

cis20.2-spring2010-sklar, lab II.3
first php exercise

A. setting up php

1. Log into the class web server using **putty (ssh)** and your username and password.
The IP address of the web server is: 146.245.250.181
2. Create a directory called `public_html`:
`unix-prompt$ mkdir public_html`
3. Set its permissions to be owner everything, group and world readable and executable:
`unix-promt$ chmod 755 public_html`
4. Create a simple test HTML file called `index.html` in your `public_html` directory, e.g., something like this:

```
<html>
<body>
hello and welcome to my web page!
</body>
</html>
```

The editors **nano**, **emacs** and **vi** are all installed on the class web server. You can also edit files on your workstation or laptop and then ftp them to the web server.

5. Make sure that your simple test HTML file works by opening a browser window and going to:
`http://146.245.250.181/~your-username/index.html`
If this part doesn't work, ask me for help! You CAN'T get any further with the lab if this part does not succeed!

6. Create a simple test PHP file called `p1.php` in your `public_html` directory, e.g.:

```
<?php print "hello\n"; phpinfo(); ?>
```

7. Make sure that your simple test HTML file works by opening a browser window and going to:
`http://146.245.250.181/~your-username/p1.php`
If this part doesn't work, ask me for help! You CAN'T get any further with the lab if this part does not succeed!

8. Add a link in your `index.html` file labeled "hello" that invokes `p1.php` when it is clicked.

Note that your files must have an extension of `.php` in order for the browser to recognize that there are PHP commands inside it that need to be interpreted.

B. exploring php

1. All the PHP examples from class have been posted on the syllabus portion of the web page, under March 9 (<http://www.sci.brooklyn.cuny.edu/~sklar/cis20.2>). Try them out from the class web page.
2. Then download the source code and try them from your page on the class web server. When you download the source, note that the files are named with a ".txt" extension. You will have to rename them to end in ".php" before you can run them.
3. Modify the **colors** example so that it cycles between 3 different colors instead of 2.
4. Modify the **colors** example so that it displays a checkerboard pattern in a table with 4 rows and 4 columns.

5. Modify the **arrays** example so that it displays the names of the people in your family and their birthdays. *Hint: the outer (states) array should be changed to contain your family members' names, and the inner (cities) array should be changed to contain two numbers per person — the month and day of their birthday.*
6. Modify the **userclass** file so that it contains a person's name, birth month and birth day. The modify the **myclass** file so that it instantiates a **user** object containing your name and birthday.
7. Modify the **myclass** file further so that it instantiates an *array* of **user** objects, each containing the name and birthday of people in your family.
8. Modify the **lunch** example so that it uses different types of input widgets on the HTML form. The example contains three **text** widgets and a **submit** widget. The other HTML form widgets are:

- `<input type="checkbox" ... >`
- `<input type="file" ... >`
- `<input type="password" ... >`
- `<input type="radio" ... >`
- `<select ... ><option> ... </select>`
- `<select ... multiple><option> ... </select>`
- `<textarea ... >`

Experiment with adding more widgets to the HTML form, then passing the information entered on the form over to the PHP page, and finally displaying the information entered from the PHP page.

A tutorial on HTML Forms is available here:

<http://htmldog.com/guides/htmlbeginner/forms/>