

Principles of Web Development

Mini Assignment 4

Due: February 24, 2022 at 11:55 PM on myCourses

This is not a responsive design assignment (do not add any responsive design elements). This is not an interactive assignment (eg. Mouseover, but you can add interactive elements if you want, but they won't be graded).

The purpose of this assignment is to practice a REST POST call from a webpage to a backend PHP program using a <form> running from your CS account on mimi in your public_html directory.

Do the following:

1. Make sure your CS account works.
2. If you do not have a public_html directory, please create one. Just use `mkdir`. Follow this tutorial if you need help: <https://www.cs.mcgill.ca/docs/tutorials/website/>.
3. Inside public_html, create a subdirectory called mini4. You will put all your code in there. You will be able to access the code from your browser using: www.cs.mcgill.ca/~yourCSusername/mini4/yourwebpagename.html.
4. You will create two pages: mini4.html and mini4.php. The HTML page will contain a <form> that will use a POST request to the PHP program. The PHP program will process the request and display a response to the browser.
5. Create the following webpage as a <form> within the file mini4.html. Unlike the previous assignments, you do not need to make it look exactly the same, but it does need to have all the elements depicted. See the image below:



The image shows a web browser window with the title 'Sample Form'. The address bar shows 'localhost/ch19/fig19_13-14/form.html'. The form is titled 'Registration Form' and includes the instruction 'Please fill in all fields and click Register.'.

User Information

First name:
Last name:
Email:
Phone:

Publications

Which book would you like information about?

Operating System

Which operating system do you use?
☒ Windows ☐ Mac OS X ☐ Linux ☐ Other

6. Your <form> must call `mini4.php` from your SOCS account.
7. Note: for the dropdown list of books, you can add any of your favourite books to the list. There should be a minimum of 3 books. The books do not need to be computer science texts.
8. Your PHP program will do two things.
 - a. Each time the PHP program is called it will **append** all the information from the form into a file called `mini4.csv`. Each “row” of the file will be a CSV (comma separated vector) of the input from the HTML <form>. This CSV row does not need to follow all the CSV rules. Simply add a comma between each field without any further processing. For example:


```
fname,lname,email,phone,book,os
```
 - b. Then, the PHP form will return a response to the browser listing **all** the CSV rows in `mini4.csv`.
9. Run your program a few times. When the TA tests your program, they should see all your previous appends and then the multiple appends they added when testing your code.

WHAT TO HAND IN

- The files `mini4.html`, `mini4.php` and `mini4.csv`. **ZIP all the files.**
- Also, the file `README.txt` stating:
 - Which browser you used.
 - The URL to your program running on mimi because the TAs will need to see it run.
- **Do not make the website responsive.**
- You are permitted to use everything you have learned from class in your submissions.

HOW IT WILL BE GRADED

This mini assignment is worth 20 points and part marks can be awarded.

Deduction points:

- -3 for not following instructions
- Late penalty points
- You will lose points for adding new material not covered during class.

Awarding points:

- +5 <form> and <input> tags work correctly and as specified
- +3 HTML source code is pretty (indentation, spacing, comments if needed, your name & ID)
- +2 HTML page displays in a nice way (does not need to be very pretty)
- +3 The HTML calls the SOCS backend PHP program properly and it works
- +4 PHP appends to CSV and replies to the browser as specified (do not need to be pretty)
- +3 PHP source code pretty (indentation, spacing, comments if needed, your name & ID)