Web Applications

Name: __________________________ (   ) Class: _________ Date: _________

Lesson 2: HyperText Markup Language

Instructional Objectives:

By the end of this task, you should be able to:

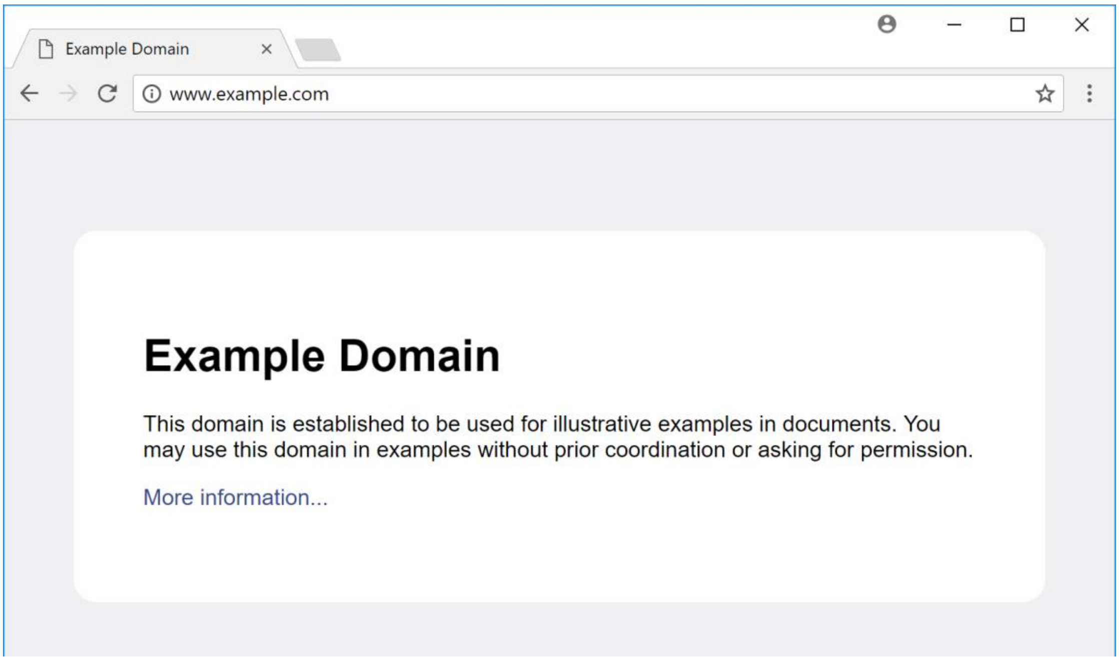
- State that web pages are written using the HyperText Markup Language (HTML) and Cascading Style Sheets (CSS) language
- Use the "View Source" feature of a web browser to examine the HTML source code of any web page
- Identify the tags, attributes, character references and comments that are present in a given static HTML document
- Explain how HTML only describes the structure of a web page (in terms of elements) and not its presentation
- Distinguish between tags for normal elements that require an end tag and tags for void elements that do not have an end tag
- Use the `<html>`, `<head>`, `<title>`, `<body>`, `<h1>`, `<h2>`, `<h3>`, `<p>`, `<a>`, `<img>`, `<table>`, `<tr>`, `<th>`, `<td>`, `<form>`, `<input>` and `<textarea>` tags with relevant attributes to create web pages that meet a given set of requirements without producing any syntax errors
- Use the Developer Tools of a web browser to examine how form data is encoded differently for HTTP GET requests and HTTP POST requests

---

Part 1: Behind the Scenes of a Web Page

Open a web browser such as Google Chrome and visit `www.example.com`. You should see a simple web page as shown in the following screenshot.

Web Applications



Example Domain

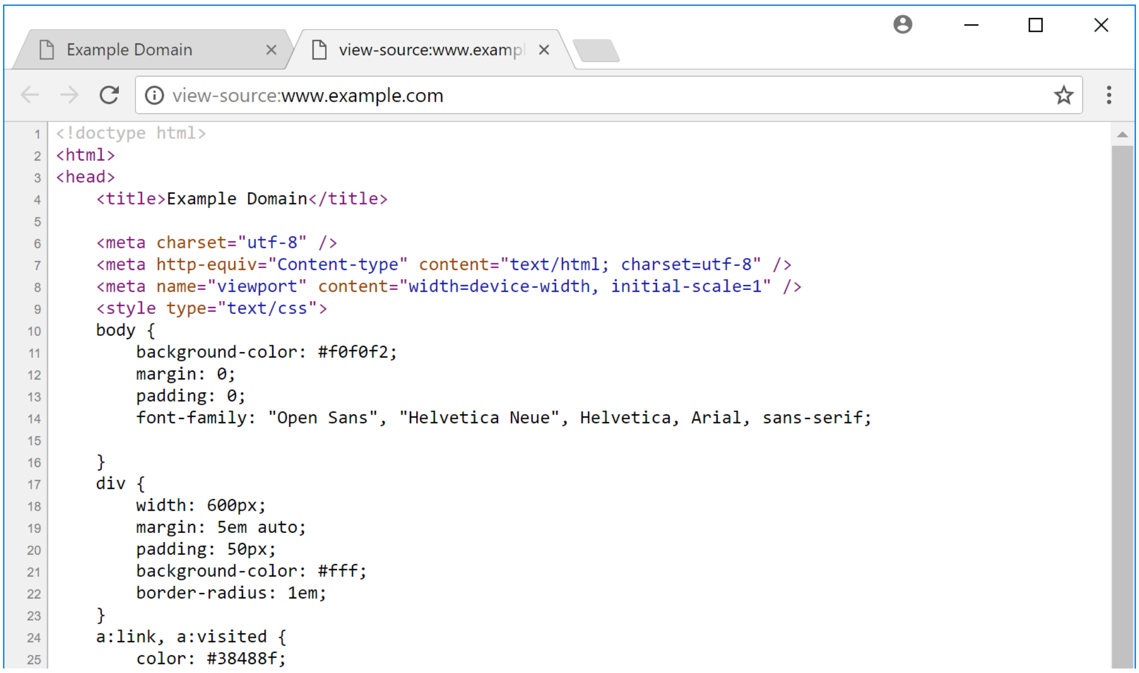
This domain is established to be used for illustrative examples in documents. You may use this domain in examples without prior coordination or asking for permission.

More information...

How do you think web pages like this one are made?

Just like how programs are written in a programming language such as Python, web pages are written using the **HyperText Markup Language (HTML)**. Unlike programming languages that are specialized for describing step-by-step instructions, HTML is specialized for describing the structure of web pages.

To view the HTML source code of a web page in Chrome, press Ctrl-U. (Alternatively, right-click and select "View page source".) Try this now for the web page on `www.example.com`:

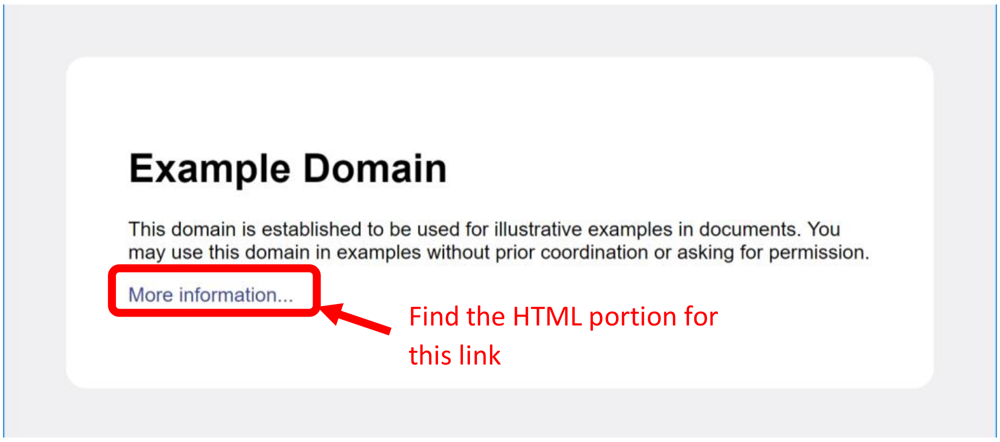


```
<!doctype html>
<html>
<head>
    <title>Example Domain</title>

    <meta charset="utf-8" />
    <meta http-equiv="Content-type" content="text/html; charset=utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <style type="text/css">
    body {
        background-color: #f0f0f2;
        margin: 0;
        padding: 0;
        font-family: "Open Sans", "Helvetica Neue", Helvetica, Arial, sans-serif;

    }
    div {
        width: 600px;
        margin: 5em auto;
        padding: 50px;
        background-color: #fff;
        border-radius: 1em;
    }
    a:link, a:visited {
        color: #38488f;
```

1 Try to find the portion of the HTML that represents the "More information..." link shown below and copy it into the blank space provided:



Example Domain

This domain is established to be used for illustrative examples in documents. You may use this domain in examples without prior coordination or asking for permission.

More information...

Find the HTML portion for this link

....

....

....

Examine the HTML source code on the web browser. Do you see that the contents of the web page are surrounded by text enclosed in angle brackets (i.e., < and >)?

```
<h1>Example Domain</h1>
```

The text surrounded by angle brackets are special processing instructions for the web browser called **tags**. These tags are underlined for clarity below:

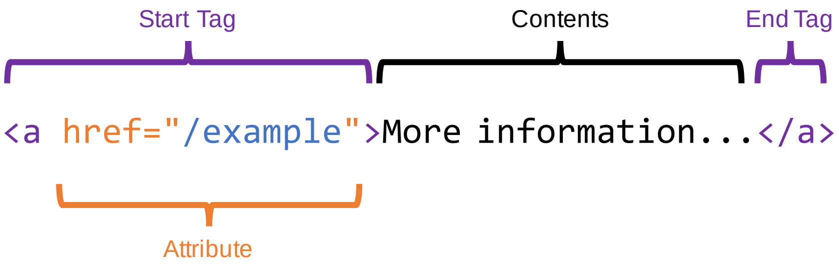
```
<h1>Example Domain</h1>
```

You may also notice that the first part of the HTML document includes some code that appears to be written in another computer language. This is a **style sheet** written in the **Cascading Style Sheets (CSS)** language that controls the web page's appearance. Without style sheets, web pages have a very limited appearance by default (i.e., black *Times New Roman* text on a white background).

In this practical task, we will focus on content first before looking at layout with CSS in the following practical tasks.

Part 2: Anatomy of a HTML Document

You have learnt to identify tags in a HTML document as portions of text that are enclosed in angle brackets. Tags can be further classified into **start tags** and **end tags**. Start tags may also have one or more **attributes**:



2 How do you tell the difference between start tags and end tags?

....

....

Like Python, certain characters (such as the angle brackets `<` and `>`) have special meanings in HTML and would cause a syntax error if they are used without escaping them. In HTML, escape codes are called **character references** that start with an ampersand & and end with a semicolon `;`.

Some common character references are as follows:

Character	Ampersand	Less-Than Sign	Greater-Than Sign	Quotation Mark
	&	<	>	"
Character Reference	`&amp;`	`&lt;`	`&gt;`	`&quot;`

3 Circle all start tags and underline all end tags in the following HTML document. (Ignore the first line that reads `<!doctype html>` for now. It will be explained in a later part of this practical task.)

```
<!doctype html>
<html>
<head><title>Welcome Page</title></head>
<body>
    <p>Welcome to our R&amp;D department.</p>
    <img src="welcome.png" alt="Welcome">
</body>
</html>
```

4 In the previous document, which start tags do not have matching end tags?

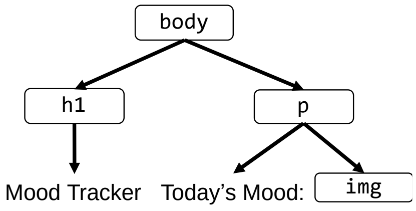
5 Circle the syntax error in the following HTML snippet.

```
<p>I was sent to A&E in an ambulance.</p>
```

Rewrite the snippet and fix the error so it works as intended on all browsers:

HTML tags are used to describe the structure of a web page by organizing its contents into a tree of **elements**. In general, each start tag corresponds to a single element. For instance, the HTML snippet on the left also describes the tree of elements on the right:

```
<body>
    <h1>Mood Tracker</h1>
    <p>
        Today's Mood:
        <img src="happy.jpg" alt="happy">
    </p>
</body>
```



You may have noticed that some start tags always have a matching end tag while other start tags do not. Start tags with a matching end tag (such as `<body>`, `<h1>` and `<p>`) correspond to **normal elements** that may contain a combination of text contents and other elements. On the other hand, start tags that do not have a matching end tag (such as `<img>` and `<input>`) correspond to **void elements** that must not contain any other content.

Finally, like Python programs, HTML documents may include comments that are readable by humans but are ignored by the web browser. Comments in HTML must start with `<!--` and end with `-->`. Furthermore, the contents of the comment must NOT contain two consecutive hyphen characters `--`.

6 For each code snippet below, place a tick in the corresponding box if the snippet is a HTML comment with no syntax errors.

Code Snippet	Valid?
`<!-- The markup language of the web -- HTML. -->`	
`<!-- The markup language of the web - HTML. -->`	
`<!- The markup language of the web - HTML. ->`	

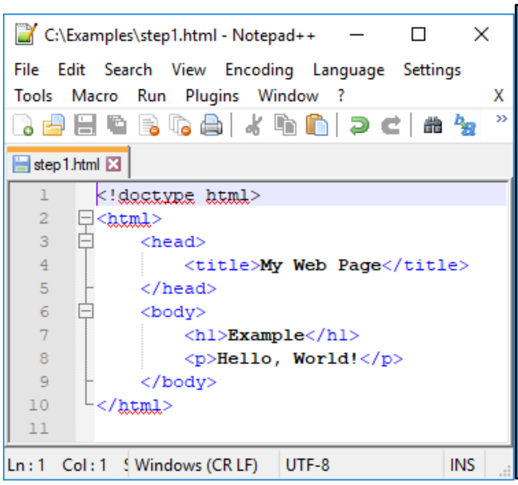
Part 3: Your Own Web Page

Now that we know about the basics of HTML, it is time to write our own web page. Like Python programs, HTML documents are just plain text files except that they use a `.html` extension instead of a `.py` extension. While we can write HTML using any text editor, we will use Notepad++ so that our code will be displayed with syntax highlighting.

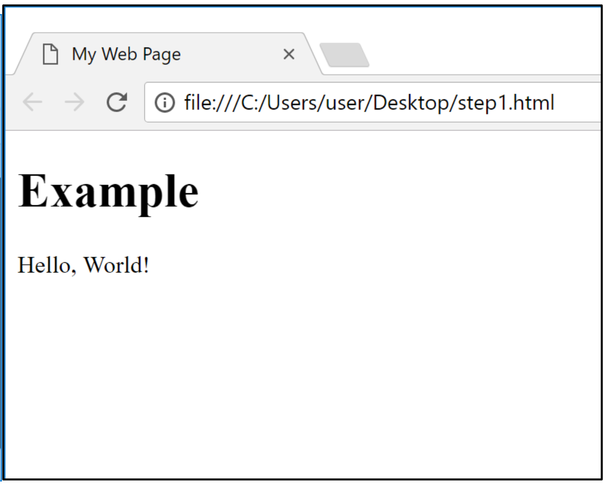
Open Notepad++, create a new file and save it in your user directory as `example.html`. Then enter the following HTML document:

```
<!doctype html>
<html>
    <head>
        <title>My Web Page</title>
    </head>
    <body>
        <h1>Example</h1>
        <p>Hello, World!</p>
    </body>
</html>
```

Save the document again and double-click `example.html` in Windows Explorer to open it in a web browser. You should see something similar to the following:


C:\Examples\step1.html - Notepad++
File Edit Search View Encoding Language Settings
Tools Macro Run Plugins Window ?
step1.html
Ln : 1 Col : 1 Windows (CR LF) UTF-8 INS

```
<!doctype html>
<html>
    <head>
        <title>My Web Page</title>
    </head>
    <body>
        <h1>Example</h1>
        <p>Hello, World!</p>
    </body>
</html>
```



Example

Hello, World!

Congratulations! You have just created a web page.

Just like how Python has many functions, HTML has a vocabulary of tags such as `<head>`, `<title>`, `<body>`, `<h1>` and `<p>` that you should become familiar with:

The remainder of this practical task will introduce these HTML tags by organising them into five groups: required tags, tags for describing sections, tags for describing text and media, tags for describing tables and tags for describing forms.

Part 4: Required Tags

All HTML documents have a number of required elements. First, every HTML document must start with a **doctype** that looks like this:

```
<!doctype html>
```

This declares that the remainder of the file is a HTML document. (Note that the doctype is technically not considered a tag. It tells the web browser the version of HTML used in the document. This is important as the web browser supports multiple versions of HTML.)

After the doctype, the remainder of the document must be enclosed between a `<html>` start tag and `</html>` end tag:

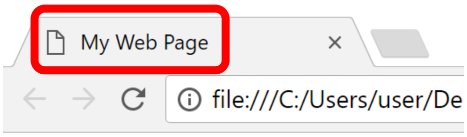
```
<!doctype html>
<html>
     ...document goes here...
</html>
```

This describes a single `html` element that serves at the **root element** of the document's structure. Inside the `html` element, there must be a `head` element for metadata (data that describes other data), followed immediately by a `body` element for the main content:

```
<!doctype html>
<html>
    <head>
        ...metadata goes here...
    </head>
    <body>
        ...main content goes here...
    </body>
</html>
```

Inside the `head` element, there must be a `title` element to set the title of the web page that appears in the browser window or tab:

```
<!doctype html>
<html>
    <head>
        <title>My Web Page</title>
    </head>
    <body>
        ...main content goes here...
    </body>
</html>
```



These required tags serve to ensure that all HTML documents are contained inside a single `html` element and that metadata is placed in a `head` element, separate from the main content in the `body` element. While `<title>` is the only metadata tag that is required inside `<head>`, for the next practical task we will learn about other examples of metadata that can also be included.

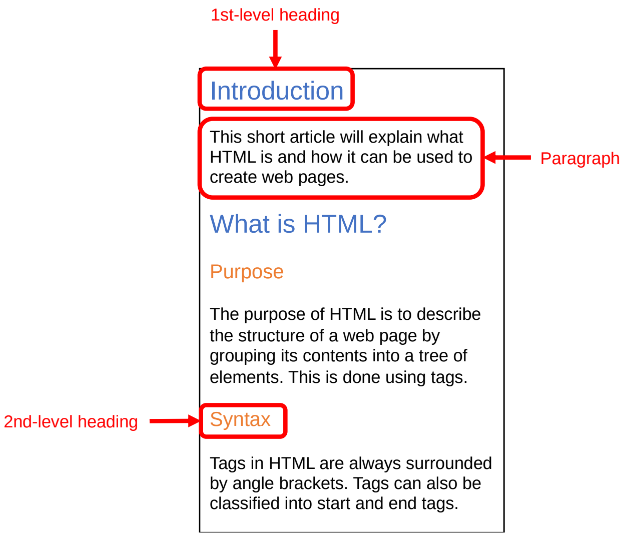
7 For each tag in the table below, circle "Normal" if it describes a normal element and circle "Void" if it describes a void element:

Tag	Description	Element Type
`<html>`	Root element	Normal / Void
`<head>`	Metadata	Normal / Void
`<title>`	Document title	Normal / Void
`<body>`	Main content	Normal / Void

Part 5: Structural Tags

Now, let us focus on the main content that goes into the `<body>` tag.

For ease of reading and navigation, long pieces of content are typically divided into logical parts such as headings and paragraphs.



Introduction

This short article will explain what HTML is and how it can be used to create web pages.

What is HTML?

Purpose

The purpose of HTML is to describe the structure of a web page by grouping its contents into a tree of elements. This is done using tags.

Syntax

Tags in HTML are always surrounded by angle brackets. Tags can also be classified into start and end tags.

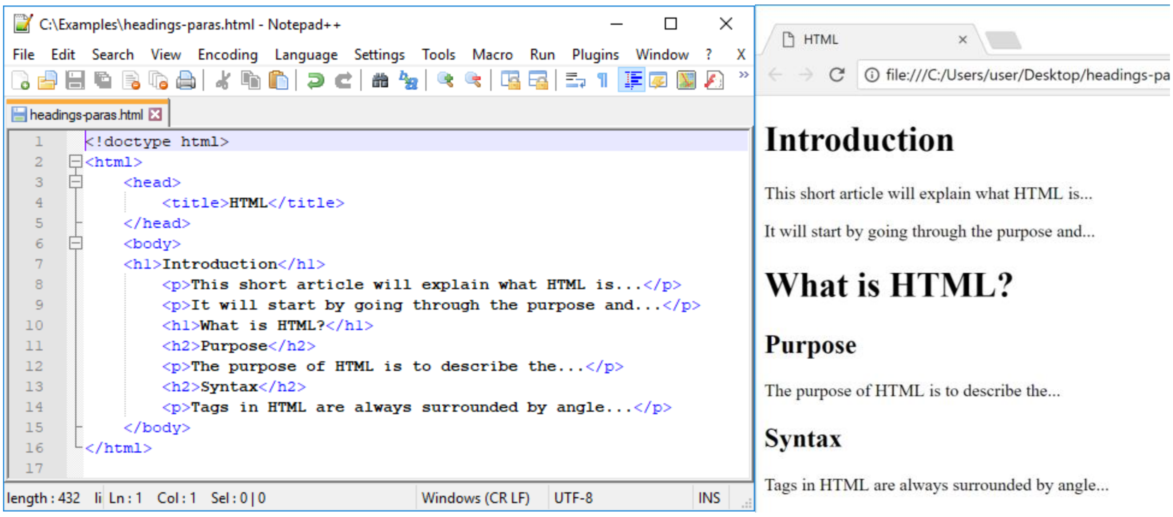
HTML provides us with 6 tags, namely `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>` and `<h6>`, to describe 1st-level headings, 2nd-level headings, 3rd-level headings and so on:

```
<body>
    <h1>Introduction</h1>
    <h1>What is HTML?</h1>
    <h2>Purpose</h2>
    <h2>Syntax</h2>
</body>
```

Content can also be organized into paragraphs using the `<p>` tag:

```
<body>
    <h1>Introduction</h1>
    <p>This short article will explain what HTML is...</p>
    <p>It will start by going through the purpose and...</p>
    <h1>What is HTML?</h1>
    <h2>Purpose</h2>
    <p>The purpose of HTML is to describe the...</p>
    <h2>Syntax</h2>
    <p>Tags in HTML are always surrounded by angle...</p>
</body>
```

Use a text editor to type out the above HTML, add any required tags (e.g., give the document a title of "HTML") and save the result as `headings-paras.html`. Double-click the document in Windows Explorer to open it in a web browser and verify that the result is similar to the screenshot on the right:



C:\Examples\headings-paras.html - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ? X
headings-paras.html
length : 432 Ln : 1 Col : 1 Sel : 0 | 0 Windows (CR LF) UTF-8 INS
HTML
file:///C:/Users/user/Desktop/headings-pa

```
<!doctype html>
<html>
    <head>
        <title>HTML</title>
    </head>
    <body>
    <h1>Introduction</h1>
        <p>This short article will explain what HTML is...</p>
        <p>It will start by going through the purpose and...</p>
        <h1>What is HTML?</h1>
        <h2>Purpose</h2>
        <p>The purpose of HTML is to describe the...</p>
        <h2>Syntax</h2>
        <p>Tags in HTML are always surrounded by angle...</p>
    </body>
</html>
```

Introduction

This short article will explain what HTML is...

It will start by going through the purpose and...

What is HTML?

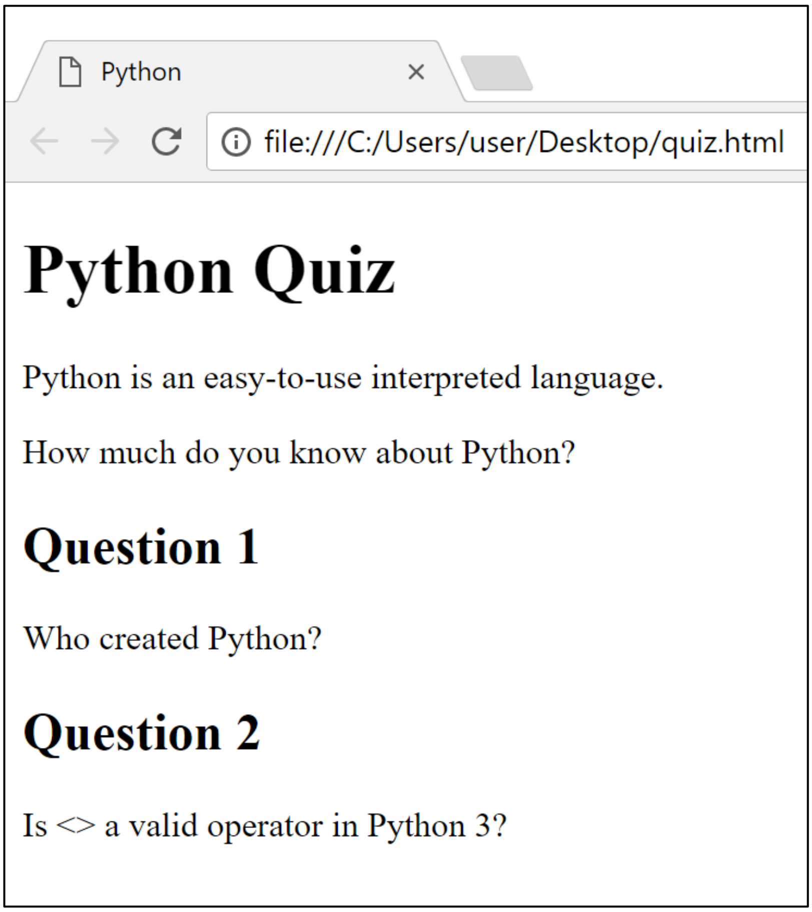
Purpose

The purpose of HTML is to describe the...

Syntax

Tags in HTML are always surrounded by angle...

8 Use only the `<html>`, `<head>`, `<title>`, `<body>`, `<h1>`, `<h2>` and `<p>` tags to create the following web page and save it as `quiz.html`:



Python Quiz

Python is an easy-to-use interpreted language.

How much do you know about Python?

Question 1

Who created Python?

Question 2

Is <> a valid operator in Python 3?

9 For each tag in the table below, circle "Normal" if it describes a normal element and circle "Void" if it describes a void element:

Tag	Description	Element Type
`<h1>` to `<h6>`	Heading	Normal / Void
`<p>`	Paragraph	Normal / Void

Part 6: Text and Media Tags

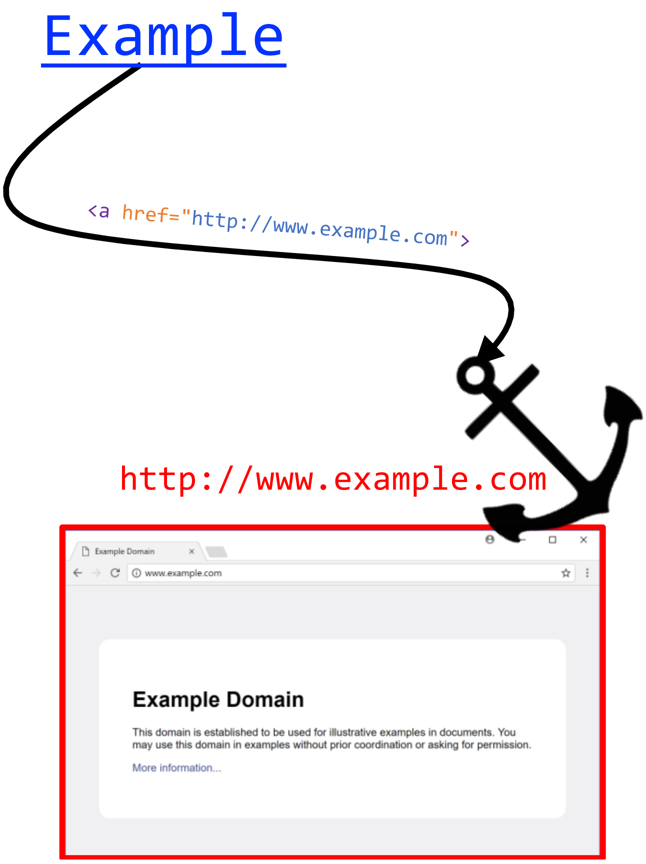
Previously, we mentioned that CSS should be used to control the appearance of web pages. However, for basic text formatting, by convention we can also use the `<b>` and `<i>` tags to format text as bold and italic respectively:

```
Text can be <b>bold</b>, <i>italic</i> or <b><i>both</i></b>.
```

Besides basic formatting, a web page may also contain images and links to other web pages. For example, the following HTML snippet describes a link to `www.example.com` using an `<a>` tag with a `href` attribute:

```
<a href="http://www.example.com">Example</a>
```

The `<a>` tag stands for "anchor" as it attaches some portion of the web page to a fixed URL, just like how a ship's anchor attaches the ship to a fixed location. The `href` attribute stands for "hypertext reference" and is set to which URL the `<a>` tag's content will be linked to.



Example Domain

This domain is established to be used for illustrative examples in documents. You may use this domain in examples without prior coordination or asking for permission.

Note that when linking to an external site, the `href` attribute must use an **absolute URL**. This means that the attribute value must include the scheme component of the URL (i.e., `http://www.example.com`) and cannot be written as `www.example.com`. (This is because `www.example.com` is interpreted as linking to a file named `www.example.com` and not the external site.)

If a URL does not include the scheme component, it is considered to be a **relative URL**. This means that the web browser will treat the URL as a path name starting from the web page's initial directory. For instance, the following HTML snippet describes a link to the `question1.html` web page in the `quiz` sub-directory:

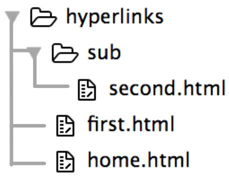
```
<a href="quiz/question1.html">View Question 1</a>
```

Be aware that while Windows uses the backslash (`\`) to separate directory names in a path, URLs always use the forward slash (`/`). You should also be aware that `..` is a special directory name that refers to the parent directory (i.e., the directory that is one level up from the current directory).

The following table shows what the resulting URL will be after following a relative URL from a given current URL of the page:

Current URL	Relative URL	Resulting URL
`http://www.sg/en`	`example`	`http://www.sg/example`
`http://www.sg/en/`	`example`	`http://www.sg/en/example`
`http://www.sg/en/sg`	`example/`	`http://www.sg/en/example/`
`http://www.sg/en/sg/`	`example/`	`http://www.sg/en/sg/example/`
`http://www.sg/en/sg`	`../example`	`http://www.sg/example`
`http://www.sg/en/sg/`	`../example`	`http://www.sg/en/example`
`http://www.sg/en/sg/`	`/example`	`http://www.sg/example`

10 Create three HTML documents using the following directory structure and contents. Open `home.html` and verify that the two links to `first.html` and `second.html` work as expected.



`home.html`

```
<!doctype html>
<html>
    <head><title>Home Page</title></head>
    <body>
        <p><a href="first.html">First link</a></p>
        <p><a href="sub/second.html">Second link</a></p>
    </body>
</html>
```

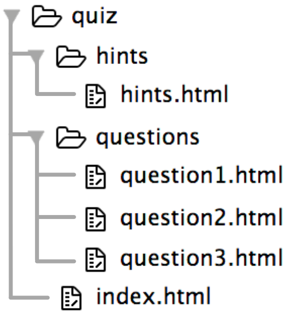
first.html

```
<!doctype html>
<html>
    <head><title>First Page</title></head>
    <body>
        <p>You are on the first page.</p>
    </body>
</html>
```

second.html

```
<!doctype html>
<html>
    <head><title>Second Page</title></head>
    <body>
        <p>You are on the second page.</p>
    </body>
</html>
```

11 You are given the following directory structure:



a. Which HTML snippet would correctly create a link from `question1.html` to `index.html`?

A	`<a href="index.html">Home</a>`
B	`<a href="questions/index.html">Home</a>`
C	`<a href="../index.html">Home</a>`
D	`<a href="../questions/index.html">Home</a>`

(   )

b. Which HTML snippet would correctly create a link from `question1.html` to `hints.html`?

A	`<a href="hints.html">Hints</a>`
B	`<a href="..\hints\hints.html">Hints</a>`
C	`<a href="../hints/hints.html">Hints</a>`
D	`<a href="http://hints.html">Hints</a>`

( )

12 Which of the following links to the MOE home page using the text "MOE"?

A	`<a href="MOE">www.moe.gov.sg</a>`
B	`<a href="MOE">http://www.moe.gov.sg</a>`
C	`<a href="www.moe.gov.sg">MOE</a>`
D	`<a href="http://www.moe.gov.sg">MOE</a>`

( )

Besides links, web pages can also include images using an `<img>` tag with the `src` and `alt` attributes:

```
<img src="images/example.png" alt="Example of an image">
```

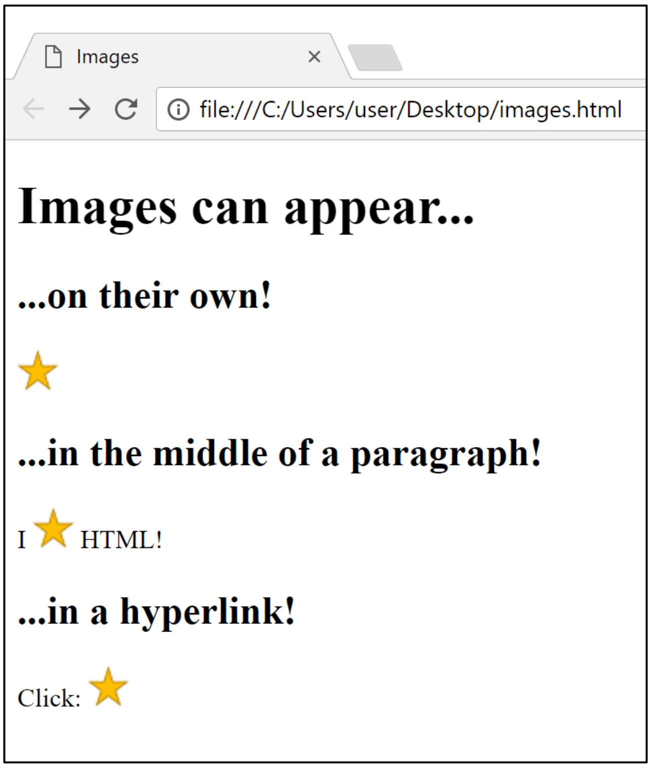
The `<img>` tag is short for "image" and its `src` attribute (short for "source") must be set to the URL of an image file. For maximum compatibility, the image file should be in either GIF, JPEG or PNG format. Furthermore, to support users who cannot or do not wish to view the image, the compulsory `alt` attribute provides an "alternative" text description that can be used as a replacement for the image.

As with the `href` attribute of the `<a>` tag, the `src` attribute of an `<img>` tag can be set to either an absolute or relative URL depending on whether the image is stored on an external site.

13 Create a small image of a star in PNG format (using graphics software such as Paint) and save it as `star.png`. Create the following HTML document and save it as `images.html` in the same directory as `star.png`.

```
<!doctype html>
<html>
    <head><title>Images</title></head>
    <body>
        <h1>Images can appear...</h1>
        <h2>...on their own!</h2>
        <img src="star.png" alt="STAR">
        <h2>...in the middle of a paragraph!</h2>
        <p>I <img src="star.png" alt="STAR"> HTML!</p>
        <h2>...in a hyperlink!</h2>
        <p>
            Click:
            <a href="http://www.example.com">
                <img src="star.png" alt="www.example.com">
            </a>
        </p>
    </body>
</html>
```

Open `images.html` and verify that the page appears as follows.



Images can appear...

...on their own!

...in the middle of a paragraph!

I HTML!

...in a hyperlink!

Click:

Also check that the last star works as a hyperlink to www.example.com.

You can also delete `star.png` and reopen `images.html` to see what happens when the required image is missing.

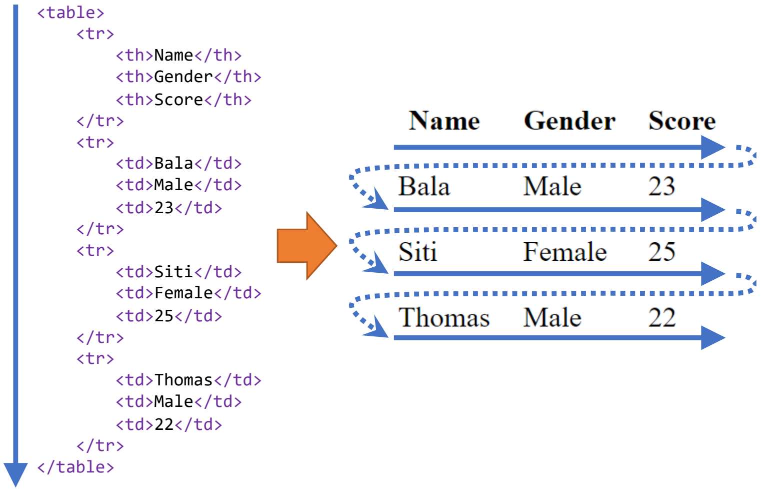
14 For each tag in the table below, circle "Normal" if it describes a normal element and circle "Void" if it describes a void element:

Tag	Description	Element Type
`<b>`	Bold (by convention)	Normal / Void
`<i>`	Italic (by convention)	Normal / Void
`<a>`	Hyperlink	Normal / Void
`<img>`	Image	Normal / Void

Part 7: Table Tags

So far, we have learn learnt how to describe headings, paragraphs, links and images in HTML. These elements follow the document's flow and are arranged top-to-bottom, left-to-right in the same order that they appear in the HTML code:

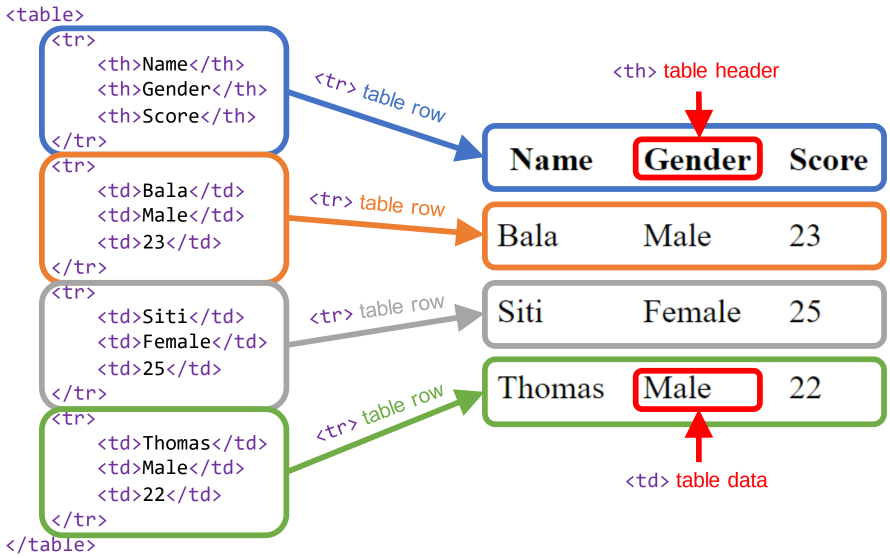
We can use the same top-to-bottom, left-to-right ordering to describe a table using the `<table>`, `<tr>`, `<th>` and `<td>` tags:



```
<table>
    <tr>
        <th>Name</th>
        <th>Gender</th>
        <th>Score</th>
    </tr>
    <tr>
        <td>Bala</td>
        <td>Male</td>
        <td>23</td>
    </tr>
    <tr>
        <td>Siti</td>
        <td>Female</td>
        <td>25</td>
    </tr>
    <tr>
        <td>Thomas</td>
        <td>Male</td>
        <td>22</td>
    </tr>
</table>
```

Name	Gender	Score
Bala	Male	23
Siti	Female	25
Thomas	Male	22

`<table>` indicates the start of one table, while `</table>` indicates the end of the table. Within those tags, each row (starting from the top) is represented by a `<tr>` (table row) tag. In turn, each row contains a sequence of `<th>` (table header) or `<td>` (table data) tags, with each tag describing a cell in that table row from left to right.

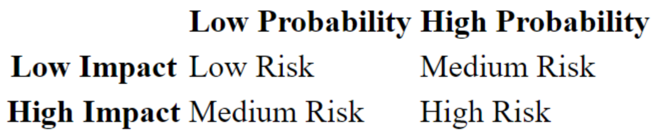


```
<table>
<tr>
    <th>Name</th>
    <th>Gender</th>
    <th>Score</th>
</tr>
<tr>
    <td>Bala</td>
    <td>Male</td>
    <td>23</td>
</tr>
<tr>
    <td>Siti</td>
    <td>Female</td>
    <td>25</td>
</tr>
<tr>
    <td>Thomas</td>
    <td>Male</td>
    <td>22</td>
</tr>
</table>
```

Name	Gender	Score
Bala	Male	23
Siti	Female	25
Thomas	Male	22

Note that while both <th> and <td> are used to describe individual table cells, <th> should only be used for **header cells** that contain row or column titles.

15 Use the <table>, <tr>, <th> and <td> tags to create the following table and save it in risk-assessment.html:



	Low Probability	**High Probability**
Low Impact	Low Risk	Medium Risk
High Impact	Medium Risk	High Risk

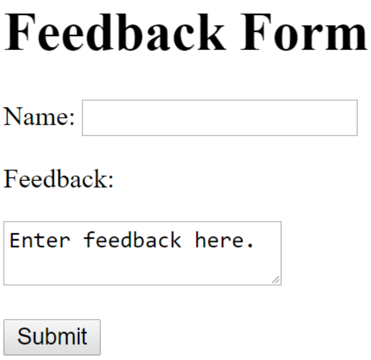
16 For each tag in the table below, circle "Normal" if it describes a normal element and circle "Void" if it describes a void element:

Tag	Description	Element Type
<table>	Table	Normal / Void
<tr>	Table row	Normal / Void
<th>	Table header cell	Normal / Void
<td>	Table cell	Normal / Void

Do not worry about the lack of table borders at the moment. You will learn about how to add them in the next practical task.

Part 8: Form Tags

An important advantage of web pages over conventional documents is that web pages can collect inputs from the reader using forms such as the following example:



Feedback Form

Name:

Feedback:

Enter feedback here.

Such forms are created in HTML using the `<form>`, `<input>` and `<textarea>` tags:

```html
<form action="http://www.example.com">
    <h1>Feedback Form</h1>
    <p>Name: <input name="username" type="text" value=""></p>
    <p>Feedback:</p>
    <textarea name="feedback">Enter feedback here.</textarea>
    <p><input type="submit"></p>
</form>
```

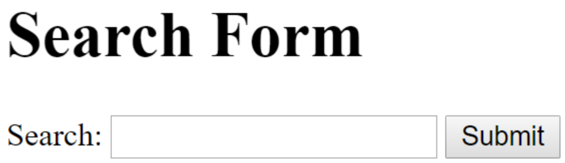
Each form is contained in a separate `<form>` tag with an `action` attribute set to the URL where the submitted data will be sent to. (In future practical tasks, you will learn how to write a Python program that runs on a web server to process this data.) Inside the `<form>` tag, you can use the `<input>` and `<textarea>` tags in addition to any of the section, text, media and table tags that you have already learnt. Each `<input>` and `<textarea>` tag represents an **input control** and may have a unique `name` attribute to help the server retrieve these inputs.

An `<input>` tag can represent either a single-line text box or submit button, depending on whether the `type` attribute is set to `"text"` or `"submit"`. For a text box, the `value` attribute determines the text box's initial contents. For a submit button, the `value` attribute determines the button's label. Note that for either case, the `<input>` tag does not have an end tag.

An `<input>` tag can also set its type attribute to "`file`". This allows the user to upload files for submission with the form. We shall learn how to handle file uploads in a future practical task.

The `<textarea>` tag, on the other hand, represents a multi-line text box. Unlike the `<input>` tag that does not have an end tag, the `<textarea>` tag does require an end tag and the text inside will be used as the text box's initial content.

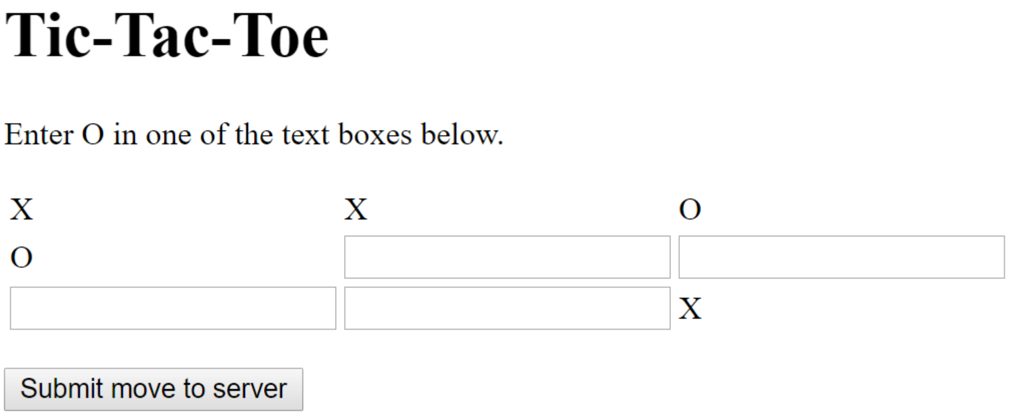
17 Use the `<html>`, `<head>`, `<title>`, `<body>`, `<form>`, `<input>`, `<textarea>`, `<h1>` and `<p>` tags to create the following form and save it in `search.html`. (You may choose any URL for the `action` attribute of the form.)



Search Form

Search:

18 Use the `<html>`, `<head>`, `<title>`, `<body>`, `<form>`, `<input>`, `<textarea>`, `<h1>`, `<p>` and the table tags to create the following form and save it in `game.html`. (You may choose any URL for the `action` attribute of the form.)



Tic-Tac-Toe

Enter O in one of the text boxes below.

X	X	O
O		
		X

19 For each tag in the table below, circle "Normal" if it describes a normal element and circle "Void" if it describes a void element:

Tag	Description	Element Type
`<form>`	User-submittable form	Normal / Void
`<input>`	Input control	Normal / Void
`<textarea>`	Text input area	Normal / Void