Lab Objectives:

1) To become familiar with basics of webpage design.
2) To become familiar with the basics of HTML/ CSS and Javascript.

Hypertext Markup Language (HTML):

On the Internet, a website is represented by a URL (uniform resource locator). As mentioned in the class, the World Wide Web (WWW), also abbreviated as the web, is a system of interlinked hypertext documents that may be viewed through a web browser. We mentioned in the class lectures that Internet Explorer and Mozilla Firefox are examples of web browsers. To understand how WWW works, it's important to first define what hypertext documents are. We use the following definition of *hypertext*:

"*Hypertext* most often refers to text on a computer that will lead the user to other, related information on demand. Hypertext allows a form of user interface, which overcomes some of the limitations of written text. Rather than remaining static like traditional text, hypertext makes possible a dynamic organization of information through links and connections (called hyperlinks). Hypertext can be designed to perform various tasks; for instance when a user "clicks" on it or "hovers" over it, a bubble with a word definition may appear, a web page on a related subject may load, a video clip may run, or an application may open." [Wikipedia]

HTML is a hypertext *markup* language that is used to describe the structure of a webpage using tags. A web browser, an application that understands HTML tags and displays web pages accordingly, utilizes the HTTP protocol to receive HTML pages from a web server (also called HTTP server).

It is important to note that the display of an HTML page in a browser is not the same text that we see if we were to edit the HTML page in a text editor. This is because the HTML file itself contains many instructions/ tags that are not displayed in the browser. The function of these tags/ information is simply to enable a browser to display the HTML file. This should become clear as we go through the exercises.

HTML Resources:

The following are some tutorials on the topic of HTML. Tutorials on advanced topics such as JavaScript can also be found at the following sites:

1) http://w3schools.com/html/default.asp

2) http://inpics.net/htmlcss.html

Lab Exercise:

Part 1---Creating directory/ index.html and format.css files

A web server is an application that runs on TCP port number 80. It simply opens up a socket on the server's IP address' port 80 and waits for HTTP request. Examples of web server application software are the Apache Software Foundation's Apache and Microsoft's IIS.

A web server (like Apache and IIS) has to be told of certain parameters (called configuration). This configuration is usually done by the webmaster managing the website. The configuration parameters include 1) What IP address/ domain name is the server tied to; 2) What port number should be used (this is 80 by default)?; 3) What is the home directory of the web server; 4) What is the name of the default HTML file in a directory hosted on the web server.

When a request for a site is sent (without any document specified), the web server application sends the default file in the home directory of the web server. As an example, if we type http://www.niit.edu.pk in the address bar of Internet Explorer or Mozilla Firefox, the default page in the default directory is served. Note that in the mentioned example, only the domain (www.niit.edu.pk) is mentioned, but not the specific web page that should be fetched from this domain; in such cases, the `default' files as configured for the web

server will be fetched. Generally, the default file names start with **index** such as index.html, index.php, etc. *[PHP is a web scripting language that is outside the scope of this lab]*

In this part, we will create a directory in which we will save today's lab work. We will also learn how to create and save *index.html* and *format.css* files.

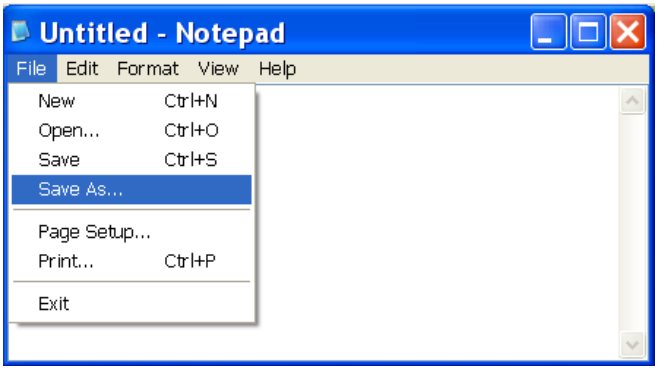
1. On the desktop, create a folder named ICTLab6

[***Hint:*** this can be done by right clicking on the desktop, and then choosing **New** and then **Folder]**

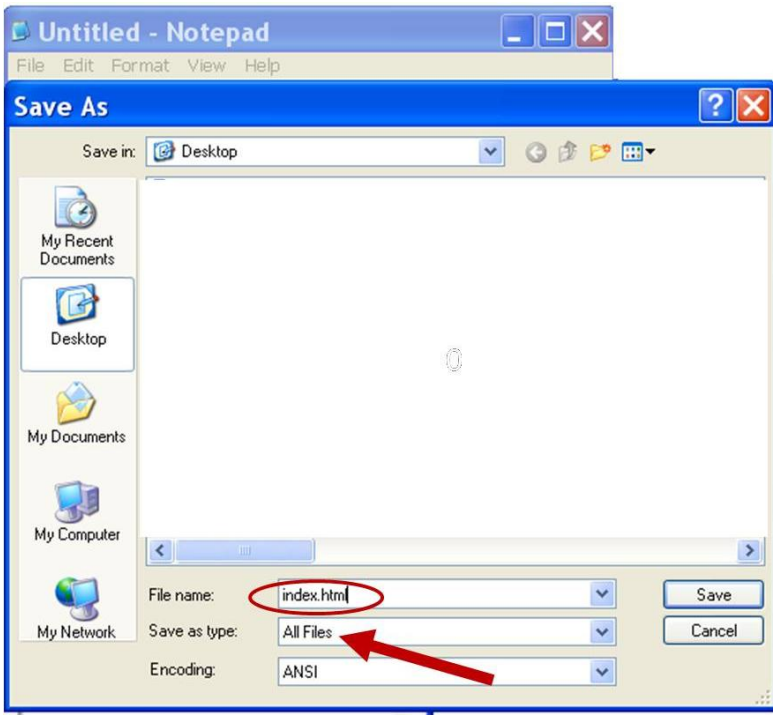
2. Create an index.html.

Open the Notepad application. [Start > Programs > Accessories > Notepad]

Save the empty file as index.html by clicking on the File menu of the Notepad.



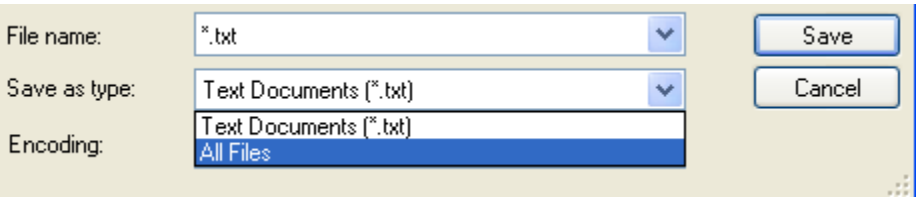
On the file menu, choose "Save As". The "Save As" window will appear as shown below:



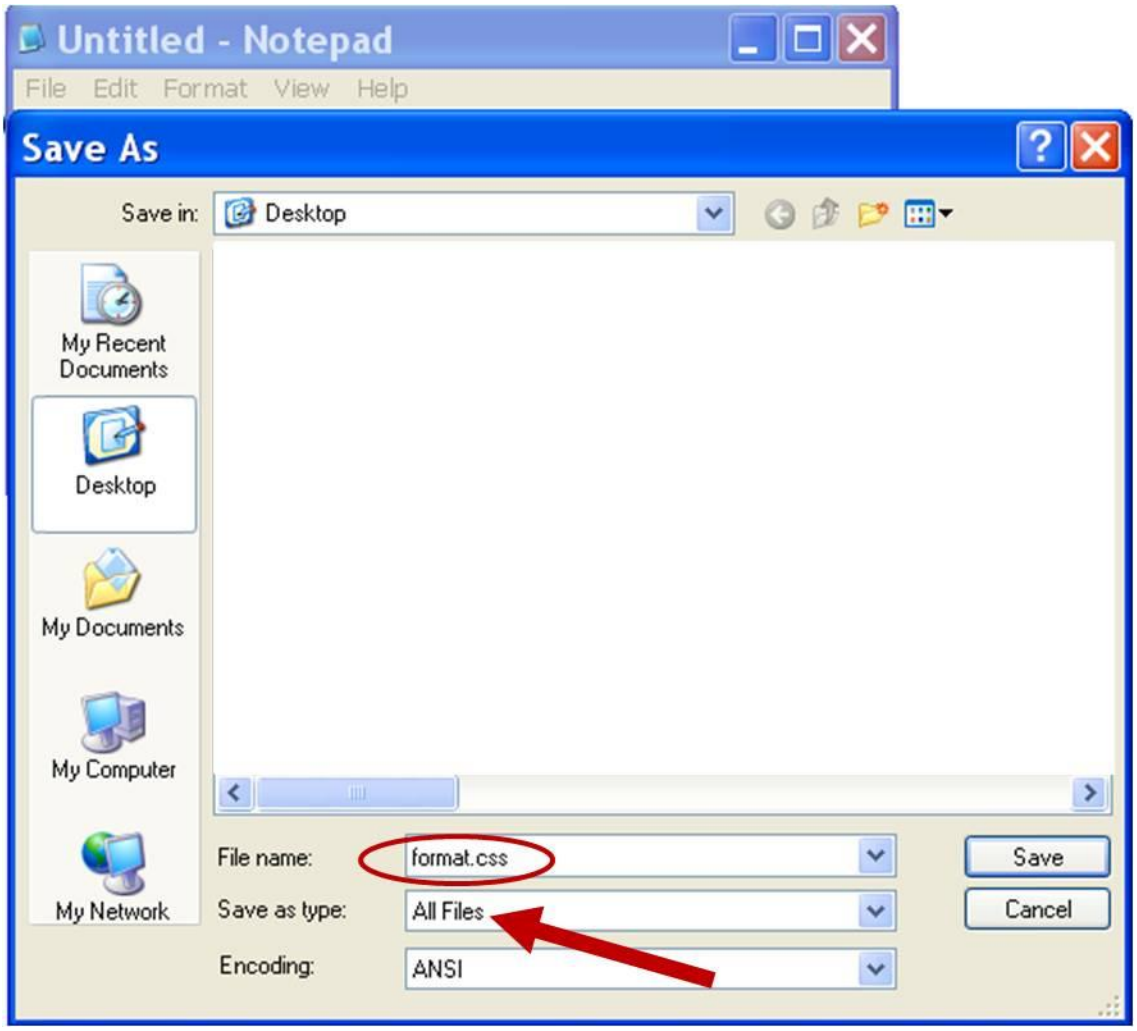
Save the empty file as index.html on the Desktop. [We will use the Desktop as HTML playground, and once the files have been finalized, then they will be transferred to the ICTLab6 folder.] To save the file as

index.html, index.html must be written as the file name with the "Save as type" set to *All Files*.

Note: As explained the "Save as type" should be set to *All Files*, if the "Save as type" field is set to *Text Documents (*.txt)* then the file would be saved as index.html.txt and not as index.html.



3. In a similar manner, create the format.css file. *[We will later see how this file is used]*

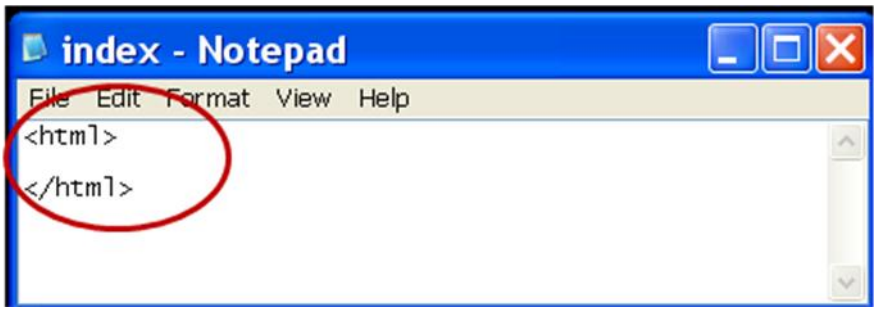


Part II---HTML basics

1. Writing the HTML basic tags

Open the index.html file on the Desktop, and enter the **<html>** and **</html>** tags on the file. Note that in HTML, tags often (but not always) follow the convention that there is an opening tag and a closing tag. The opening tag in our context is *<html>* and the closing tag is *</html>*. These tags are used to inform the

browser that the text between these tags is formatted according to HTML guidelines. The <html> tags are displayed in the figure below.



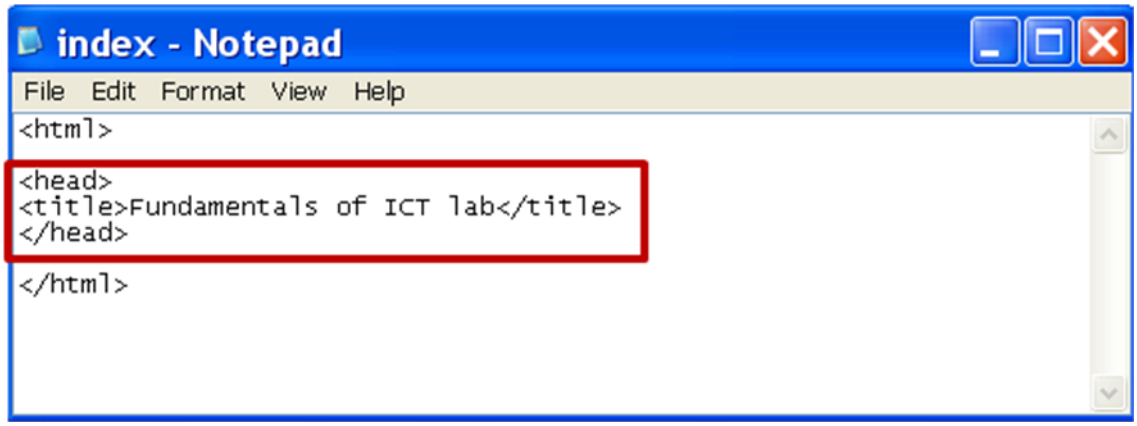
In the figure above, we do not have any HTML code in between the <*html*> tags. We will now start to code the HTML page.

2. Inserting the title of the web page

We will, first of all, learn the <*head*> tag and the <*title*> tag.

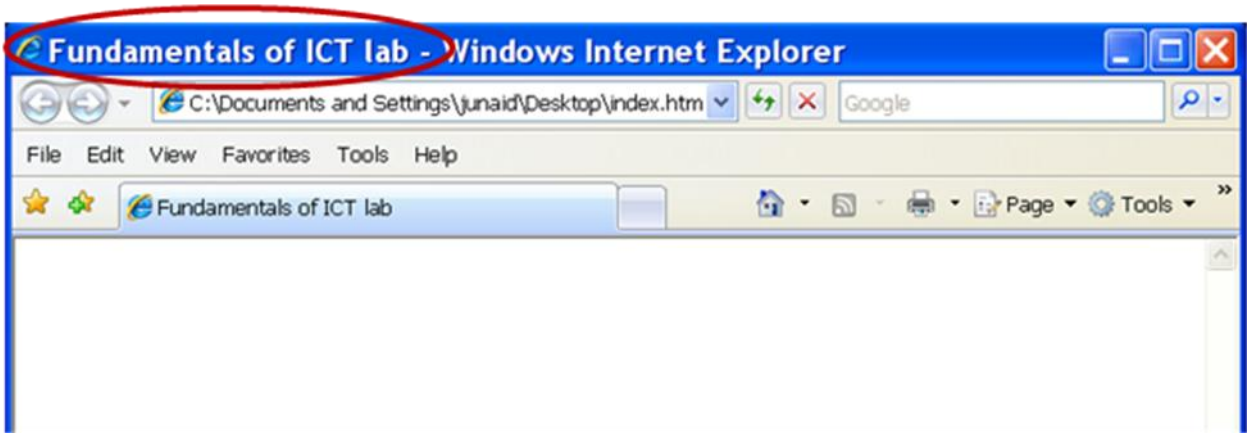
The **head** element (text between the *head* tags) contains general information, also called meta-information, about a document. The **title** (text between the *title* tags) is not part of the text of the document, but is a property of the whole document. It may not contain anchors, paragraph marks, or highlighting. The title may be used to identify the node in a history list, to label the window displaying the node, etc.

We will name the title of our webpage as Fundamentals of ICT lab. Note how the title text is enclosed in the **head** tags and the **title** tags.



Open the index.html file on the Desktop (modified as above) with Internet Explorer (this can be done by simply double clicking on the index.html file). You will find that the Internet Explorer displays "Fundamentals of ICT lab" as the title of the web page as configured in the figure above. The webpage is still empty as we have not added any text in the main page's body yet (we will do that soon).

For now, the index.html page should look like the following.

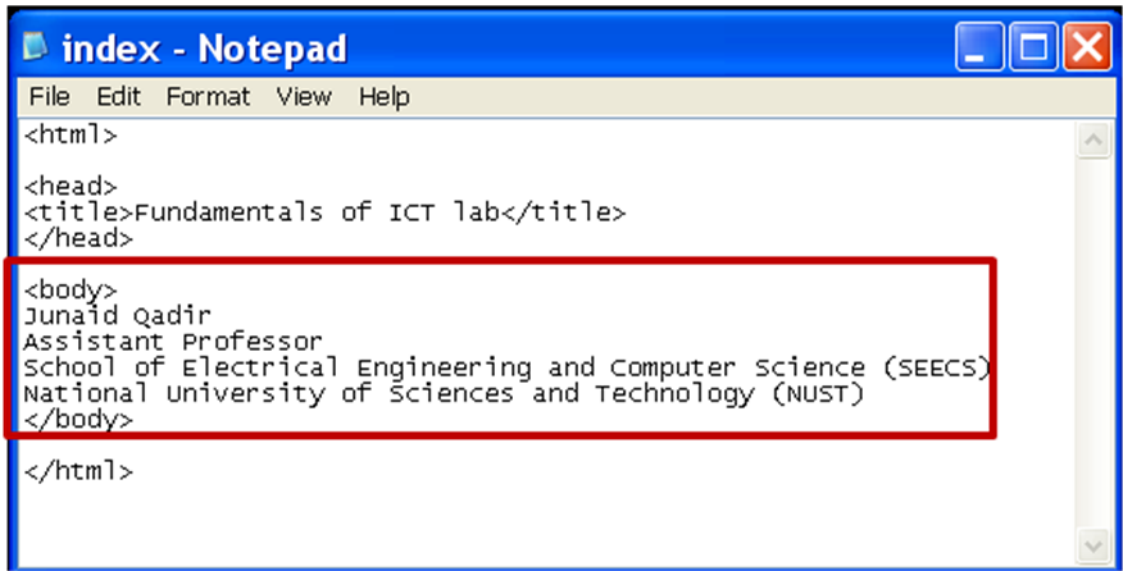


Copy the index.html into the ICTLab6 folder on Desktop and rename it as **Title.html**. [***Hint:*** a file can be renamed by selecting, right clicking, and choosing the rename option.]

3. Inserting text in a HTML page

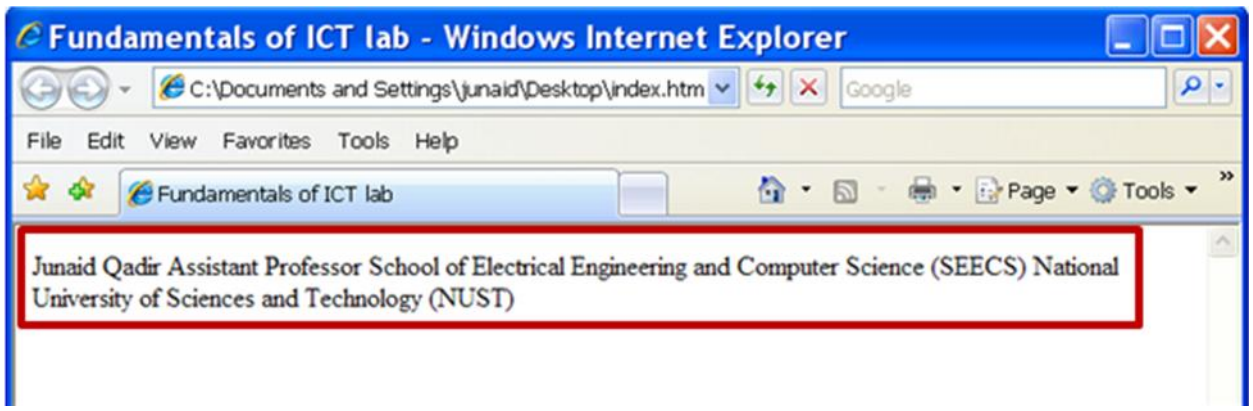
We will now add text to the main body of the HTML page. This is done using the **<body>** and **</body>** tags. The text contained between the *<body>* and *</body>* tags is displayed by a browser unlike data that is contained between the *<head></head>* tags.

Add the text contained in the following figure in the index.html file on the Desktop. Save the file when done. The index.html file on Desktop should like the following.



After saving the index.html file on Desktop, open the file in a browser (by double clicking on the file). If the index.html file is properly configured, the browser display should be as follows. You will notice that all there is no line break, and all the text we had put between the <body> tags is on one line.

A browser has to be told by HTML tags where line breaks, new paragraphs, etc. should occur. Since we have not used those tags yet, all the text is on one line as shown in the figure below.

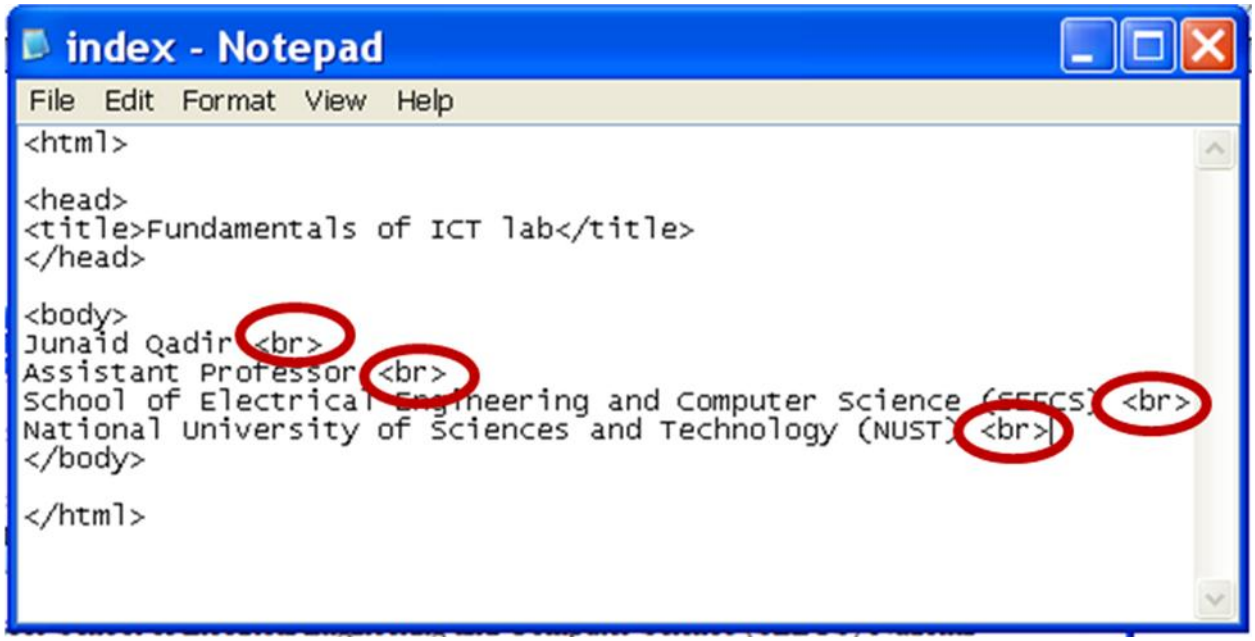


Save the index.html file on the Desktop in the ICTLab6 folder as **Allon1line.html**.

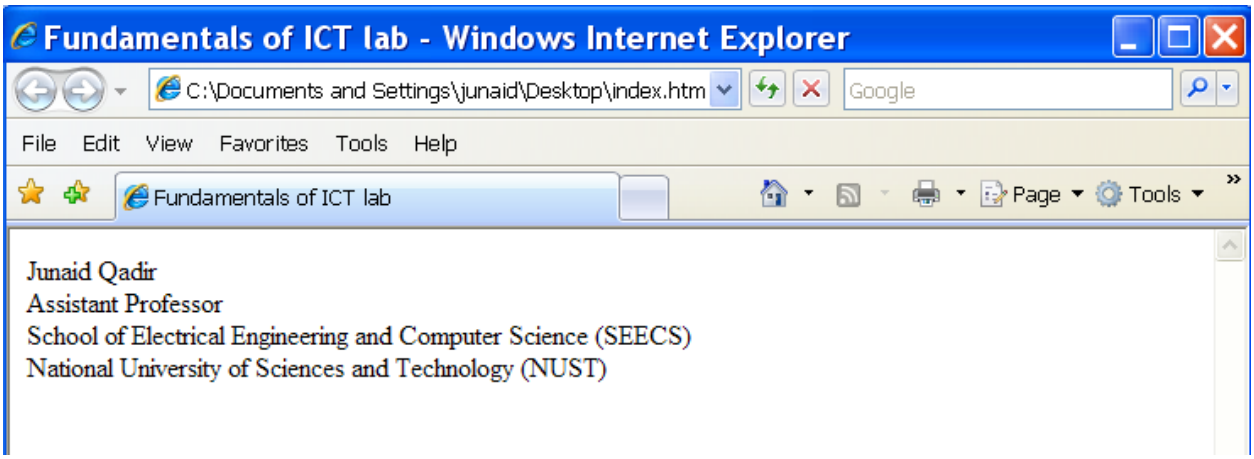
4. Inserting a line-break

We will now use the **
** HTML tag to introduce line breaks. This tag does not have a corresponding close tag and is used for introducing line breaks.

Open the index.html file on the Desktop and make the changes highlighted in the following figure.



After making the changes, save the index.html file and open it using a browser. The browser should display the page with proper line breaks as shown in the figure below. You will see that all the text is not on the same line any more.

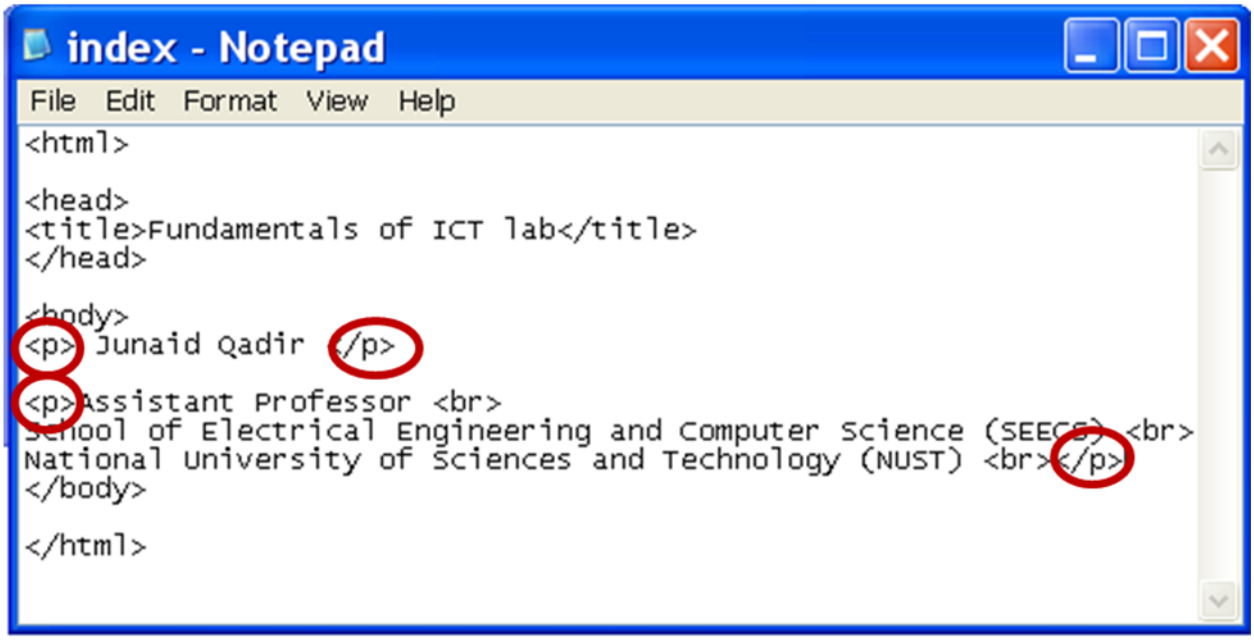


If you see the browser display as above, save the index.html file on the Desktop to the folder ICTLab6 as **Linebreak.html**.

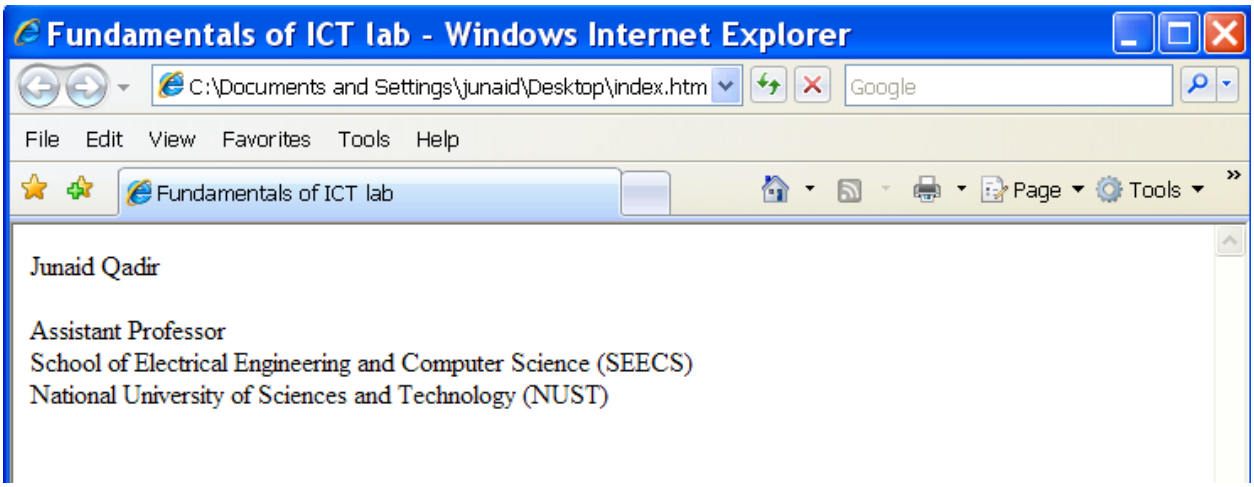
5. Insert a paragraph

In the figure above, all the text is in a single paragraph (although separated by line breaks). To start a new paragraph, the HTML tag ** ** should be used. The tag <*p*> is used to mark the start of a paragraph, and the end of paragraph is marked by ** **.

To observe how a paragraph is formatted in HTML, open the index.html file on the Desktop. Make the changes highlighted in the following figure and then save the file.



Open the index.html file on the Desktop (modified with the changes highlighted above) with a browser. You will see now that the web page is composed of two paragraphs. The second paragraph also has line breaks within the paragraph. The browser display should look like the following.

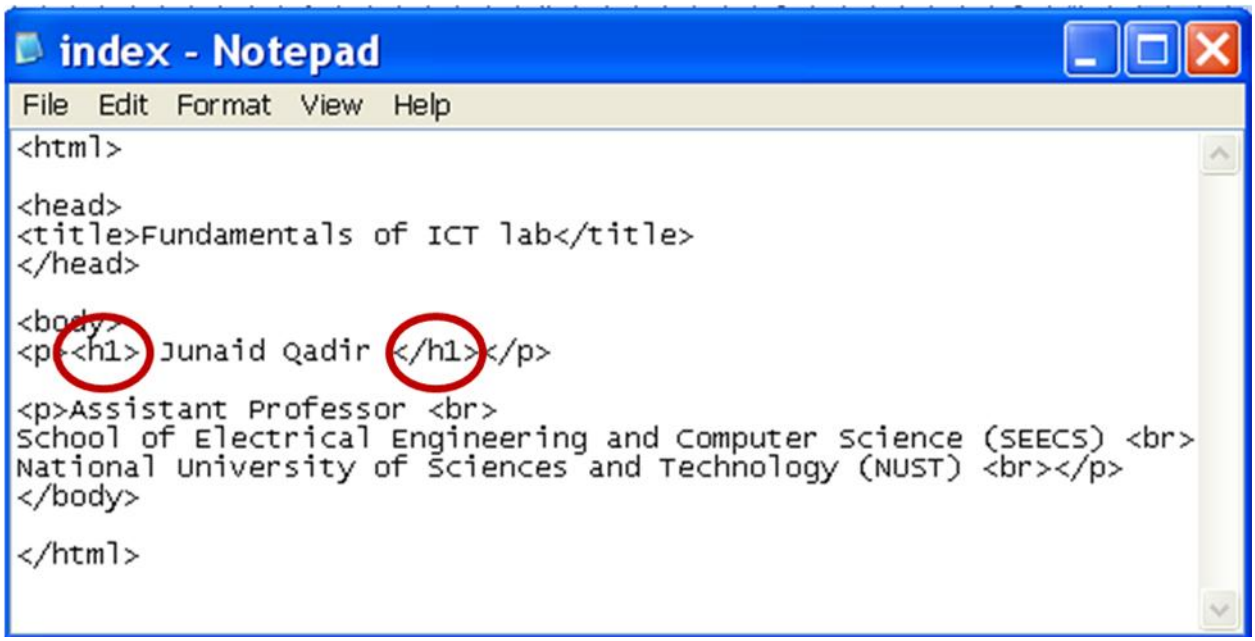


If you see the browser display as above, save the index.html page on the Desktop to the ICTlab6 folder and name it as **Paragraph.html**.

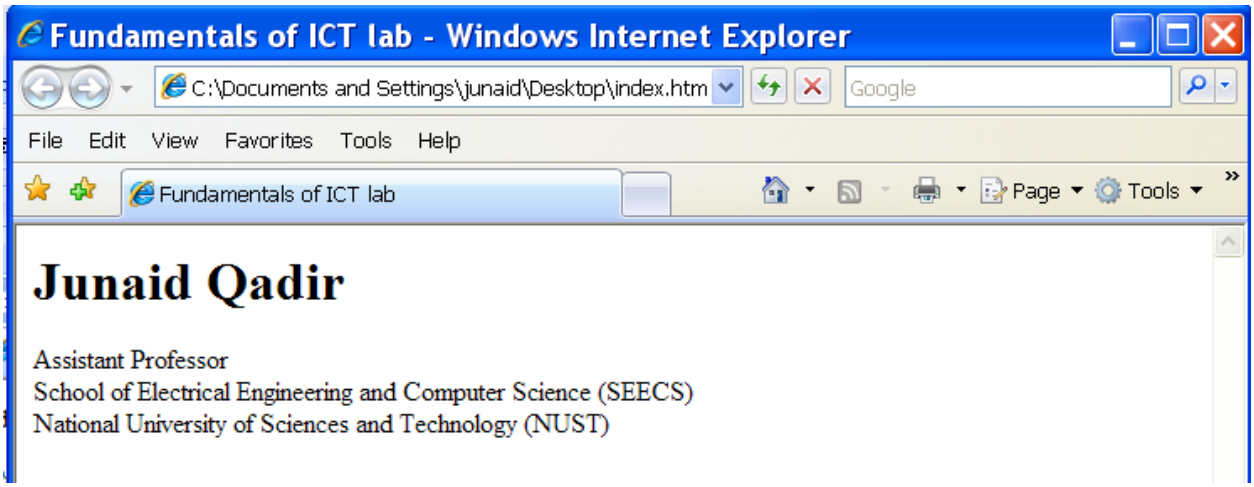
6. Inserting headings

In the index.html file on the Desktop, currently no text is highlighted as a heading. If certain text is to be a heading, the HTML tags **<h1></h1>**, **<h2></h2>**, **<h3></h3>**, **<h4></h4>** (and so on) may be used. The tag <*h1*> is used for bigger sized heading with the heading size decreasing for <*h2*> <*h3*> and so on.

Open the index.html file on the Desktop and make the changes highlighted in the following figure. The first paragraph is highlighted using the <*h1*> tags.

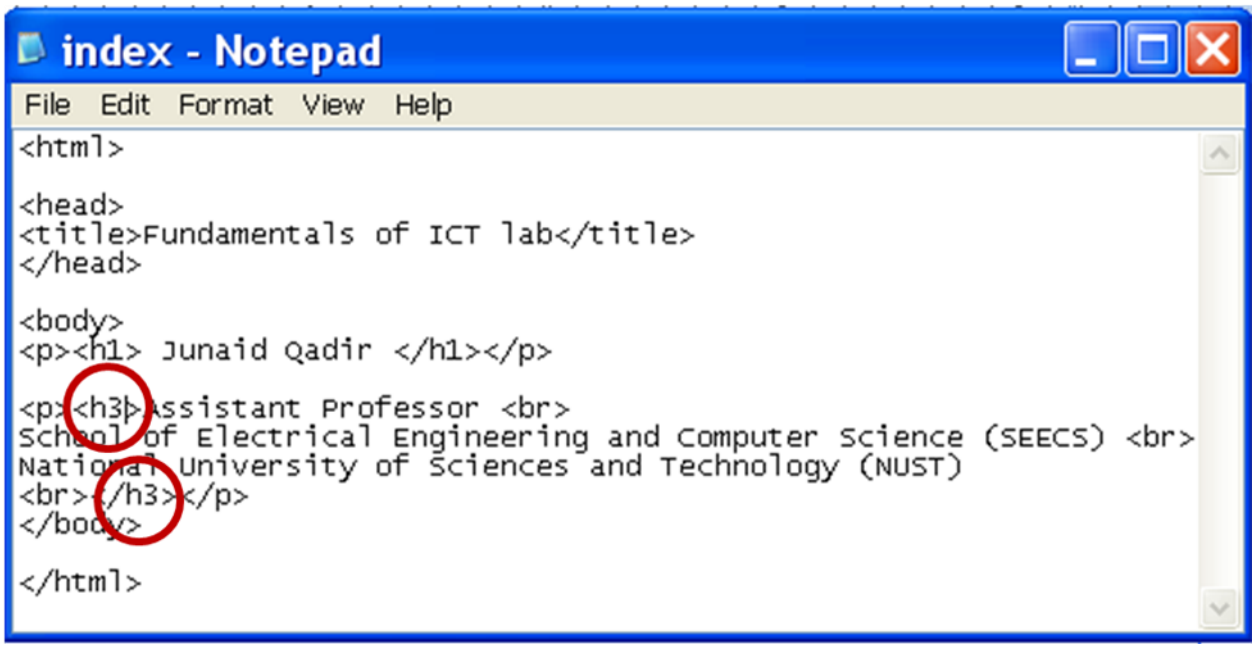


After making the changes in the index.html file on the Desktop as shown above, save the file. Open the file now with a browser. The browser display should look like the following. You will note that the first paragraph now highlighted as a heading with a large sized font.

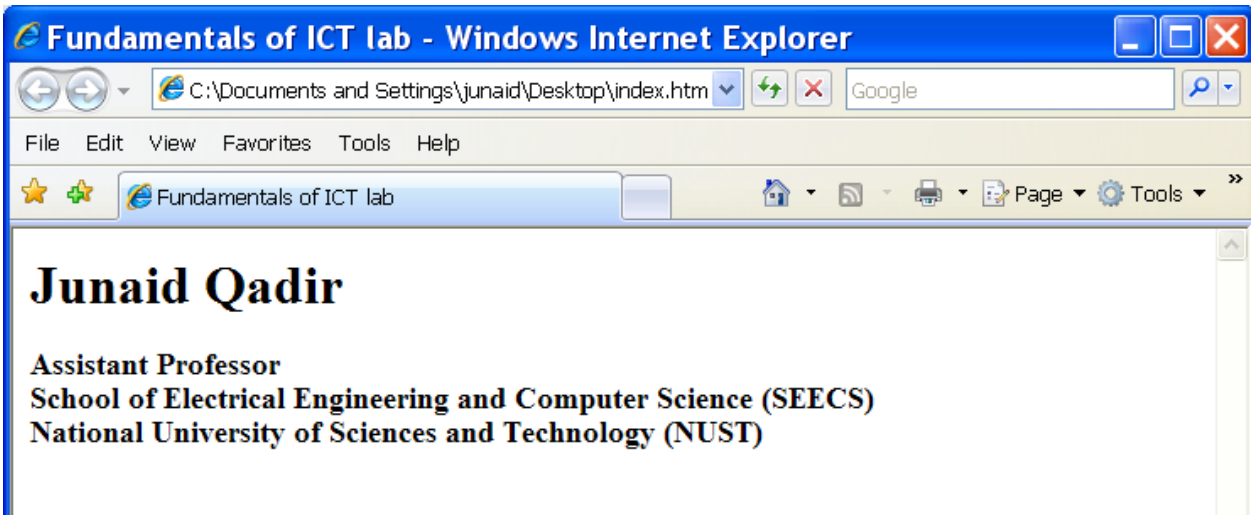


If the browser display is as shown above, save the index.html file (on the Desktop) in the ICTLab6 folder as **Headings1.html**.

Open the index.html file on the Desktop, and make the changes highlighted in the figure below.



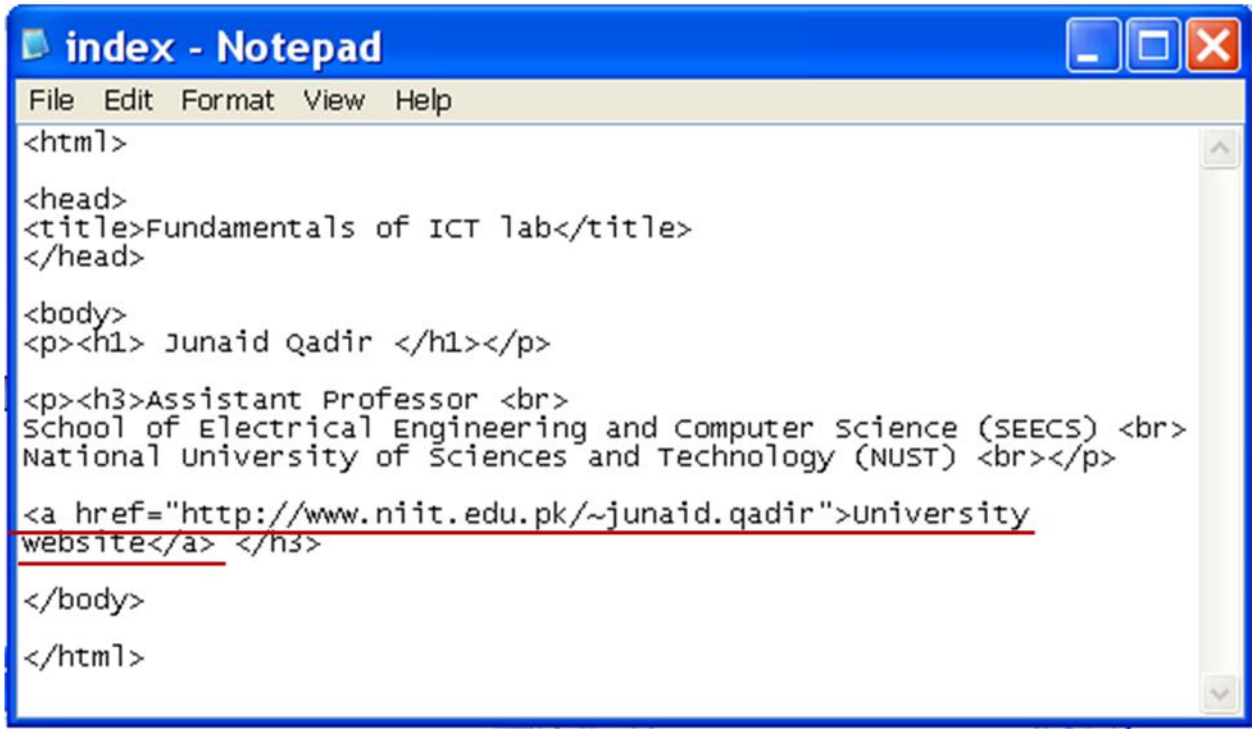
Save the index.html file after making the changes highlighted above, and open the file using a browser. You should now be able to see the second paragraph text highlighted as well as a smaller sized heading.



If the browser display is similar to as shown above, save the index.html file (on the Desktop) to the folder ICTLab6 as **Headings2.html**.

7. Inserting a hyperlink

We will now add a hyperlink on the index.html file. Open the index.html file on the Desktop and make the change highlighted in the following figure.



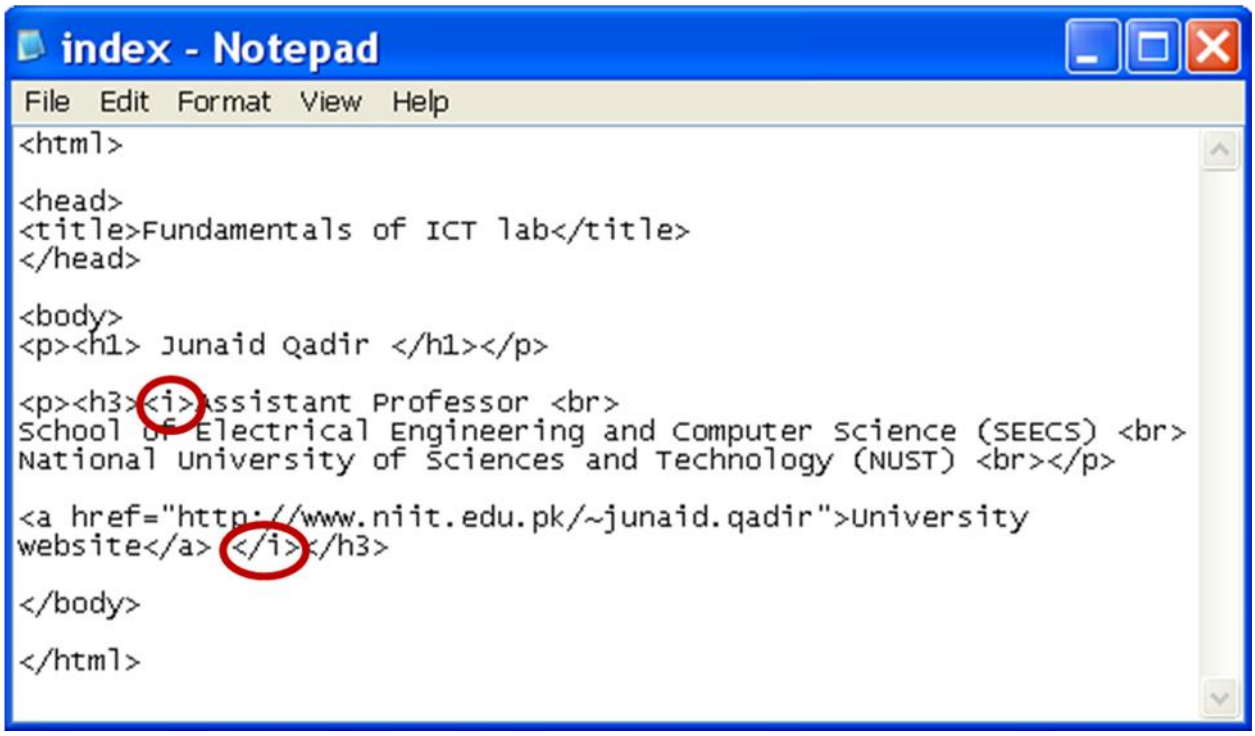
Save the file after making the change highlighted above. Open the file using a browser. You should see that the browser display is similar to as shown below. You will observe that a hyperlink to the specified URL is now added. Click on the URL and note where the target of the hyperlink.

If the browser display is similar to as shown above, save the index.html file on the Desktop to the folder ICTLab6 as **Hyperlink.html**.

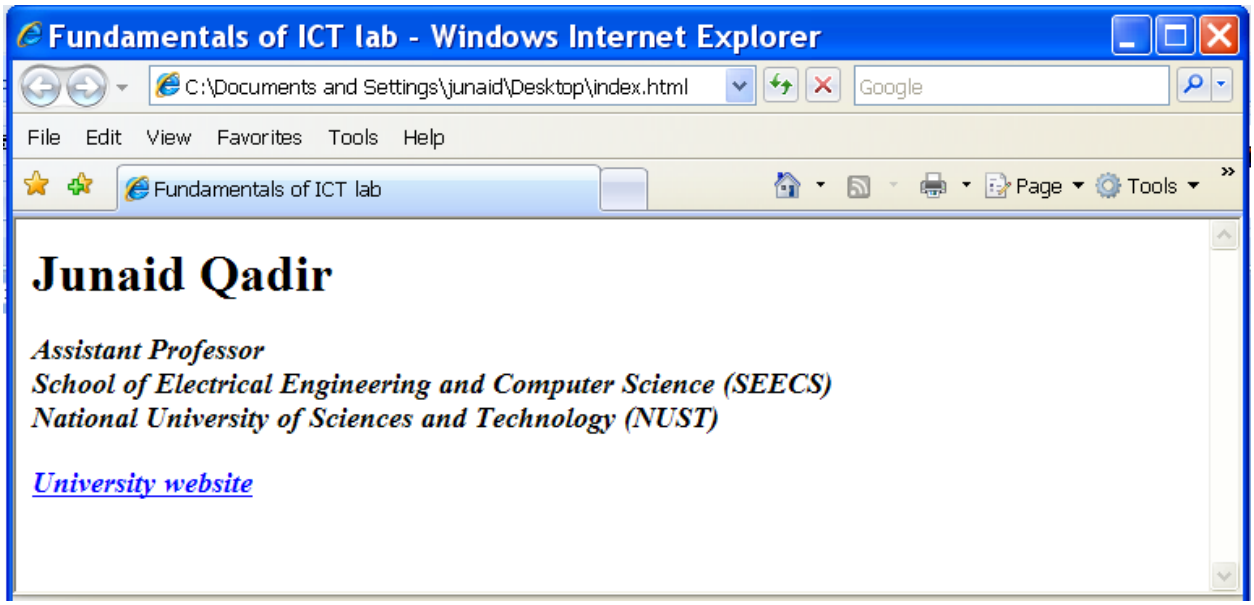
8. Making text italic, underlined, or bold

We will now see how a text can be *italicized*, underlined, or made **bold** in HTML. The tags for these tasks are, respectively, ****, **<u></u>** and ****.

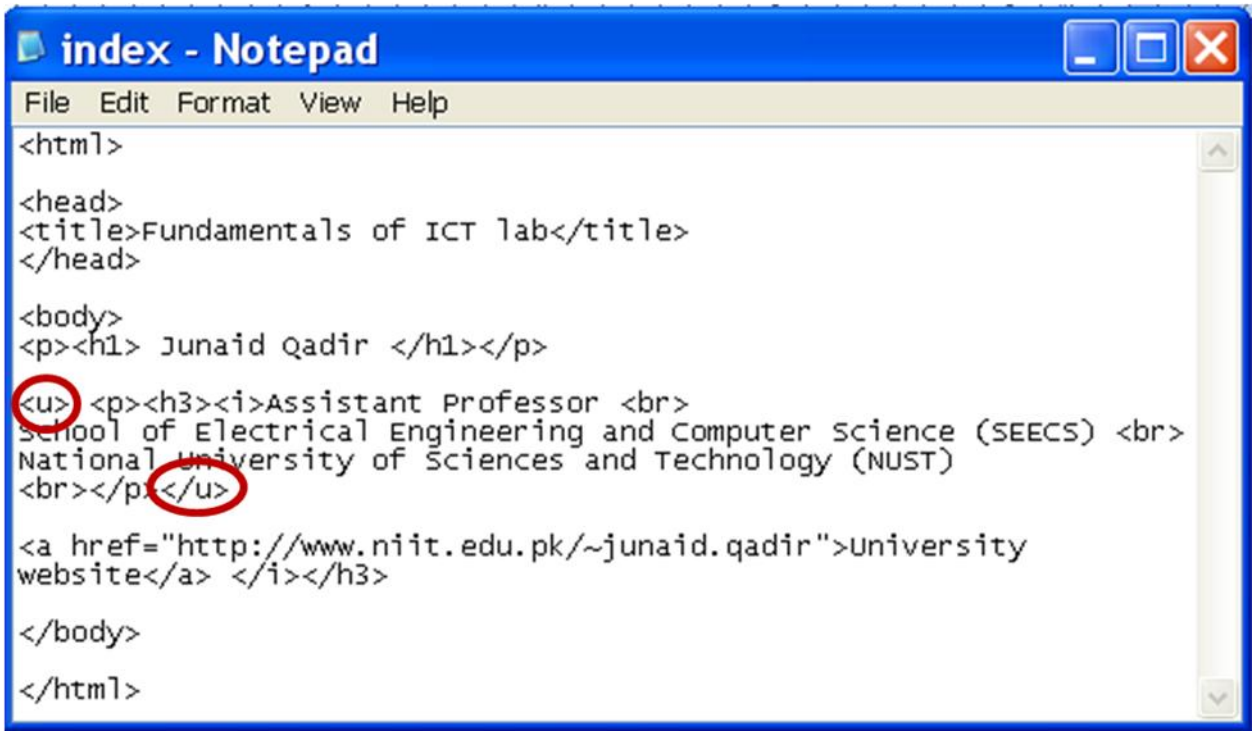
Italicizing: To italicize a body of text, it must be enclosed in   tags. To demonstrate this, open the index.html file on the Desktop and make the changes highlighted in the following figure.



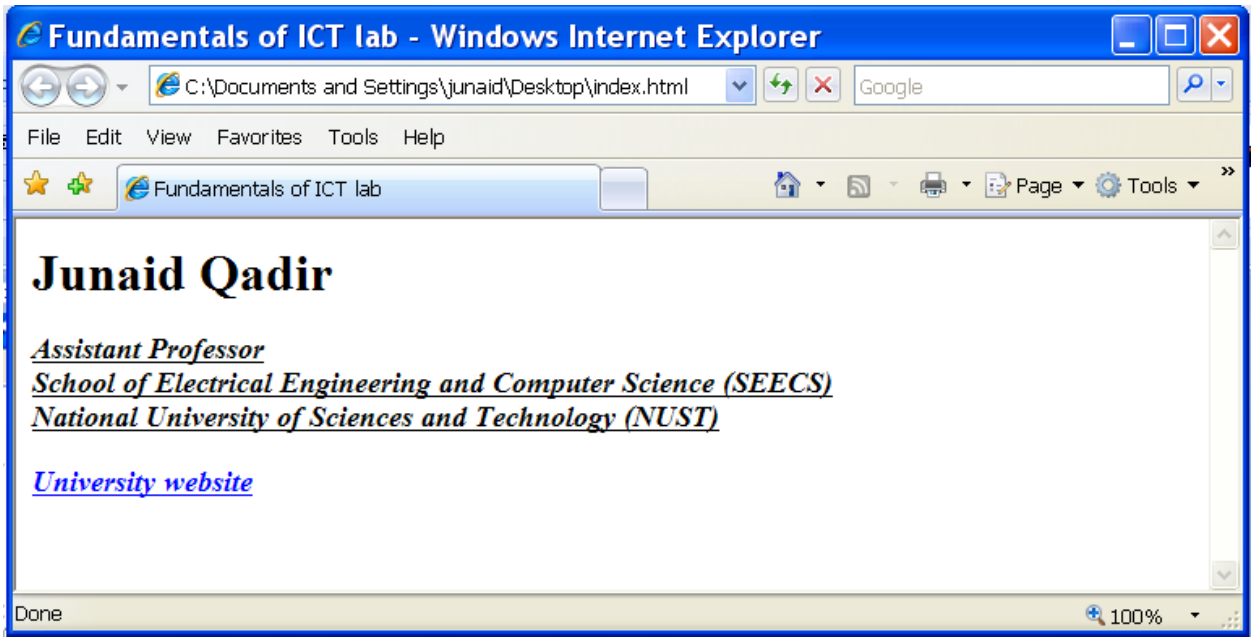
After making the changes shown above, save the file and then open in using a browser. You will observe that the text between the ** and ** tags is italicized. If the browser display is as shown below, save the index.html file in the ICTlab6 folder as **Italics.html**.



Underlining: Similar to the approach shown above, use the *<u>* and *</u>* tags for underlining some text. Open the index.html file on the Desktop and make the changes as highlighted in the figure below.



After making the changes above, save the file and then open it using a browser. You should see the browser display similar to as shown below.

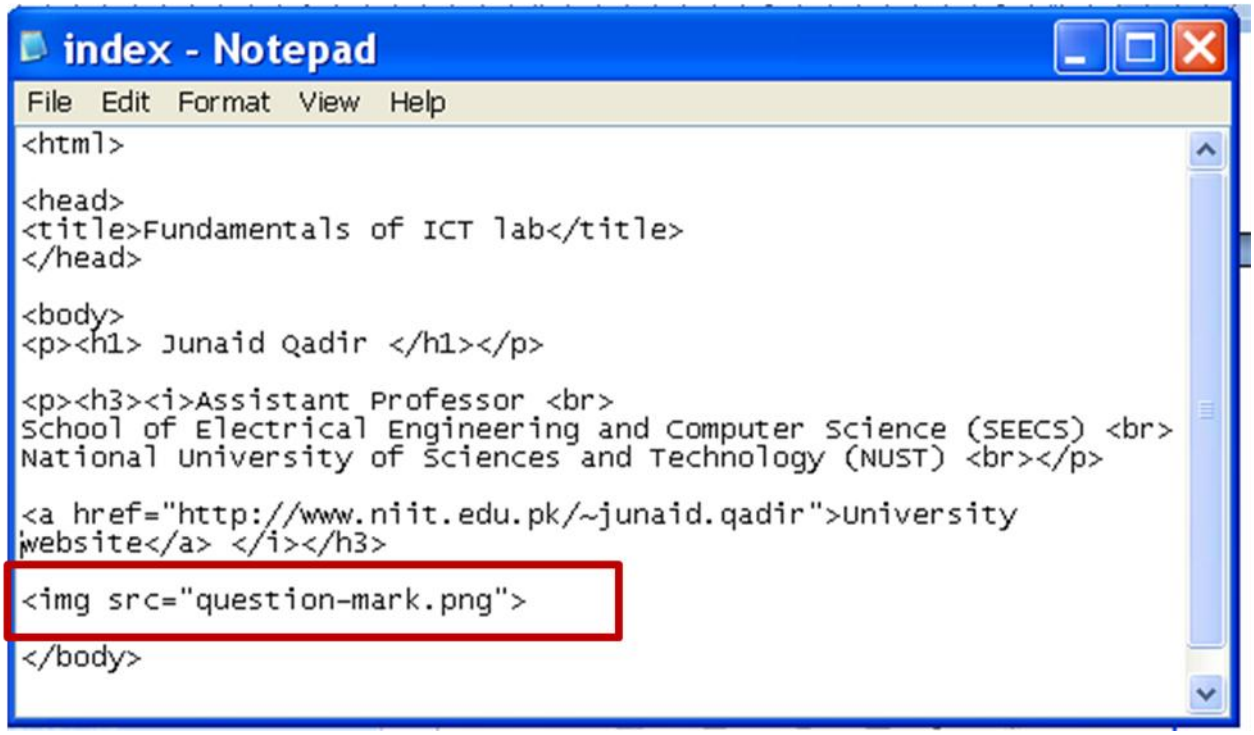


If the browser display is similar to as shown above, save the index.html file (on the Desktop) to the ICTLab6 folder as **Underline.html**.

9. Inserting an image

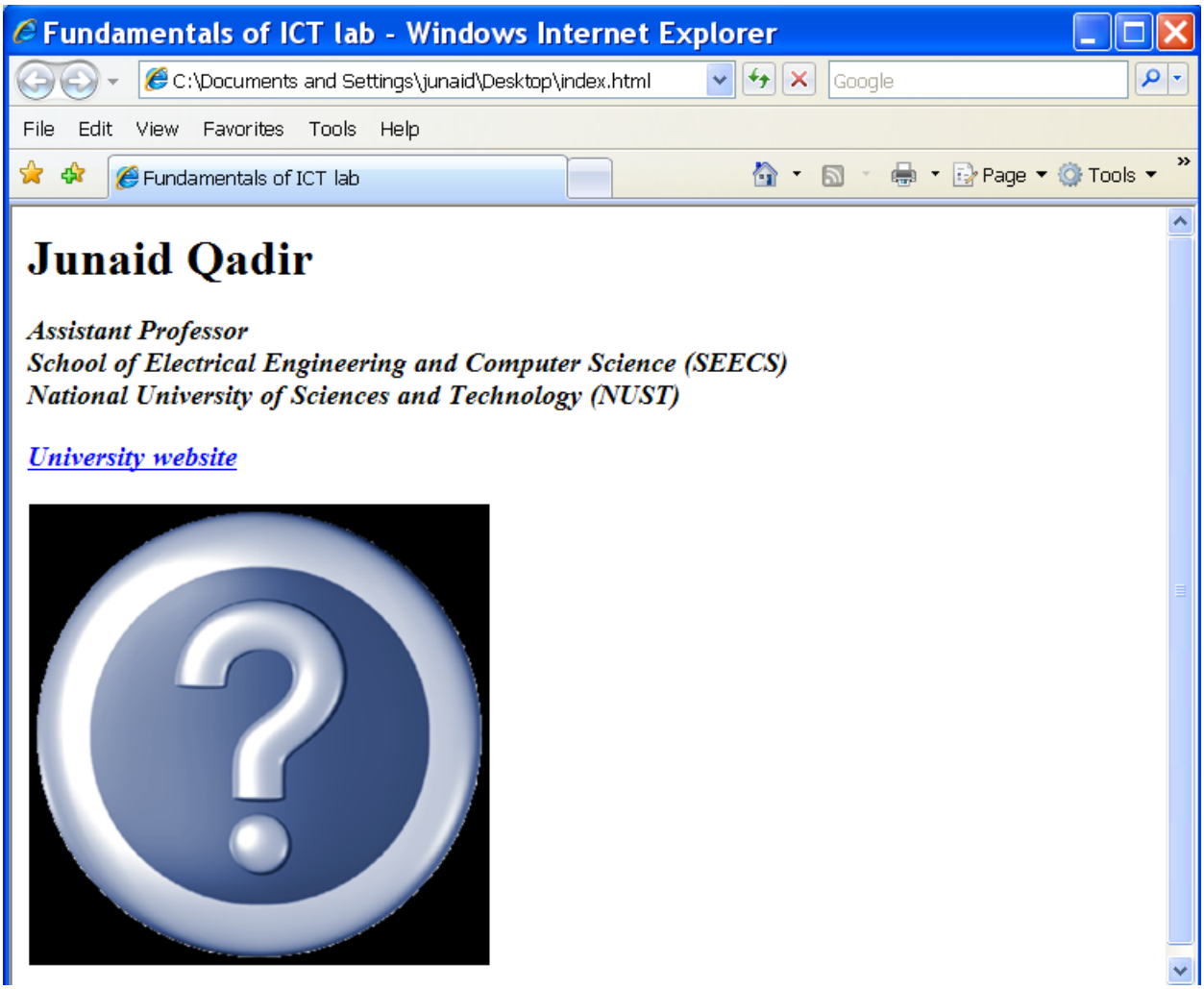
To insert an image in an HTML page, **<img>** tag is used.

We will now demonstrate how an image is added to an HTML page. Modify the index.html page on the Desktop to as shown below and save the file.



The HTML page will include the file question-mark.png as an image. For this to work, question-mark.png should be in the same folder as the index.html file. Download the question-mark.png file from the course website and also place on the Desktop.

With the image file also on the Desktop, open the index.html file using a browser. You should be able to see the image as part of the web page displayed in the browser (as shown below).



If you find the browser display as shown above, copy the index.html page to ICTLab6 folder and rename it as **Image.html.** You **must** also copy the question-mark.png file to the ICTLab6 folder.

10. Employ style sheets

A Cascading Style Sheet (CSS) can be used to define the styling format of an HTML page. To use CSS in our HTML page, first of all, we will have to define our style in a file; secondly, that file must be linked to the HTML page. *[This should become clear soon.]*

Open the **format.css** file on the Desktop and modify it to as shown in the figure below.

format - Notepad

File Edit Format View Help

```
h1 {
font-family: verdana,helvetica,sans serif;
font-weight: bold;
}
```

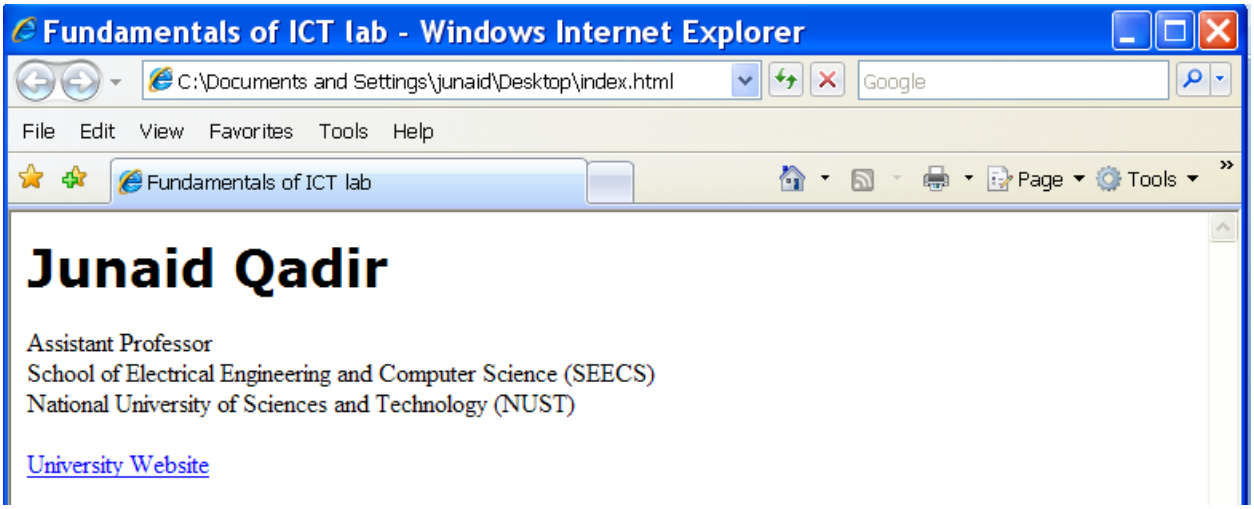
The style sheet above is defining how text marked with <h1> tag should be displayed in a browser. After creating a style sheet, it has to be included in the web page too. This is done using the **<link>** tag as shown in the figure below.

index - Notepad

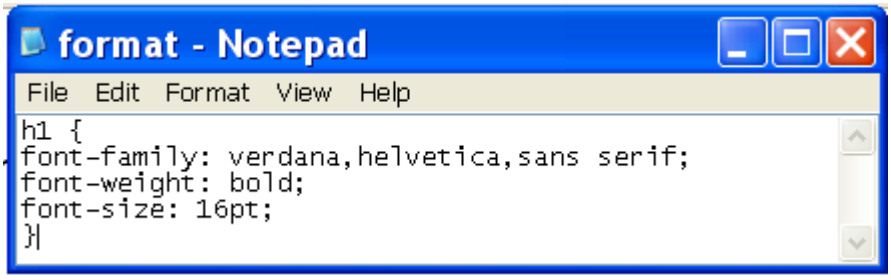
File Edit Format View Help

```
<html>

<head>
<link rel="stylesheet" href="format.css">
<title>Fundamentals of ICT lab</title>
</head>

<body>

<p><h1> Junaid Qadir </h1></p>

<p>Assistant Professor <br>
School of Electrical Engineering and Computer Science (SEECS) <br>
National University of Sciences and Technology (NUST) <br></p>

<a href="http://www.niit.edu.pk/~junaid.qadir">University Website</a>

</body>

</html>
```

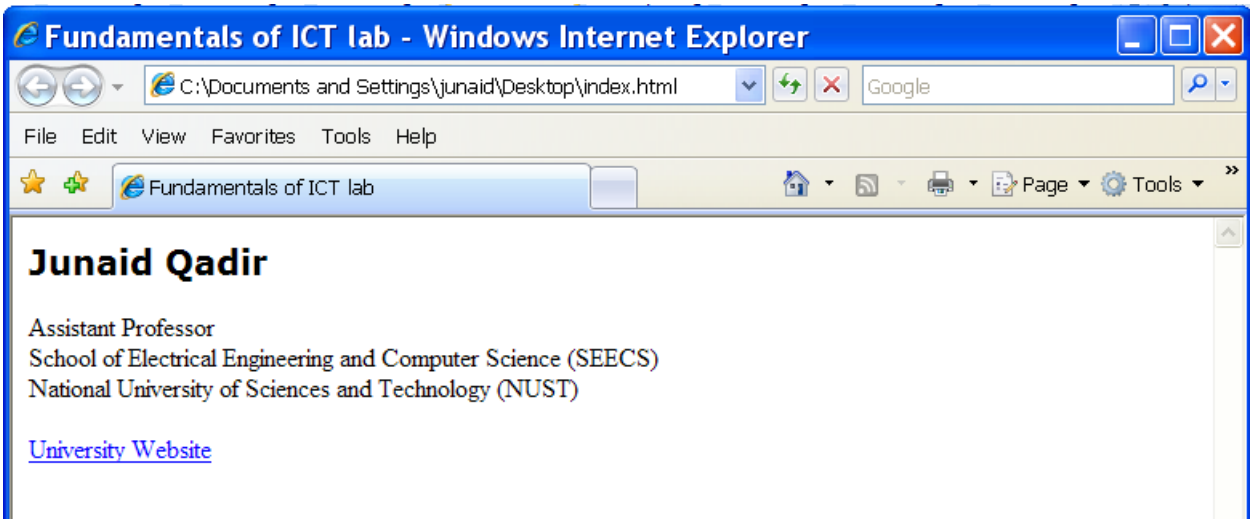
The display of the web page in a browser will now follow the style dictated by the format.css file.



The format defined by the CSS file can be further modified to affect all parts of the HTML page. As an example, we can also modify the default font size associated with heading types. This is shown below for the heading type **h1**.



Using the format.css file above, the browser display would look the following



If the browser display is as shown above, save the index.html and the format.css files (on the Desktop) in the ICTLab6 folder as **CSS.html** and the **Format.css**, respectively.

Part II---Basic Javascript (Dynamic web pages).

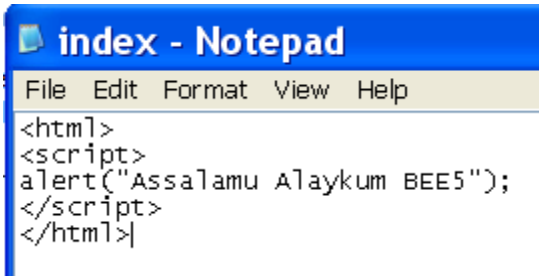
Static web pages (that use simple HTML) do not allow users to interact with the website. Dynamic web pages allow a user to interact with the website, and customize content according to the user.

The **<script> </script>** tag set specifies to the browser that any code between the tags is not written in HTML, but in a scripting language. You can place script tags anywhere in an HTML document. All of the code between <*script*> and <*/script*> is written according to the rules of JavaScript. JavaScript is a different language than HTML, and its rules are different.

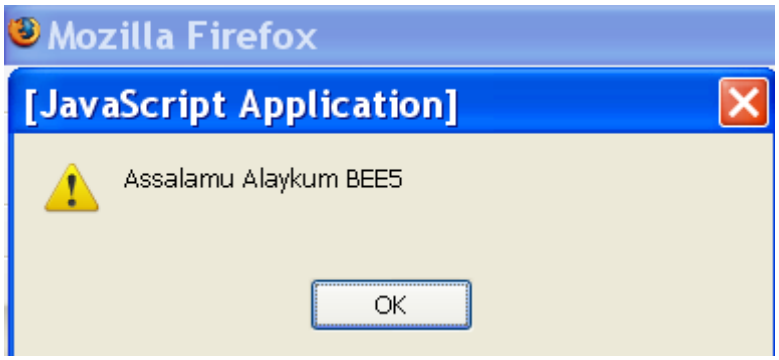
1. Output a message prompt

Create a new file on the Desktop and name it as Javascript.html.

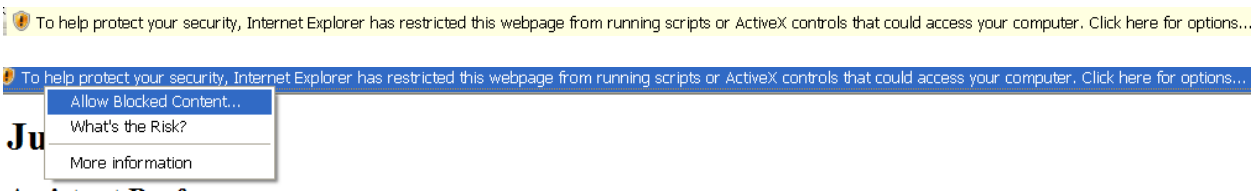
Modify the Javascript.html file to as shown in the figure below.



Save the file and open it using a browser. You should be able to find a message prompt similar to the prompt shown in the figure below.



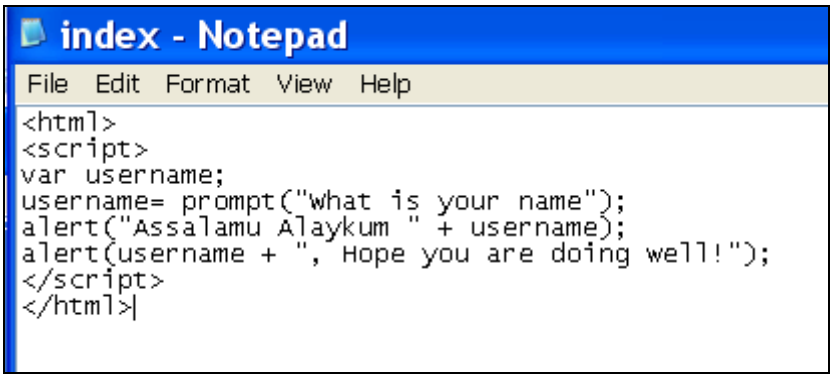
[**Note: Internet Explorer users---**If you see a warning prompt such as the following, allow the Blocked Content]



If you are receiving a prompt as shown above when the Javascript.html file is opened, save the **Javascript1.html** file in the ICTLab6 folder.

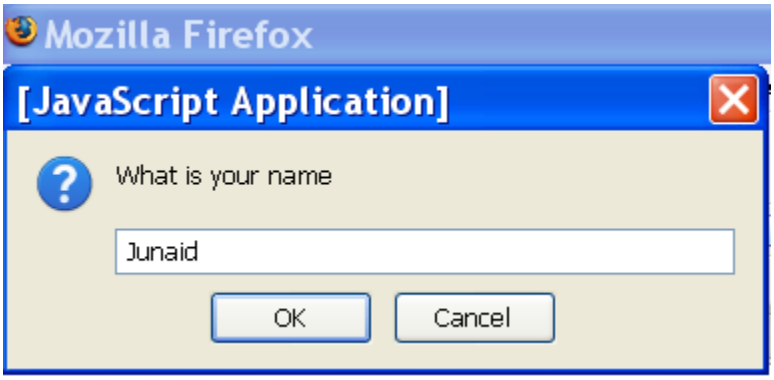
2. Prompt for user's name and then output a customized greeting.

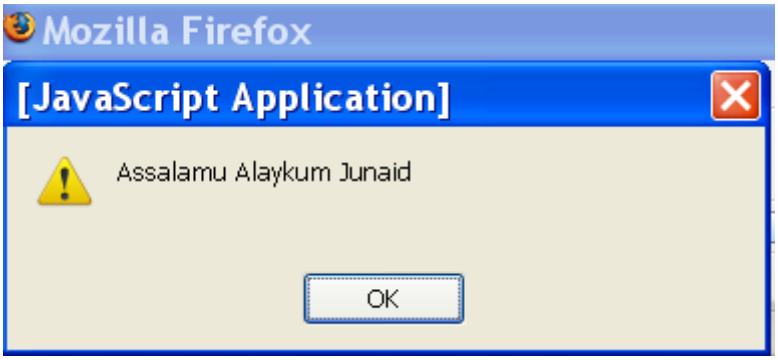
Modify the Javascript.html file on the Desktop to as shown in the figure below.

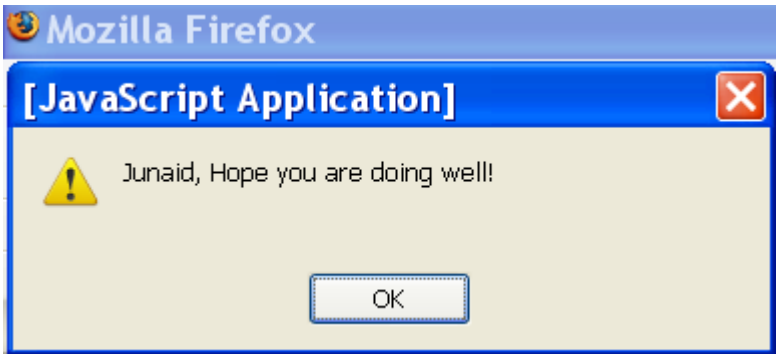


Save the file, and open it using a browser.

You will find that the browser will output a series of alerts (also called prompts) as shown below.







If you are receiving a prompt as shown above when the Javascript.html file is opened, save the **Javascript2.html** file in the ICTLab6 folder.

Next week's lab: Web-applications and Computer Security