Setting up your nrs-projects web directory

We have the access to create web pages on `nrs-projects` that you should be able to access from the web. This walks through the basics for setting this up.

BACKGROUND - Fun facts, part 1 (info, no actions here yet)

- nrs-projects is set up so that individual users may put files they'd like accessible from the web into a subdirectory, in their home directory, whose name is **`public_html`**

- For the web server process to access a file, it must be:

  - in this **`public_html`** directory or in a **subdirectory** of **`public_html`**

  - at least **world-readable**

  - in a subdirectory such that **EVERY directory in its path** is at least **world-executable**

TO DO: Set up your `public_html` directory

To start, make sure that you have a **`public_html`** directory in the right location, set up with appropriate permissions. The following steps should make sure this is the case:

- Use ssh to connect to your account on `nrs-projects`**`-ssh`**`.humboldt.edu`

- Make sure that your **home directory** is world-executable. (**~** is a nickname for your home directory.):

  `ls -ld ~`

  - IF the permissions you see end in **`--x`** (for example, `drwx--x`**`--x`**) **or** **`r-x`**, then you are fine.

  - IF **NOT**, use this command to make your home directory **world-executable**:

  `chmod 711 ~`

  - Now **verify** that your home directory's permissions end in **`--x`**:

  `ls -ld ~`

- See if you have a **`public_html`** directory. To see if you have this directory, use the command:

  `ls -ld ~/public_html`

  - IF you get a message back complaining that "`ls: cannot access public_html: No such file or directory`", then you need to create this directory. Do so using this command:

  `mkdir ~/public_html`

  - Now verify that it has been created:

  `ls -ld ~/public_html`

- When you just tried:

  ```
  ls -ld ~/public_html
  ```

  ... CHECK to see if **IT** is world-executable -- that is, check that its permissions end in either **--x**
  (for example, drwx--x**--x**) **or r-x**

  - IF they don't, make it world-executable with the command:

    chmod 711 ~/public_html

  - Now again do:

    ls -ld ~/public_html

    ...and verify that it is now world-executable (that its permissions end in **--x**)

MORE BACKGROUND - Fun facts, part 2 (for your information)

- IN GENERAL: Say that there is an nrs-projects user whose username is *ab123*.

  When a URL (Uniform Resource Locator) **starts** with:

  https://nrs-projects.humboldt.edu/~*ab123*

  ...then nrs-projects' web server process looks inside *ab123*'s public_html directory for a file.

  - WHICH file? That depends on what follows!

- if NOTHING follows -- if the URL is indeed JUST

  ```
  https://nrs-projects.humboldt.edu/~ab123
  ```

  - ...then the web server process looks for a file with the special name **index.html** (or
    index.php) within *ab123*'s public_html directory, and tries to send that index.*'s
    contents as the request result back to the requesting browser.

  - NOTE that index.html **must** be world-readable! You can make it so using:

    ```
    cd ~/public_html       # make sure you are in public_html
    ```

    chmod 644 index.html

- if a plain file name follows -- if the URL is something like:

  https://nrs-projects.humboldt.edu/~*ab123*/*desired_file.html*

  - ...then the web server process looks for a file named *desired_file.html* within *ab123*'s
    public_html directory, and tries to send *desired_file.html*'s contents as the request result back
    to the requesting browser.

  - NOTE that *desired_file.html* must be world-readable! You can make it so using:

    ```
    cd ~/public_html       # make sure you are in public_html
    ```

    chmod 644 *desired_file.html*

- if a SUBDIRECTORY path name follows -- if the URL is something like:

`https://nrs-projects.humboldt.edu/~`*`ab123/subdir1/subdir2`

- ...then the web server process looks for a file with the special name `index.html` (or `index.php`) within *ab123*'s `public_html`/*subdir1*/*subdir2*, and tries to send that `index.*`'s contents as the request result back to the requesting browser.

- NOTE that *subdir1* must be world-executable,

   subdir2 must be world-executable, AND

   `index.html` within *subdir2* must be world-readable! You can make these so using:

`cd ~/public_html`	# make sure you are in `public_html`
`chmod 711 `*`subdir1`*	# make *subdir1* (within `public_html`) world-executable
`cd `*`subdir1`*	# change directory to *subdir1* (under `public_html`)
`chmod 711 `*`subdir2`*	# make *subdir2* (within *subdir1*) world-executable
`cd `*`subdir2`*	# change directory to *subdir2* (within *subdir1*)
`chmod 644 index.html`	# make `index.html` (within *subdir2*) world-readable

- if a relative path name for a plain file follows -- if the URL is something like:

`https://nrs-projects.humboldt.edu/~`*`ab123/subdir1/subdir2/looky.html`*

- ...then the web server process looks for a file named *looky.html* within *ab123*'s `public_html`/*subdir1*/*subdir2*, and tries to send *looky.html*'s contents as the request result back to the requesting browser.

- NOTE that *subdir1* must be world-executable,

   subdir2 must be world-executable, AND

   looky.html within *subdir2* must be world-readable! You can make these so using:

`cd ~/public_html`	# make sure you are in `public_html`
`chmod 711 `*`subdir1`*	# make *subdir1* (within `public_html`) world-executable
`cd `*`subdir1`*	# change directory to *subdir1* (under `public_html`)
`chmod 711 `*`subdir2`*	# make *subdir2* (within *subdir1*) world-executable
`cd `*`subdir2`*	# change directory to *subdir2* (within *subdir1*)
`chmod 644 `*`looky.html`*	# make *looky.html* (within *subdir2*) world-readable