Discussion Directions

*Please refer to the general homework directions in the course and from week 1!*

Homework 7

**Learning Outcomes**

1. Discuss the security issues with using third party web programs like JavaScripts
2. Implement a JavaScript program from your textbook as practice.
3. Create a complete web site using, HTML 5, CSS 3, Forms, and JavaScript

==================================================

 Part 1

*Learning Outcome*

1. Discuss the issues with using third party web programs like JavaScripts

*Directions*

There are many web sites with third party plug-ins and scripts. However, programmers do not always know if they are safe. OWASP Top Ten is a project to identify web application security issues.

OWASP. <https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project>

We know many sites have been hacked.

Korolov, Maria. (Mar 24, 2015). CSO | Study: One-third of top websites vulnerable or hacked. <http://www.csoonline.com/article/2900449/browser-security/study-one-third-of-top-websites-vulnerable-or-hacked.html>

OWAP tells us that XSS is number two in the list of the top ten. Review the following information sheets about security risks with web sites.

* 1. <https://www.owasp.org/index.php/Cross-site_Scripting_(XSS>)
  2. <https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet>
  3. <https://www.owasp.org/index.php/XSS_Filter_Evasion_Cheat_Sheet>
  4. <https://www.owasp.org/index.php/Top_10_2013-A1-Injection>

Many web sites show how XSS occurs such as <http://www.hacking-tutorial.com/hacking-tutorial/basic-hacking-via-cross-site-scripting-xss-the-logic/>. So hackers know ‘how’ to do this. So how do we fight back? By learning to prevent the attack. XSS attacks can be persistent where the information is saved such as in a database or non-persistent.

1. Read: Cross Site Scripting (XSS) Complete Tutorial for Beginners~ Web Application Vulnerability. (October 14, 2011) <http://breakthesecurity.cysecurity.org/2011/10/cross-site-scriptingxss-complete-tutorial-for-beginners-web-application-vulnerability.html>
2. What is **cross site scripting** (XSS) and how can it be prevented? (Make sure not to ‘copy’ the information directly from the page, but rather ***explain*** your answer and your reasoning.)

*Part 1 Submission*

1. Place your responses in Part 1 of the Word document.

==================================================

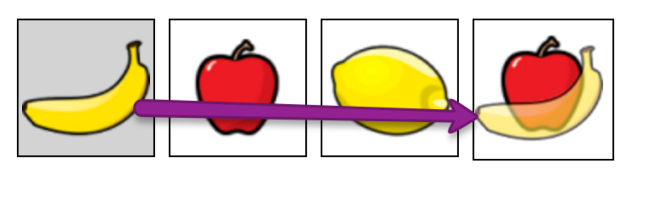
 Part 2

*Learning Outcome*

1. Implement a JavaScript program from your textbook as practice.

*Directions*

The book has many examples (<http://www.apress.com/9781430239604?gtmf=s>). Review **listing-05.html** from **Chapter 37.** Create a new page **puzzle.html** by duplicating the about page and renaming it. It is already in the menu. Use the drag and drop example as a **book** to guide you. But make sure to write your own code! I expect to see comments thoroughly documenting the game. Warning: Most of the web sites listed on Code Project and Stack Overflow are way more advanced then what you are asked to here. Do your own work!



1. Create a new page called **puzzle.html** from your template.html page.
2. Create an image from the **Veteran’s History section** from the **Library of Congress** site, and trim it into 4 pieces or more. Save each piece as a separate image. You can use most graphic editors like **GIMP** (free) to do this.
3. Instead of one target to frag the image to, create 4 squares!
4. Make sure to store all your JavaScript in a file called **puzzle.js.**
5. Allow the user to drag and drop the pieces of the **image**, to each of the squares, and put it together, like a puzzle. You can optionally have the background image as a faded version of the image so you can give the user an idea of what the finished **image** would look like.
6. When the user has all 4 matching, let them know by **alerting** them with a message.

*Part 2 Submissions*

1. Take screen shots of the **puzzle.html** and insert in Part 2 of the Word document
2. Copy the code from the **puzzle**.**html** you edited and insert in Part 2 of the Word document
3. Copy the code from the **puzzle**.**js** and insert in Part 2 of the Word document

==================================================

 Part 3

*Learning Outcome*

3. Create a complete web site using, HTML 5, CSS 3, Forms, and JavaScript

*Directions*

1. Create an external JS file named **login.js**.
2. Create a new page called **login.html**. Again, you can copy the template.html page. Link the **login.js** file to the **login.html** file.
3. Create a new page called **private.html**. Again, you can copy the template.html page.
   1. Add an **ordered list** of your 5 recommended books from the Gutenberg Project.
4. Open the **login.html** page. The user must fill out their username and password.
   1. If the information is correct, then:
      * Write out a cookie called **Library**.
      * In the cookie set the expiration date for **14 days from today**, (so you have to calculate the date) and assign values from the form to the cookie for the username, current date and password).
      * Use the **alert** method to let the user know that the cookie was written to their computer.
      * Redirect the user to the **private.html** page using **location.href** or other JavaScript based method. If the information is not correct, then redirect them to the **home.html** page.
   2. If the information is not correct, then:
      * Use the **alert method** to let them know there was an error.
      * Redirect the user to the **home.html** page.
5. Open the **recommended.html** page.
   1. Link the **login.js** file to the **private**.**html** file.
   2. Write JavaScript code in the **login.js** file to retrieve the write the values from the cookie when the page opens, and write the values onto the page, in a **label** or **div** tag, in the main content area of the page.

*Part 3 Submissions*

1. Take screen shots of the **login.aspx** and **private.html** and insert in Part 3 of the Word document
2. Copy the code from the **login.js** you edited and insert in Part 3 of the Word document

**\*\*\*\*\* Don’t forget to submit the Word document with the entire project folder in a zip file \*\*\*\*\*\*\*\***

**Homework 7 Grading Criteria**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Part** | **ULO** | **Exceeds** | **Meets** | **Does not Meet** | **No Evidenc**e |
| Part 1  Discuss the issues with using third party web programs like JavaScripts | 1 | Thorough explanation with references cited properly | Basic explanation with references cited properly | Basic explanation with no references or not cited properly | \* |
| Points | | 40 | 28 | 24 | 0 |
| Part 2  Implement a JavaScript program from your textbook as practice | 2 | **Book puzzle** game works with no errors.  **puzzle**.**html**  **puzzle.js** | **Book puzzle** game has minor errors. | Did not complete the **Book puzzle** game, major errors or copied from other sources | \* |
| Points | | 40 | 28 | 12 | 0 |
| Part 3  Create a complete web site using, HTML 5, CSS 3, Forms, and JavaScript | 3 | Created the form and processed the form data. Files named correctly.  Read and write to cookie correctly. Redirection worked. Values displayed to the user.  **login.aspx**  **login.js private.html** | Created the form and processed the  form data correctly.  Files not named correctly or did not read or write to cookie correctly.  Redirection did not work or the values not displayed to user | Created the form but no form processing or redirection.  Files named incorrectly.  Read or write to cookie incorrectly | \* |
| Points | | 60 | 42 | 36 | 0 |
| Total | | 100 | 70 | 60 | 0 |