

CNMT 310: Session Assignment  
Points & Due Date: See Canvas

This is an individual assignment. No group work or collaboration should be used for this assignment.

## Details:

Create a new folder called `session` to hold the files from this assignment. All other filenames should remain the same from the original Bookmarks assignment. See Notes section for further clarification around the requirements for this assignment.

- Update the bookmark page assignment to include error messaging and to enable bookmarks to be displayed based on the user who is logging in. For this assignment, you will utilize a pre-built class with three users for authentication called `User.class.php`
- Add logout functionality by displaying a link to `logout.php` when the user is logged in. The link should appear within the navigation header. Use the code given in class.
- Place `session_start()` within your `autoload.php` file and `require_once()` the `autoload` file within all files that require it.
- Replace the destination for all Location redirects with constants. For example, rather than using

```
header("Location: index.php");
```

you would use

```
header("Location: " . HOME);
```

Be sure to `define()` the constants within your `autoload.php` file.

- Move the final version of the `getBookmarks($userid)` function into the `User` class. You should account for execution paths where the user is not found or has no bookmarks.
- Where necessary, redirects should include error messaging stored in session and it is recommended to use the format shown in class. The destination page for the redirect should display the error messages, if they are present. This will require changing some of the session code that was "given" for the previous assignment. Consider the user experience of submitting a form, getting an error, resubmitting, and being shown an

error in the next field. Yuck! Instead, avoid redirecting until you have gathered all of the errors into session and then redirect as needed, as demonstrated in class.

## Notes:

User	Password
jim	jimpass
bob	bobpass
mary	marypass

Do not hardcode or plan on knowing the number of bookmarks that will be returned. You may find that the user does not have any bookmarks defined!

Never output any part of the HTTP response directly from a function or a class method to the browser. Use `return` instead.

Consider where `User.class.php` needs to be required. Does it need to be required in every file?

Other than the changes specified, no other changes should occur within the User class.

All pages must be valid HTML along all execution paths and PHP must be used to produce all HTML.

No inline CSS or JavaScript or event handlers should be used and both CSS and JavaScript (if any) must be contained in external files. No JavaScript is required for this assignment.

Code must be committed using the git command line interface.

Code efficiency and maintainability are two important factors when assessing this assignment. Be deliberate and intentional about each line of code and avoid repetition where possible.

When creating the test plan, it will be up to you to determine if the high-level requirement (RQ-1) from the previous assignment has changed or if any additional high-level requirements need to be added.

The `getBookmarks()` method is not a user-related function and thus we are violating a best practice by placing it with the User class. A future assignment may correct this when we add more detail around object-oriented PHP.

## Deliverables:

A commit ID corresponding to the committed code.

A test plan reflecting the requirements, test cases, and test execution for each case.