CREATING A BASIC WEB SITE

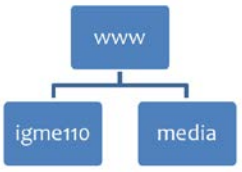
Goal

The goal of this exercise is to create a set of linked and illustrated web pages, using relative paths.

DO NOT COPY AND PASTE FROM THIS EXERCISE TO YOUR HTML DOCUMENT. The characters used in word docs or PDFs (particularly quotation marks) will not always copy properly, and can cause errors in your HTML.

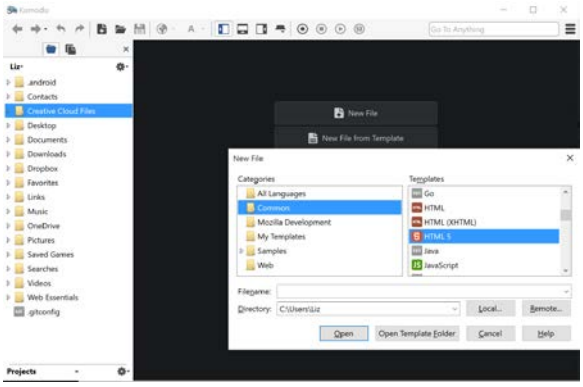
Part 1: Creating the Folder Structure

1. Create a folder called www on some persistent storage you can access from another computer after today's class is over. You can use cloud storage, a networked drive, or a USB drive. Inside of your new **www** folder, create two folders--one called **igme110**, and one called **media.** (All folder names should be lower case.)



2. Using your preferred HTML editor, create a new HTML 5 document. (This exercise uses Komodo Edit as the editor, but any HTML editor will work.)

   Add your name in the title of the document.

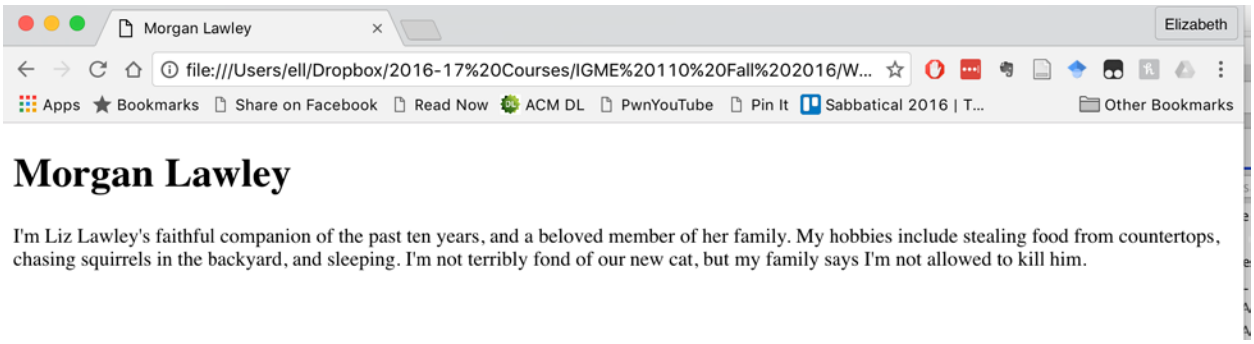


3. Make sure your cursor is on the line below the <body> tag, and put add your name here, as well. Make it a level 1 heading by putting <h1> before it and </h1> after it.

4. Put your cursor on the line below the heading, and add the opening and closing tags for a paragraph. (and) Between the two tags, write a line or two about yourself. (The browser will ignore whitespace and line breaks, so you can press return to keep lines at manageable lengths.)

5. Save the file to the igme110 folder you created, with the file name index.html. Now open a web browser, and choose File→Open File. Locate the index.html file you just saved, and open it. You should see something similar to this:

6.



Part 2: Adding An Image

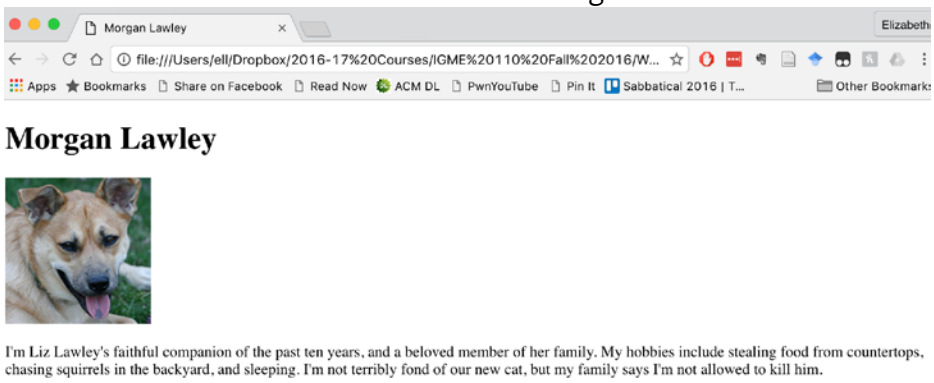
1. Find a Creative Commons-licensed image on the web, or an image that you've got the rights to, that you'd like to include on your page. Save a copy of it to the igme110 folder.

2. You're going to add code to your HTML to display the image properly on the page. In your HTML document, put your cursor after the heading with your name, but before the first paragraph of text. (If there's not a blank line there already, you can add one, to make the code easier to read.) Type in

3. Between the two quote marks following src, we're going to add the *relative* path to the image you downloaded. Because the image is in the same directory as your html file, the relative path is just the file name—if you don't provide any other information besides a file name, the browser will assume that the image is in the same directory as the HTML file it's displaying. My image file is called dog.jpg, so my tag would now look like this:

4. Between the quote marks following *alt*, you're going to add text that will display if the browser is unable to display images. My tag now looks like this:

5. Save your file, go back to the browser window that you used to view it in the last section, and reload the file. It should now look something like this:

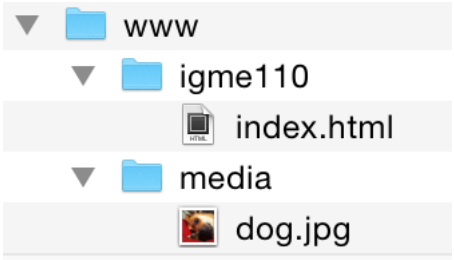


6. If your image isn't showing up properly, ask for help now! (If it's much too big to display properly on the page, try finding and downloading a smaller image from the web.)

Part 3: Relative References to Other Directories

As your web site gets bigger, keeping all your files in the same directory makes it harder to keep them organized. Web developers typically have different directories for different parts of their site, and also often keep images in a separate directory. You created a media folder in your www directory, and that's where we're going to put the images you use on your site.

1. Open the www folder you created. Move your image file from the igme110 folder to the media folder. (Your structure should look like what you see on the right.) Now try reloading your page in the browser again. If you followed the directions properly, the image won't load now. The browser is still looking for it in the same directory as the HTML file, but it's not there. We need to tell the browser how to find the image, using a *relative path*.



7. In your html file, change the img tag so that it looks like this (using your image name, of course):

   This tells the browser to go up one level in the directory structure (that's what **..** is shorthand for in UNIX and HTML), which means it's now looking in the www directory. The /media that follows tells it to look in the media folder that's inside the www folder. Reload the page in the browser to see if it worked. (If not, ask for help!)

Part 3: Creating and Linking Another HTML File

1. In your HTML editor, create a new HTML 5 file. Give it a title of "*Your Name's* Second Page." (Put that in the title tag, as well as in a heading on the page, just as we did with the first file.)

2. Add a paragraph of text after the title.

3. Add another paragraph tag, and inside of the   tags, add an href link back to the index page. Because we're using relative links, and because the file is in the same igme110 directory, you only have to specify the file name, like this:
   My First Page

4. Save the new file to the igme110 directory with the file name page2.html. You should now have two documents open in your editor—the index.html file that you started with, and the page2.html file you just created.

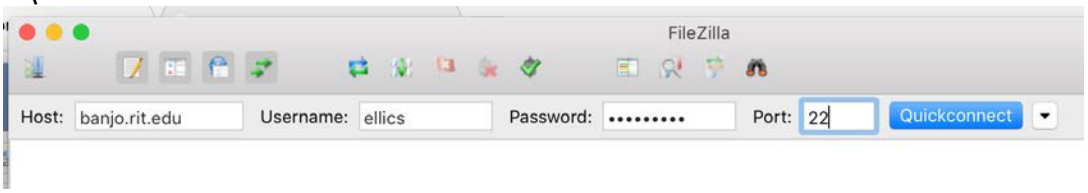
5. In the index.html file, and add a link at the end of that page to your new HTML file, like this:
   My Second Page

6. Go back to the browser, reload your index.html page, and see if the links between your pages work properly. If they don't, ask for help!

Part 4: Putting Your Files Online using FTP

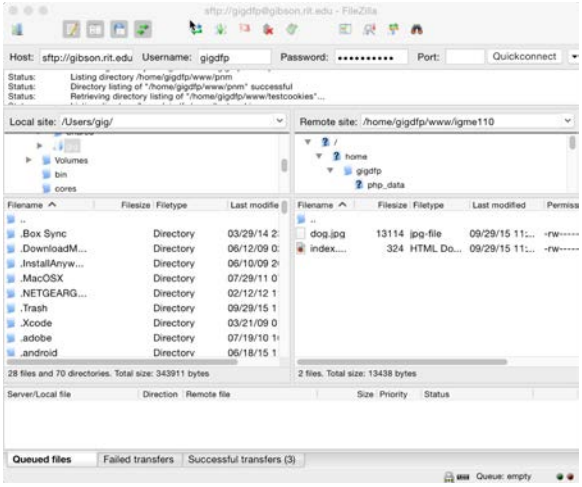
Now we'll use FTP (actually SFTP, which uses a secure connection to the server) to put your web pages onto the banjo server.

1. Launch FileZilla. Fill in the fields at the top with the following information, and click "Quickconnect":



2. FileZilla may ask you if you want it to remember passwords—if you're doing this in the lab, tell it no. (You don't want the next person using this computer to have access to your account!) It may also give you a warning about an "unknown host key"—if that happens, check the box saying "Always trust this host" and then click OK. If you entered your user ID and password correctly, you should now see something like the screen pictured to the right.

   The pane on the bottom left shows the files on your computer's hard drive. In this case, it's showing everything, including hidden files, that are on the



main level of my computer's hard drive. The pane on the right shows all the files in your directory on RIT's web server.

3. In the pane on the left side (the local site files), you'll need to find the directory that you created at the beginning of this exercise. If you put it on a thumb drive, the drive should show up in the Volumes directory. If you put it on the Desktop, it should be in the student directory inside of the Users directory. You want to find your www folder and open it, so that you can see the igme110 and media folders in the left pane. (If you're using Mac, there may also be a hidden file called .DS_Store, which you can ignore—it's a Mac system file that is hidden.) If you have trouble finding your files, ask for help.

4. In the pane on the right side, double click on the www directory. You should see the 110 directory that you created in Tuesday's exercise. Leave that alone until you've gotten credit for the exercise!

5. Once you've got the www directory on your local computer on the left side, and the www directory on banjo on the right side, you're going to drag the igme110 and media directories from the www folder on the left to the www folder on the right. This will copy the folders and their contents to the web server. The file list on the right side should update to show the the new directories.

6. Now you need to test the files on the web server, to see if they're accessible.

   Use a browser to go to http://people.rit.edu/youruserid/igme110

   If the files and images show up, great! But it's possible that they won't, because the access to the files isn't set properly by default. I'll review how to change permissions on files in our class, but for reference purposes, here's how to do it:

   In FileZilla, right click on the folder in the right pane, choose "File Permissions" and make sure the permissions include read write and execute for the owner (that's you), and read and execute for everyone else. The number in the box at the bottom should read "755" which is shorthand for those permissions. (You can either type the number into the box at the bottom, or check the boxes next to the permissions you want.)

   Once you've checked the permissions on the folders, you need to also check the permissions for the individual files. Double click on a folder to open it, and repeat the process of right-clicking and choosing File Permissions for each of the individual files. The HTML and image files don't need execute permissions, so the shortcut for their permissions is 644 rather than 755.

   Once you're sure all the files have the right permissions, go back to the browser and try loading the URL above one more time.

7. Notice that the URL I gave you only has the directory name (igme110) and not a file name. That is because banjo, like most web servers, automatically looks for a file called index.html

in a folder if no other file was specified. If you were to include the file name, like this: http://people.rit.edu/youruserid/igme110/index.html

it would also recognize that file and load it. But if you end the URL with the directory name and don't specify a file, the server will automatically load the index.html file if there is one in that directory.

Submitting your work: You don't need to do anything other than upload your pages to the server and test them using the instructions above. We will go to the URL that your files are *supposed* to be located at, http://people.rit.edu/youruserid/igme110 (where "youruserid" is your actual RIT user ID), and check to make sure that the required pages display. Please have the file online *no later than* noon on Saturday.
