**Pusat Pengajian Informatik dan Matematik Gunaan, Universiti Malaysia Terengganu (UMT)**

**Lab Manual 3 Week 6**

**Web Programming**

|  |  |
| --- | --- |
| **Name:** | **Arun Mugilan a/l Sargunan** |
|  |  |
| **Semester:** | **3** |
| **Lab:** | **MP3** |
| **Demonstrator** | **Mr. Arizal** |



# Instructions:

This lab manual is for use by students of the School of Informatics and Applied Mathematics, Universiti Malaysia Terengganu (UMT) only. Not allowed to print and distribute this manual without official permission of the author.

# Please follow the step by step as described in the manual. Tick ​​(√) each step has been created and write the conclusion of each activity has been completed. Conclusion of each activity must be written in a file named according to the following format:

# [nomatrik] \_ [LabX]. docx. Replace the 'X' with your lab session. Examples of correct file name is 'UK12345\_Lab1.docx'.

# Please prepare assignments with individual effort without any 'copy-paste' from other parties.

Best of luck!

## Elemen Blok *(block)* dan Sebaris *(inline)* Di Dalam HTML

|  |  |
| --- | --- |
| **Objective:** | How to define elemen block and inline in HTML. |
| **Explanation:** | HTML elements basically consist of block or inline elements. Block elements are elements that are displayed with a new line in the browser. The block elements you have used are like <h1>… <h6>, <p>, <ul>, <ol> and <table>. While inline elements are elements that are displayed on the same line in the browser. Examples are <b>, <d>, <td>, <tr> and <img>.  An <div> element is a block element that can be used to group HTML elements in a group. It will simplify the decorating process and set the style of a website. <div> is widely used in CSS topics that you will learn later. Using the <div> element, each element displayed will have blank rows before and after the display.  In addition, the <span> element is specially used to collect text elements. Technically, <div> and <span> have no specific purpose except for the purpose of decorating HTML elements. |

|  |  |  |  |
| --- | --- | --- | --- |
| **STEP-BY-STEP** | | | **Finish? (√)** |
| Step 1: | | Create a file named block.html. | / |
| Step 2: | | Type the html code below into a new file that is created. | / |
| Step 3: | | Save block.html file and click on the file to see the output. | / |
| Step 4: | | Study the HTML code that has been written line by line. Customize the colors used according to the color codes you have learned before. | / |
|  | **Conclusion:** |  | |

## Susun Atur (Layout) Using <div>

|  |  |
| --- | --- |
| **Objective:** | Use <div> to organize the website. |
| **Explanation:** | Previously, you used <table> to arrange the form. Apart from <table>, <div> can also be used for the same purpose. All you have to do is set the style for the blocks created. The activity below is to create a website consisting of headers, menus, content and footers. |

|  |  |  |  |
| --- | --- | --- | --- |
| **STEP-BY-STEP** | | | **Finish? (√)** |
| Step 1: | | Using Notepad, create a file called div.html. | / |
| Step 2: | | Type the HTML code below into the newly created file. | / |
| Step 3: | | Save the file and click on the file icon to see the display results. | / |
| Step 4: | | In this activity, there are many attributes for the style you use such as background-color, height, width, text-align and float. Try modifying the values ​​of these attributes and note the results. Don't worry about the messy output as long as you understand what the attributes used are doing. | / |
|  | **Conslusion:** |  | |

## Entities In HTML

|  |  |
| --- | --- |
| **Objective:** | Understand the use of entities in HTML. |
| **Explanation:** | Entities are used to display reserved symbols in HTML. For example, the symbols ‘<’ and ‘>’ are reserved symbols and we cannot write as follows so that the ‘<’ sign is displayed on the browser:  <html>  <body>  This is a sign ‘<’ </body>  </html>  This is because the browser will assume that the ‘<’ symbol is part of the HTML tag. Therefore, entities should be used to display reserve symbols in HTML code.  Entities can be used according to the name or number specified. For example, the symbol ‘<’ can be represented by & lt; or & # 60; . The following table shows some of the HTML entities that can be used:   |  |  |  | | --- | --- | --- | | **Symbol** | **Entities Name** | **Entities Symbol** | | £ | &pound; | &#163; | | € | &euro; | &#8364; | | © | &copy; | &#169; | | ® | &reg; | &#174; | | < | &lt; | &#60; | | > | &gt; | &#62; | | Blank space without a newline | &nbsp; | &#160; |   Note: Entity names use lowercase letters as a whole and are case-sensitive. Do a search on the internet for a full list of usable entities. |

|  |  |  |  |
| --- | --- | --- | --- |
| **STEP-BY-STEP** | | | **Finish? (√)** |
| Step 1: | | Create a file name entity.html. | / |
| Step 2: | | Type the code below into the newly created file. | / |
| Step 3: | | Save the file and click on the file icon to see the display results. | / |
| Step 4: | | Do a search on the search engines for a list of other entities and experiment with those entities. | / |
|  | **Conslusion:** |  | |

# Introduction to HTML5 (Hypertext Mark Up Language 5)

## Introduction to HTML Versions

|  |  |  |  |
| --- | --- | --- | --- |
| **Objective:** | | Understanding HTML Versions | |
| **Explanation:** | | Before we learn HTML5 in more depth, we should first get to know the existing HTML versions. As we learned in the lecture, HTML has various versions that change from year to year. In the early stages of the emergence of HTML, it was only known as HTML without any version number at the end. However, starting in 1995, HTML was first given a version number with the name HTML2.0, followed by HTML3.2 (1997), HTML4.01 (1999) and most recently, HTML5 (2012).  In addition to HTML, there is another version called XHTML. XHTML is an improved version of HTML4.01 by W3C in 2000. So what is the difference between HTML and XHTML? Simply put, XHTML is a more strict version of HTML where XHTML writing has to comply with certain requirements. Among them:   * Every web document written needs to be declared with <! DOCTYPE>. Example:   <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">   * XHTML tag pair block blocks need to be sorted correctly. For example:<b><i> This text is bold and tilted.</i></b>   This is because in the previous HTML, the following order is still being processed correctly by the browser: <b><i> This text is bold and tilted.</b></i>   * Every XHTML element needs to be closed including an empty element. For example, <br> should be written as <br/>. * XHTML elements need to be written in lowercase. For example, <p> </p> instead of <P> </P> * Attribute names also need to be written in lowercase. For example, <table width = "70%"> instead of <table WIDTH = "70%"> * The value for the attribute needs to be written inside the quote. For example, <table width = "70%"> instead of <table width = 70%>   In addition, XHTML was introduced to support XML-based HTML writing where, if HTML requires new tags, then it can be declared in a DTD (Document Type Declaration) file. In XHTML web documents, the DTD file in question can be accessed at http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd. Discussions on XML and DTD will be in the next topic.  So, what is the appropriate version of HTML to learn and use? As of this writing, it is HTML5. HTML5 is the latest version of HTML that to some extent uses the features available in XHTML. Furthermore, it is considered easier than the use of XHTML which is considered too strict for some website developers.  HTML5 was produced as a result of a collaboration between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG) which started in 2006. Among the advantages of HTML5 are:   * Reduce the need for external plugins (such as Flash) * Improved error handling. * More tags to replace scripts (JavaScript for example). * Ability to work on multiple devices. For example, smartphones, tabs and so on. * A simpler <! DOCTYPE> declaration of <! DOCTYPE html> than previous HTML or XHTML versions.   The lab activities you have done before are using the original HTML version which is also the basis of the latest HTML versions. Therefore, the next activities are based on the latest version of HTML, HTML5. | |
| **STEP-BY-STEP** | | | **Finish? (√)** |
| Step 1: | Create a new file named intro5.html. | | **/** |
| Step 2: | Type html code below: | | **/** |
| Step 3: | Save the newly created file and double-click on the file icon to view the output. You will find that the browser gives the correct display on the screen. However, does it comply with the rules for HTML5? To be sure, we can use the free service offered by W3C at http://validator.w3.org/. | | **/** |
| Step 4: | Click on the 'Validate by File Upload' tab. | | **/** |
| Step 5: | Click on the ‘Choose File’ button and select the previously created intro5.html file. Press the ‘Validate’ button. What display did you find? Does your HTML5 file comply with the requirements? If not, what are the mistakes made? Please note in the conclusion section. | | **/** |
| Step 5: | Fix the previous HTML code with the following code:    Upload the file to http://validator.w3.org/ and make sure your HTML code has passed the validation test. Note: If you see a ‘warning’, just ignore it. | | **/** |
| **Conclusion:** | After modification | |  |

## Elemen-elemen Borang *(form)* in HTML5

|  |  |  |  |
| --- | --- | --- | --- |
| **Objective:** | | Recognize form elements in HTML5. | |
| **Explanation:** | | The HTML elements previously studied can still be used in HTML5. The most important thing to remember is that the use of <! DOCTYPE html> is mandatory in every document that uses HTML5. In addition, some of the HTML5 elements that will be used in subsequent activities are not supported by all browsers.  Among the significant improvements made in HTML5 are related to form (form). There are three new elements to the form introduced namely <datalist>, <keygen> and <output>.  The <datalist> tag is used to offer autocomplete features to the user when filling out a form. This will save time for filling in the blanks on the form.  The <keygen> tag is used to increase security during the verification process of users who want to log into a web system. When the form is sent to the server, there are two keys that will be generated, namely the public key and the private key. The Public Key will be sent to the server while the private key will be stored locally. This public key will be used to generate a digital certificate (digital certificate) that will be used in communication between the browser and the server in the future. If you would like more information on how Public Key Infrastructure works, please do a search on the internet.  Note: Because there are still tags that are not supported by all browsers. Please note the browser type used. If the activity requires Mozilla, use Mozilla, if the activity requires Chrome, please use Chrome. However, it is not a mistake if you want to experiment with whether a tag is supported by the browser or not. | |
| **STEP-BY-STEP** | | | **Finish? (√)** |
| Step 1: | Create a new file named list5.html. | | **/** |
| Step 2: | Type html code below: | | **/** |
| Step 3: | Save the newly created file. Click on the file icon to see the output. View the output of files you created using Mozilla or Chrome browsers. | | **/** |
| Step 4: | Enter the letter ‘T’ in the input box and note what happens. Then, try with other letters such as ‘M’, ‘P’ and ‘K’. Record your observations in the conclusion section. | | **/** |

|  |  |
| --- | --- |
| **Conclusion:** |  |

## 

|  |  |  |  |
| --- | --- | --- | --- |
| **Objective:** | | Applies the HTML5 <output> element | |
| **Explanation:** | | The <output> tag is used to display the output of a calculation usually performed using a script. This will make it easier for website builders to place computational output anywhere on the website view they build. | |
| **STEP-BY-STEP** | | | **Finish? (√)** |
| Step 1: | Create a new file named output5.html. | | **/** |
| Step 2: | Type html code below: | | **/** |
| Step 3: | Save the newly created file. Click on the file icon to see the output. View the output of files you created using Mozilla or Chrome browsers. This element is not currently supported by Internet Explorer. | | **/** |
| Step 4: | In the <form> element, oninput refers to which event is in the code above. When the user enters the input, then the operation in “z.value +…” will be executed. The <input> type range element is the element used to display the slider on HTML5 pages. In the above activity, the meaning of the attributes used are as follows:  ● min = smallest value on the slider  ● max = largest value on slider  ● step = increase value on slider  ● value = default value (default) on the slider.  The second <input> element is a number type. It will display a list of numbers. Each input element has an id as a reference for any upcoming operation.  For the <output> element, the values ​​in the for attribute refer to the id involved with the calculation. For example for = “x y” refers to the id x and y used in the addition operation as in the example above. | | **/** |

|  |  |
| --- | --- |
| **Conclusion:** |  |

## Types of Inputs In HTML5

|  |  |  |
| --- | --- | --- |
| **Objective:** | Apply the types of input in the HTML5 form. | |
| **Explanation:** | Inside a form, there are various types of input boxes that can be used to get input from users. Among the types of inputs offered in HTML5 are as follows:  ● number  ● range  ● search  ● color  ● date  ● datetime  ● datetime-local  ● email  ● month  ● search  ● tel  ● time  ● url  ● week  The input classification provided by HTML5 will facilitate the provision of space and validation of inputs on a website. However, not all HTML5 input types are currently supported by web browsers. | |
| **STEP-BY-STEP** | | **Finish? (√)** |
| Step 1: | Create new file named input5.html. | **/** |
| Step 2: | Enter the following code into the newly created file: | **/** |
| Step 3: | Save and view the output of files you create using the Chrome browser. | **/** |
| Step 4: | Test each of the input fields by filling out the form that has been created. For the Website and E-mail space, try experimenting with the following data:  Website: urlsaya  E-mail: my email  Notice what happens when you click the 'Submit' button. | **/** |
| Step 5: | Record the observations in the conclusion space. | **/** |

|  |  |
| --- | --- |
| **Conclusion:** |  |

## Attribut for Input HTML5

|  |  |  |
| --- | --- | --- |
| **Objective:** | Apply attributes placeholder, autofocus and required. | |
| **Explanation:** | Place *Placeholder* is a text box that has shadowed text when no value is entered and not focused. The term ‘focus’ refers to the cursor that flashes in a text box.  *autofocus* is an attribute used to make a text box focused as soon as the website is finished displaying in the browser.  *required* is an attribute used to require an input space to be filled with a value before being sent to a browser. | |
| **STEP-BY-STEP** | | **Finish? (√)** |
| Step 1: | Create new file named inattribute5.html. | **/** |
| Step 2: | Enter the following code into the newly created file: | **/** |
| Step 3: | Save and view the output of files you create using Chrome or Mozilla browsers. | **/** |
| Step 4: | Note the output displayed and note the position of the cursor. Which text box has a flashing cursor? Fill in the ‘Amen’ value in the ‘Student Name’ box and leave it blank in the ‘Father Name’ column. Then, click on the 'Save' button. What happened?  Refresh your web files and make sure both spaces are empty. Then, enter the letter ‘A’ into the ‘Student Name’ box. What did you find? Does autocomplete work? | **/** |
| Step 5: | Record the observations in the conclusion space. | **/** |

|  |  |
| --- | --- |
| **Conclusion:** |  |