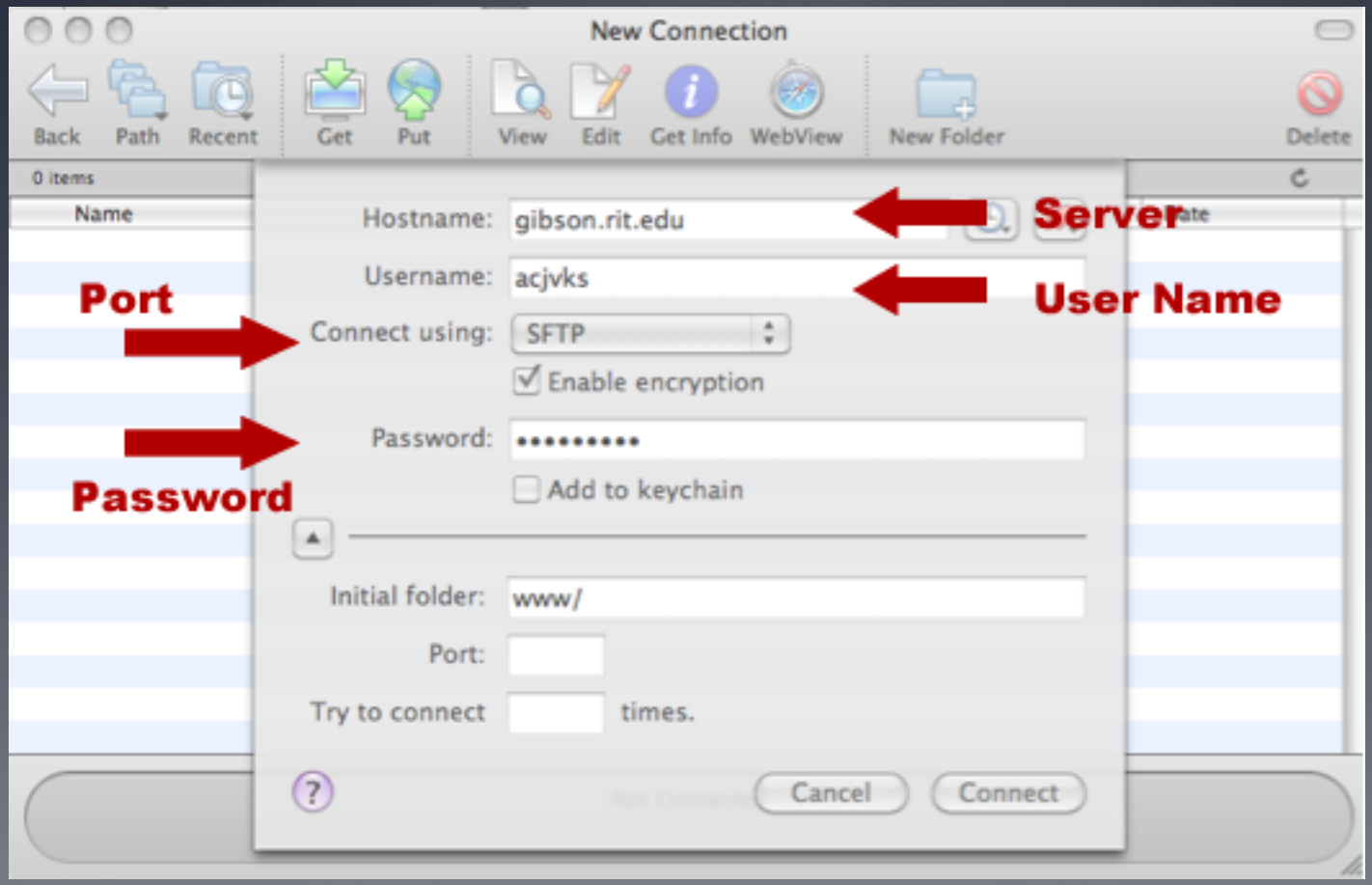
2) A **user name** ex. *abc1234*

3) The account's **password**

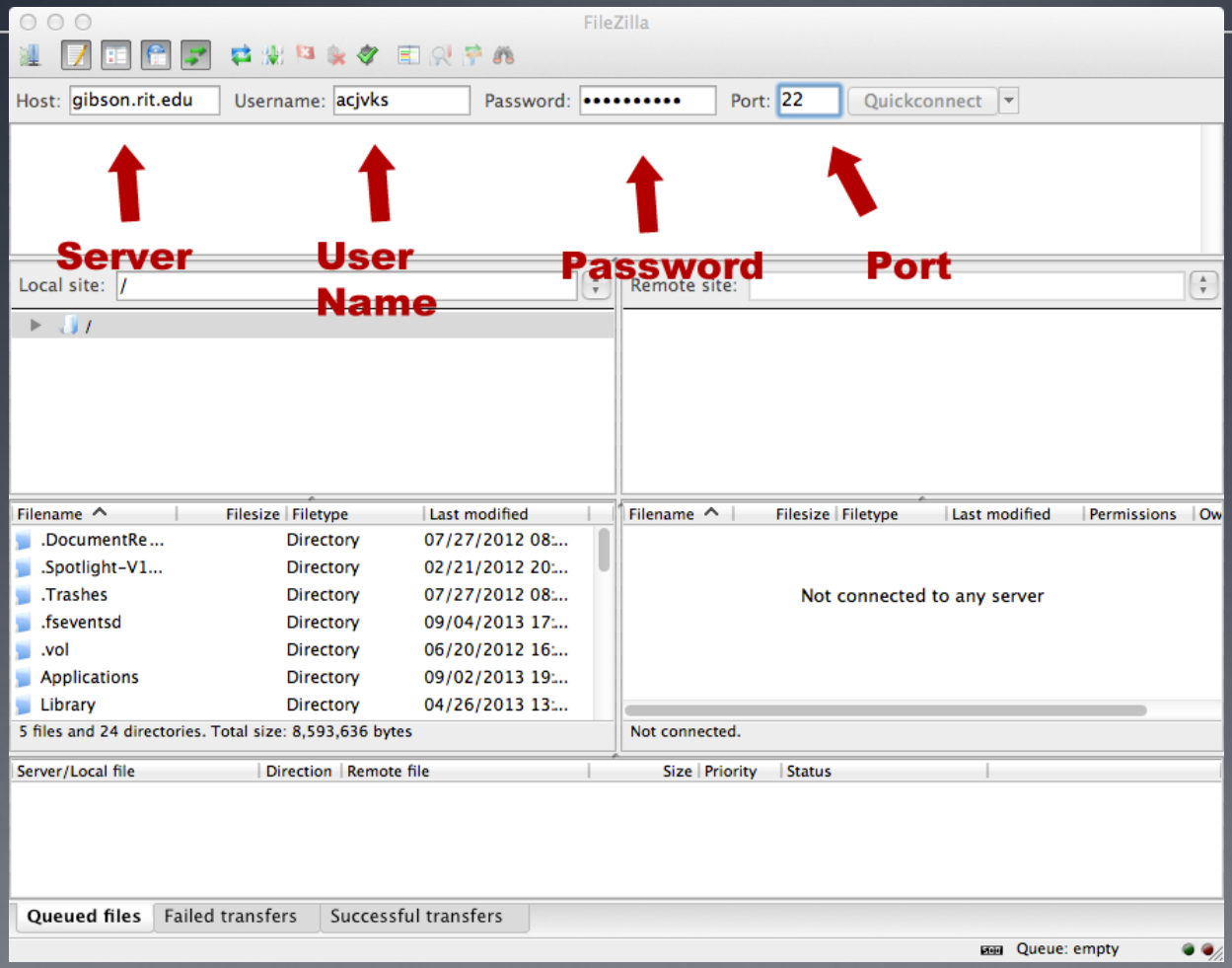
4) What **protocol** to use:
   - FTP -Port 21 **OR**
   - Secure FTP (SFTP) - Port 22 - *this is what banjo .rit.edu uses*

- Once you are connected, you need to decide which folder to place the files in. On banjo, this needs to be your *www* folder.

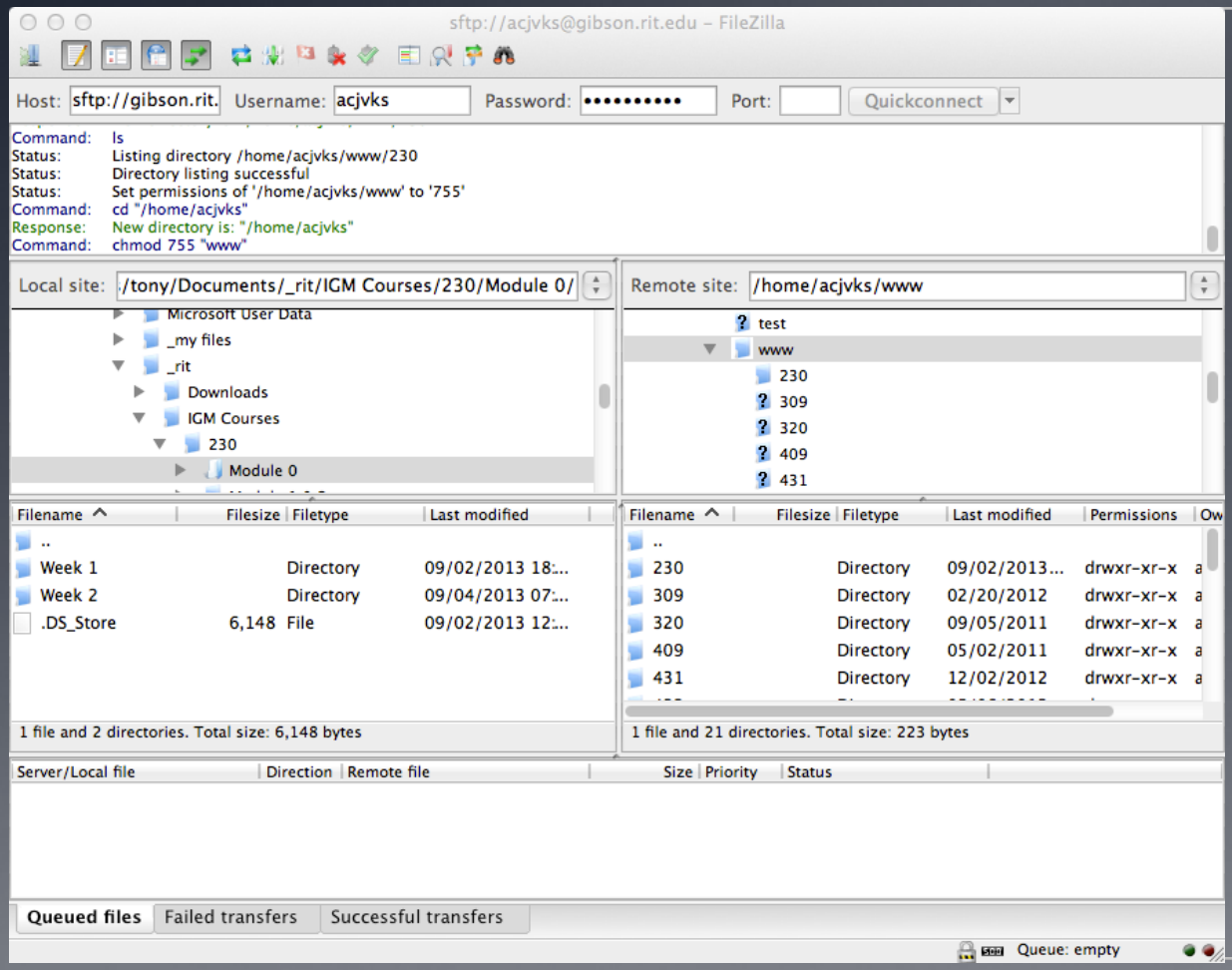
Example: Fetch on Mac



Example: Filezilla on Mac



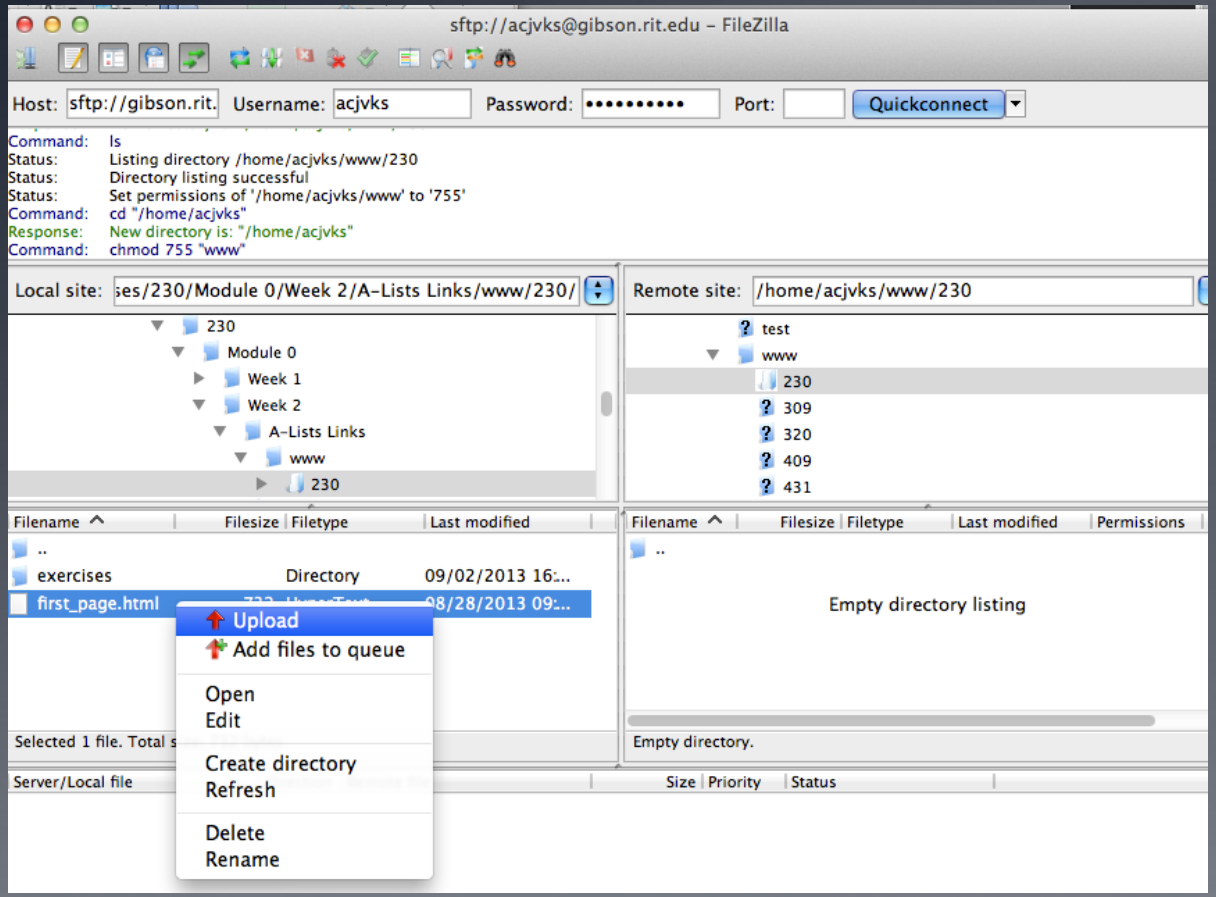
Example: Filezilla on Mac



Once you are logged in, you will see your *local disk* on the left, and your *remote account* on the right.

All your web files will go in the *www* folder, or one of its sub folders.

Example: Filezilla on Mac



To upload a file to the web server, you can drag and drop to the correct folder, or right-click on the file and choose *Upload*.

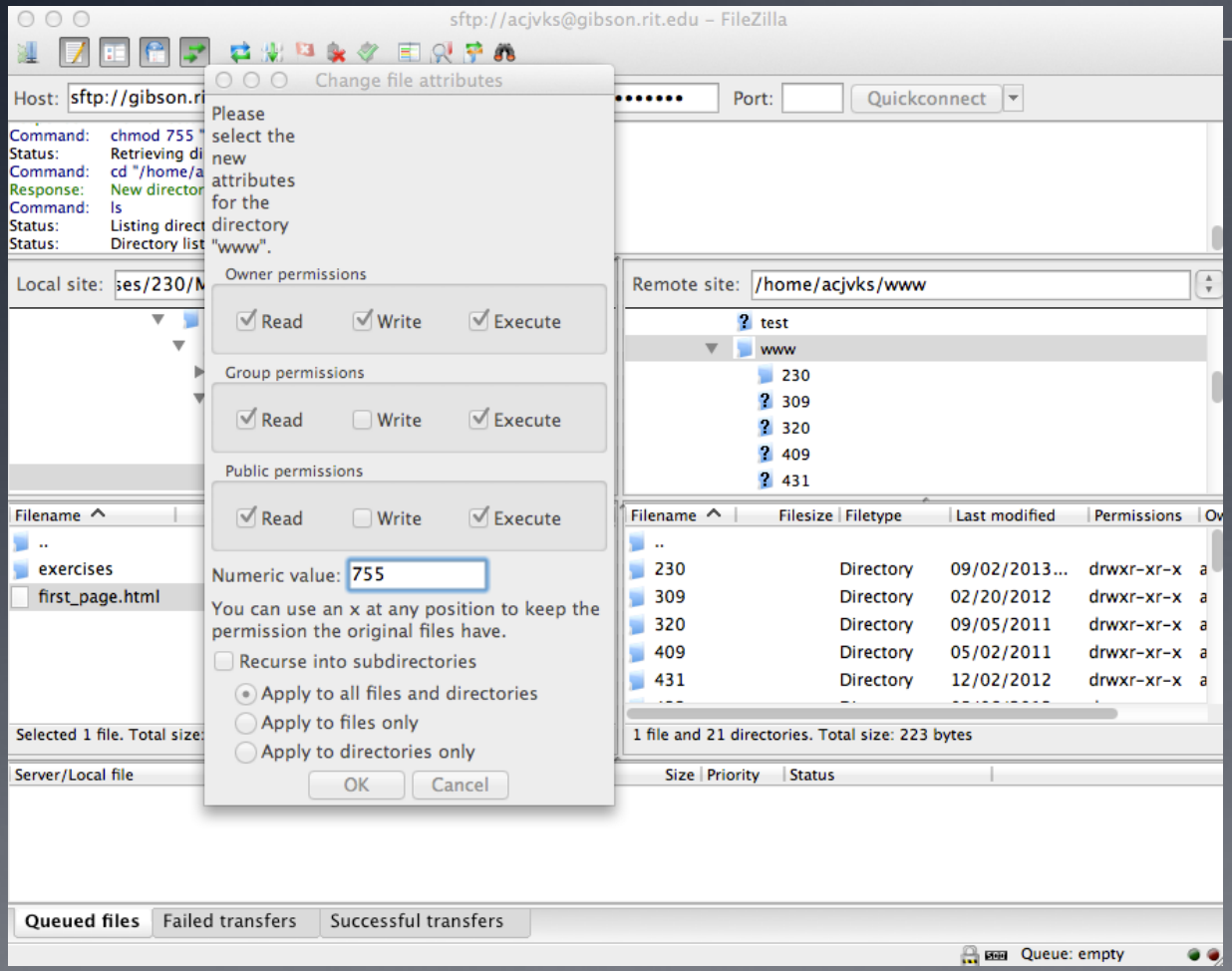
(Note: Downloading files is similar)

One more issue, file permissions

- Files and folders have to have the correct *file permissions*, or the web server will not be able to retrieve them.
- You can easily set these permissions using an FTP client.
- For **folders** the permission level is **755**, for **files** the permission level is **644**, and your **home directory** (abc1234) should be **711** (see next slide)
- We will explain what these numbers mean later in the course

R·I·T

Example: Setting permissions with Filezilla



1) right-click on the file or folder you want to change file permission on and choose "File Attributes …"

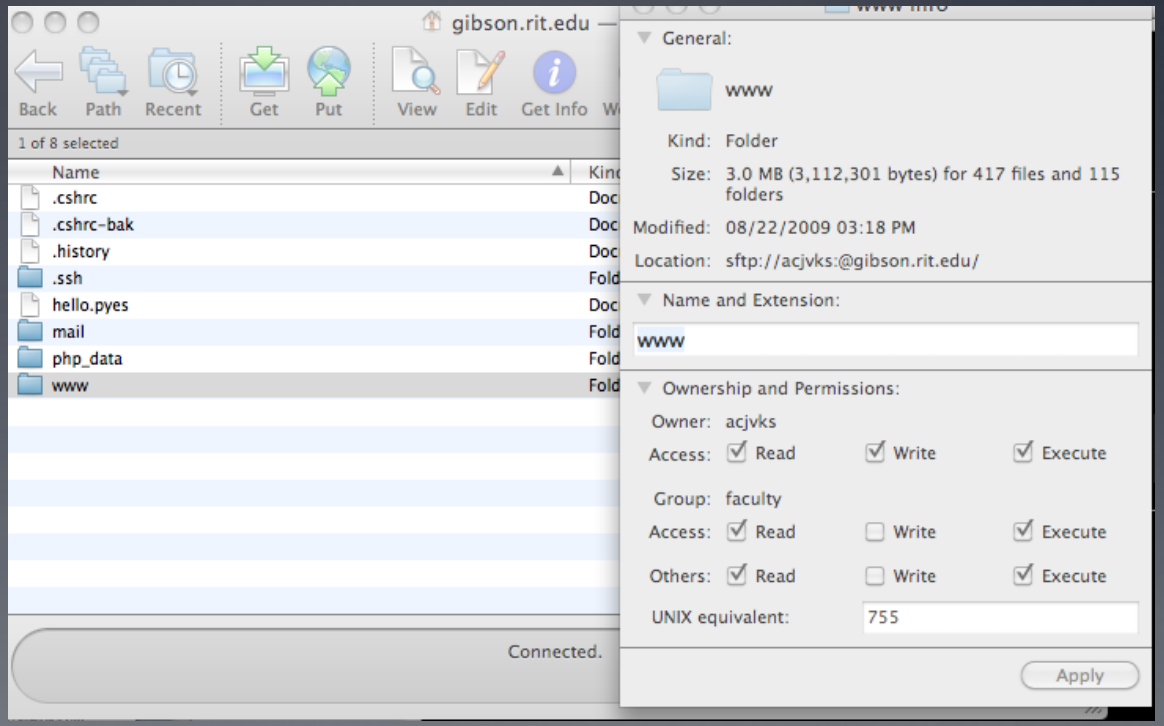
2) Type in the desired value and press OK.

Setting file permissions in Fetch

To set file permissions in Fetch:

1) Select a file and click the "Get Info" button.

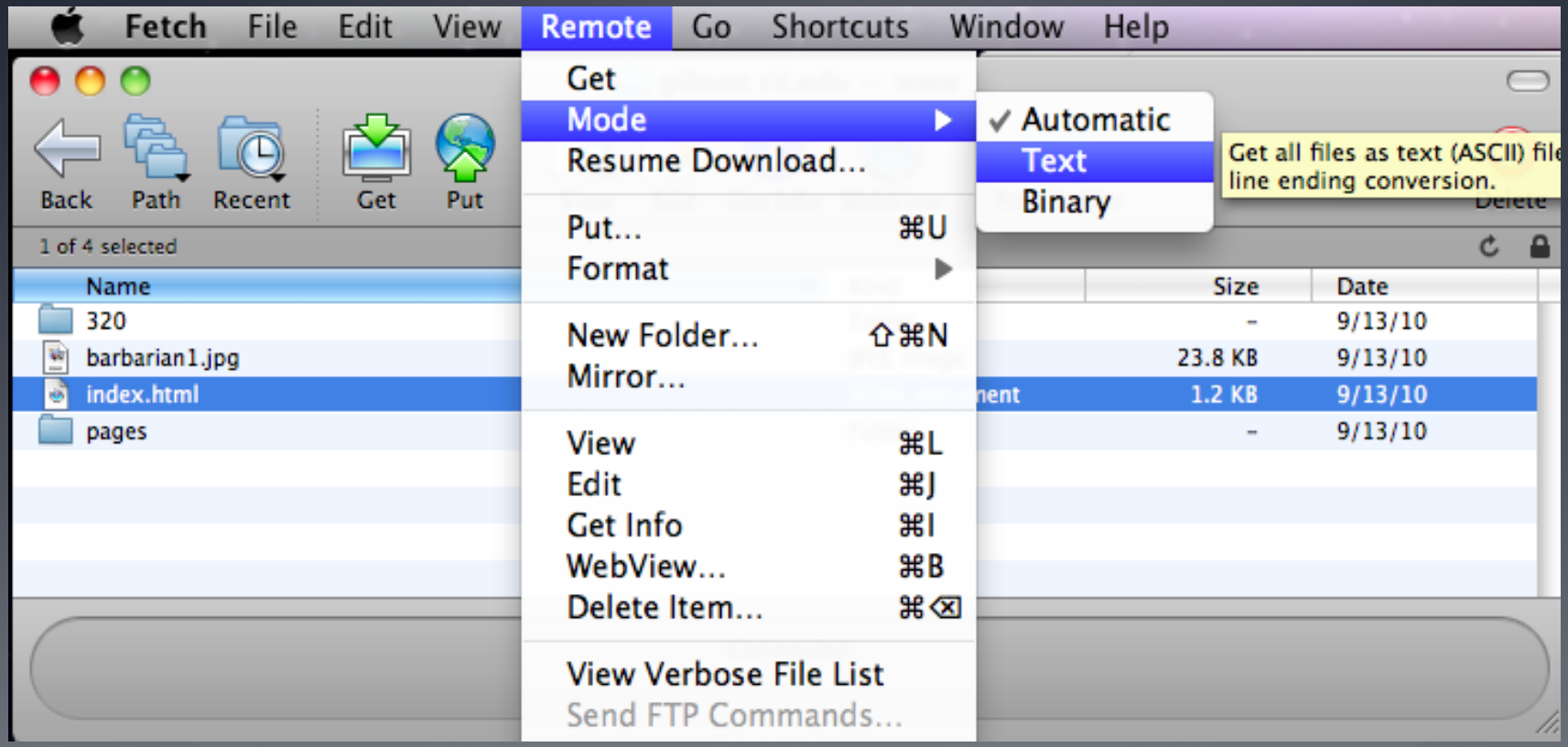
2) Type in the correct number and click the "Apply" button.



Lastly, FTP transfer *modes*

- FTP programs will transfer your files in one of two *modes*, either as:
  - text (HTML, CSS, and JavaScript files) OR
  - binary (images, Flash movies, everything that's not text)
- FTP programs will *usually* make the right choice for you, but sometimes you have to set it manually.

Setting the transfer mode in Fetch



In Filezilla, Choose Transfer > Manual Transfer …

Transferring Files to your web account

A) Gather the 4 pieces of information you need:

1) Server Name: banjo.rit.edu
2) Port: 22 or SFTP
3) Account: Your *abc1234* login
4) Password: your RIT password

B) Launch your FTP client, fill in the information, and connect to the server.

Transferring Files to your web account

C) Set the permissions on your *www* folder to 755

D)In your *www* folder, create a folder named *230*. Also set its permissions to 755

E)In the *230* folder, create a folder named *exercises.* Also set its permissions to 755

F)Upload the files and folders we created last time: *index.html*, *recipe.html* (in the *exercises* folder), *movies.html* (in the *exercises* folder), and the image file that goes with the movies page (also in the *exercises* folder)
If you've set up your local directories properly, you shouldn't actually need to post everything separately… just upload your whole 230 directory to *www*

Transferring Files to your web account

G) Now set permissions on all of the files you just uploaded to 644. Double-check that the permissions on all the folders are 755

H) In a web browser, navigate to:http://people.rit.edu/abc1234/230

(of course, replace "abc1234" with your ID) and you should see your files and folders. Go ahead and click on each on to be sure that they load into the browser. Also check the links we added to them last time.

Issues?

G) If you get a permissions error, double check the file and folder permissions. Also verify that your "home directory" on banjo (the abc1234 directory) has 711 or 755 permissions.

H) **Submission:** Make sure the link to your page from my classlist (linked from myCourses) works. If not, you need to fix your locations!