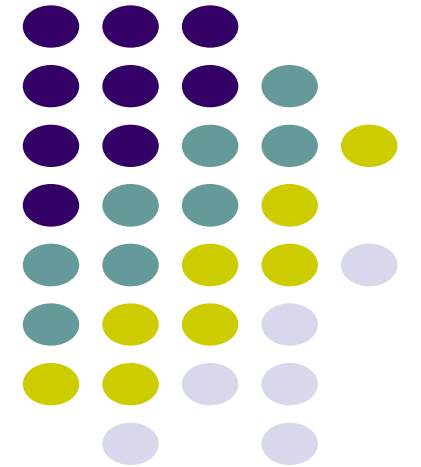
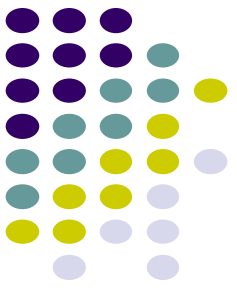


Introduction to HTML



HTML



- Stands for HyperText Markup Language
- Used to create a Web page
- Made up of tags that specify the structure of the document (this section is a heading, this section is a paragraph, etc..)
- An excerpt from a sample HTML document:

```
<html>

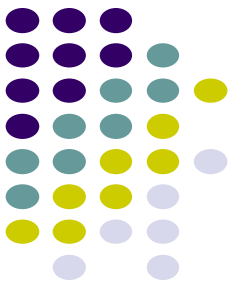
  <head>

    <title>Bob's Web page</title>

  </head>

  <body>

    <h1>This is my first Web page</h1>
```

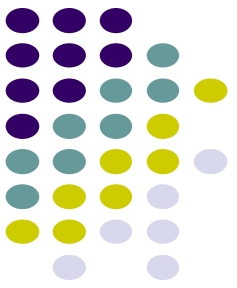


HTML Tags

- Most HTML tags work in pairs. There is an opening and a closing tag. For example:

`<p>Some content here.</p>`

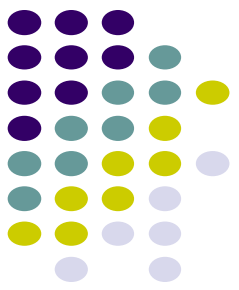
- The `<p>...</p>` tag displays a paragraph
- `<p>` opens the paragraph (opening tag)
- `</p>` closes the paragraph (closing tag)
- “Some content here.” will be displayed on the page



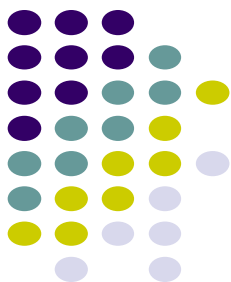
How does HTML work?

- HTML stands for HyperText Markup Language. "Markup language" means that, rather than using a programming language to perform functions, HTML uses tags to identify different types of content and the purposes they each serve to the webpage.

Example



- Take a look at the article below.
- If I were to ask you to label the types of content on the page, you'd probably do pretty well:
 - There's the header at the top,
 - then a subheader below it,
 - the body text,
 - and some images at the bottom
 - followed by a few more bits of text....



Orchids of South-East England

A successful Associate Panel by Mark Monckton ARPS

I had been interested in photography for many years when, in 2006, I decided it was time to join the Royal Photographic Society. My choice for my free one year membership of a special interest group was Nature.

I applied for, and was successful in obtaining, my LRPS distinction in the same year with a panel of wildlife pictures. At that point I decided to start working towards my Associateship. I knew that I wanted to produce a Nature panel but was not sure what subject to focus on.

During the following year I attended a two day workshop on the 'Wild Orchids of Kent' organised by Kent Wildlife Trust (www.kentwildlifetrust.org.uk) and lead by Fred Booth. At this point I realised how interesting it would be to locate and photograph wild orchids around the South East of England (all within a two hour drive from my home). My idea, at this point was to also include fungi in the panel, but I later decided that my orchid pictures were stronger and, at that point, I almost had enough for a single subject panel.

I took the majority of the photographs during the rest of 2007 and throughout 2008. Towards the end of 2008 I arranged a one-to-one with George McCarthy FRPS to show him my panel in progress and obtain advice as to how best complete it. I already knew George as I had attended several of

his workshops over the years (including, co-incidentally, one on orchids) and have always admired his work.

Originally my panel contained a mix of orchids and fungi but it was George who convinced me to go with all orchids. He pointed out that the best of my orchid pictures were far better than the fungi pictures and if I could present a panel with just orchids it would be much stronger. At that point I had twelve good orchid pictures so needed to get at least three more.

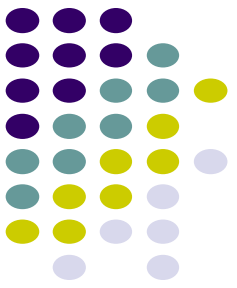
Early in 2009 I did a lot of research, mainly on the internet, to identify the best places to look for some of the Orchids I did not already have pictures of. I photographed Early Spider Orchid, Early Marsh Orchid, Bee Orchid, Military Orchid and Burnt Orchid to complete my panel.

My idea for the panel was to take close up portraits of the flower spike of each species of orchid, and therefore needed the minimum distractions in the background. In order to achieve this I used a Canon 180mm f3.5 L macro lens and a Canon 5D & Canon 1Ds Mk III. The 180mm macro lens in combination with the full frame sensor of the 1Ds helped to produce the background I wanted.

By the end of June 2009 I had 19 orchid photographs to choose from. I printed them all out

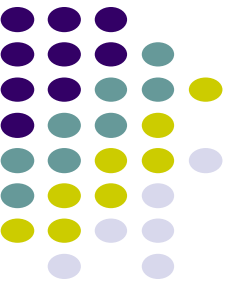
Layout





What is this?

- Markup languages work in the same way as *you* just did when you labeled those content types, except they use code to do it -- specifically, they use HTML tags, also known as "elements." These tags have pretty intuitive names: Header tags, paragraph tags, image tags, and so on.
- Every web page is made up of a bunch of these HTML tags denoting each type of content on the page. Each type of content on the page is "wrapped" in, i.e. surrounded by, HTML tags.



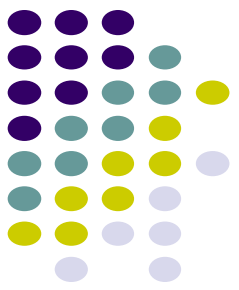
Self-closing Tags

- Some HTML tags are self closing. For example:

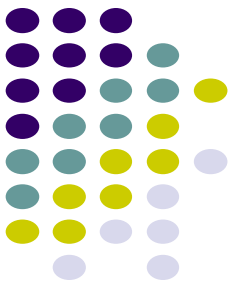
`
`

- The `
` tag will display a line break.

Required Tags

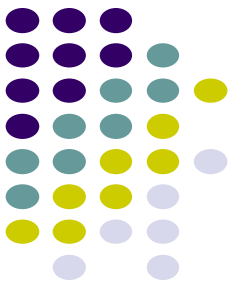


- All HTML documents should have **html**, **head** and **body** tags, along with the **DOCTYPE** identifier.
 - **!DOCTYPE** – Tells the browser which set of standards the page adheres to
 - **<html>...</html>** -- Surrounds the contents of the entire page
 - **<head>...</head>** -- Lists the identification information on the page, such as the title
 - **<title>...</title>** -- Gives the name of the page that appears in the top of the browser window
 - **<body>...</body>** -- Frames the content of the page to be displayed in the browser



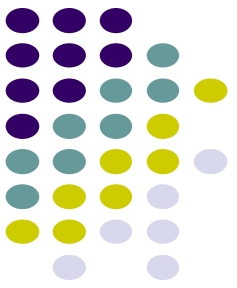
Tags.. What they mean?

- **<!DOCTYPE HTML>**: This special tag is used to inform the browser that
- the document type is HTML. This is how the browser knows you'll be
- writing an HTML5 document. You will sometimes see other values for the doctype, but HTML5 is the way to go these days.



Language

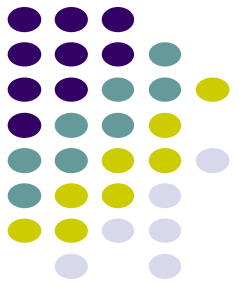
- **<html lang = “en”></html>**: The **<html>** tag is the foundation of
- the entire web page. The tag begins the page. Likewise, **</html>** ends the page.
- For example, the page begins with **<html>** and ends with **</html>**.
- The **<html></html>** combination indicates that everything in the page is defined as HTML code.
- In HTML5, you’re expected to tell the browser which language the page will be written in.
- Because we write in English, specifying with the code **“en.”**



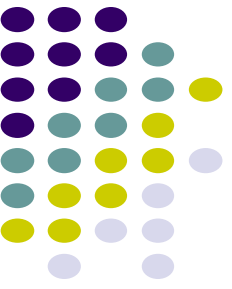
Meta tag

- **<meta charset="UTF-8">**: The meta tag is used to provide a little more information to the browser.
- This command gives a little more information to the browser, telling it which character set to use.
- English normally uses a character set called (for obscure reasons) UTF-8.

Comments .. Not for execution

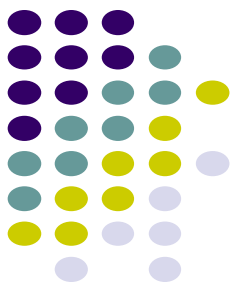


- `<!--/-->`: This tag indicates a *comment*, which is ignored by the browser.
- However, a comment is used to describe what's going on in a particular part of the code.



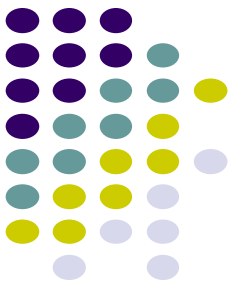
More on tags

- **<title></title>**: This tag is used to determine the page's title. The title usually contains ordinary text.
- **<body></body>**: The page's main content is contained within these tags.
- **<h1></h1>**: H1 stands for *heading level one*.
- **<p></p>**: p stands for the paragraph tag.



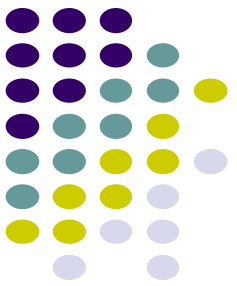
Basic Elements

- A text header, denoted using the `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>`, `<h6>` tags.
- A paragraph, denoted using the `<p>` tag.
- A horizontal ruler, denoted using the `<hr>` tag.
- A link, denoted using the `<a>` (anchor) tag.
- A list, denoted using the `` (unordered list), `` (ordered list) and `` (list element) tags.
- An image, denoted using the `` tag
- A divider, denoted using the `<div>` tag
- A text span, denoted using the `` tag



Text headers and paragraphs

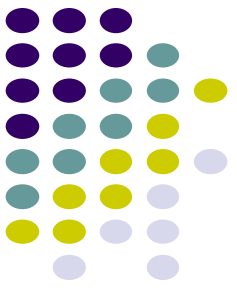
- six different types of text header you can choose from, h1 being the topmost heading with the largest text, and h6 being the most inner heading with the smallest text.
- In general, you should have only one h1 tag with a page, since it should be the primary description of the HTML page.



Example of Header –Save this and execute

```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
<h1>My First Page</h1>
<p>This is my first page.</p>
<h2>A secondary header.</h2>
<p>Some more text.</p>
</body> </html>
```

Horizontal rulers



- A horizontal ruler `<hr>` tag acts as a simple separator between page sections.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
</head>
```

```
<body>
```

```
<h1>My First Page</h1>
```

```
<p>This is my first page.</p>
```

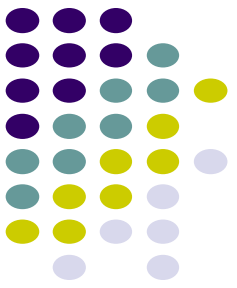
```
<hr/>
```

```
<p>This is the footer - all rights are reserved to me.</p>
```

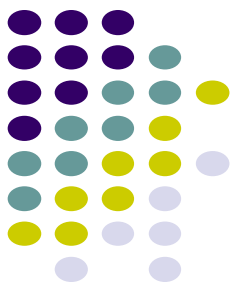
```
</body>
```

```
</html>
```

Exercise



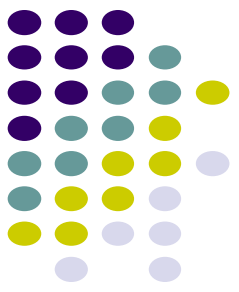
- Add an HTML `<h1>` tag with the text "I'm the most important!"
- Add an HTML `<h2>` tag with the text "I'm less important!"
- Add an HTML `<h6>` tag with the text "I'm the least important!"



Text Formatting

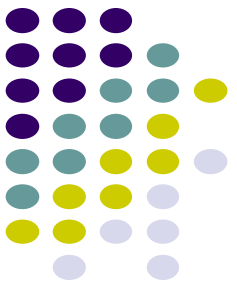
- Text formatting tags modify the text between the opening tag and the closing tag
 - Ex. `Hello` makes “Hello” bold

<code></code>	bold
<code><i></i></code>	<i>italicized</i>
<code><u></u></code>	<u>underlined</u>
<code><sup></sup></code>	Sample ^{superscript}
<code><sub></sub></code>	Sample _{subscript}
<code></code>	strong
<code></code>	<i>emphasized</i>
<code><pre></pre></code>	Preformatted text
<code><blockquote></blockquote></code>	Quoted text block
<code></code>	Deleted text – strike-through



Links

- A link ("anchor") is a small span of text that will direct you to a different section in the page, or to a different page. To create a link, you will need to specify where you would like the user to be directed to when the link is clicked by specifying the href attribute.
- For example:
- `A link to Google`
- To create a link to a different section in the same page, you will need to use a hash sign along with the element ID to where you would like the browser to jump to.
- For example:
- `Click here to read the Frequently Asked Questions</p>`



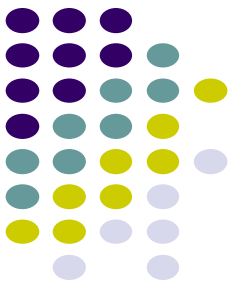
Hyperlinks: <a> Tag (2)

- Link to an external Web site:

```
<a href="http://www.devg.org" target="_blank">BASD</a>
```

- Always use a full URL, including "http://", not just "www.somesite.com"
- Using the target="_blank" attribute opens the link in a new window
- Link to an e-mail address:

```
<a href="mailto:bugs@example.com?subject=Bug+Report">  
Please report bugs here (by e-mail only)</a>
```



Hyperlinks: <a> Tag

- Link to a document called form.html on the same server in the same directory:

```
<a href="form.html">Fill Our Form</a>
```

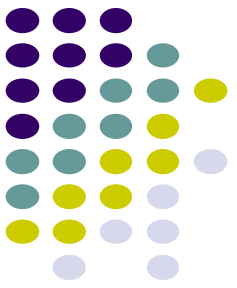
- Link to a document called parent.html on the same server in the parent directory:

```
<a href=" ../parent.html">Parent</a>
```

- Link to a document called cat.html on the same server in the subdirectory stuff:

```
<a href="stuff/cat.html">Catalog</a>
```

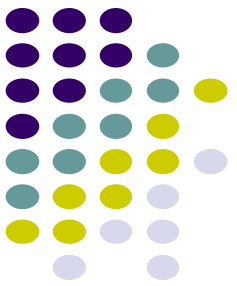
Element ID



- The element ID is denoted using the id attribute:

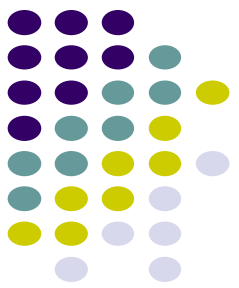
```
<h3 id="faq">Frequently asked questions  
</h3>
```

```
<p>The first rule about fight club is that you do  
not talk about fight club.  
</p>
```

Try this now...

```
<!DOCTYPE html>
<html> <head> </head> <body>
<h1>My First Page</h1>
<p>This is my first page.</p>
<a href="#faq">Click here to read the Frequently Asked
Questions</a>
<hr/> <h3 id="faq">Frequently asked questions</h3>
<p>The first rule about fight club is that you do not talk
about fight club.</p>
<p>However, if you do have questions, please e-mail me at
foo@bar.com</p>
</body> </html>
```



Lists

- HTML provides a way to create both an ordered list (with elements counting up, 1, 2, 3...) and an unordered list with bullets instead of numbers. Lists are a good way to formalize a list of items and let the HTML styling do the work for you.

- **Ordered lists**

- Here is an example of how to create an ordered list:

```
<p>Here is a list of ordered items:</p>
```

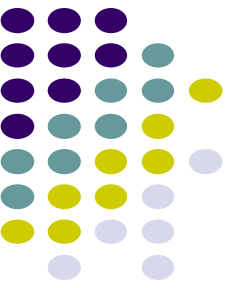
```
<ol>
```

```
  <li>First item</li>
```

```
  <li>Second item</li>
```

```
  <li>Third item</li>
```

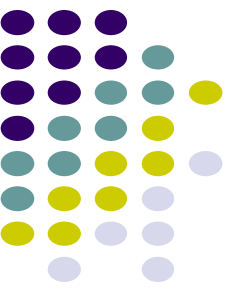
```
</ol>
```



Ordered Lists

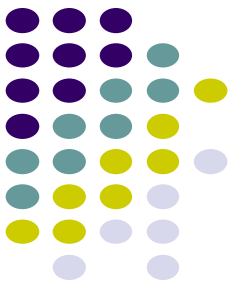
- Ordered lists have a "type" attribute which defines the numbering convention to use.
- To count using numbers, use type="1":

```
<p>Here is a list of ordered items:</p> <ol type="1">  
  <li>First item</li>  
  <li>Second item</li>  
  <li>Third item</li>  
</ol>
```



Types:

- To count using uppercase letters, use type="A"
- To count using lowercase letters, use type="a"
- To count using uppercase roman numerals, use type="I"
- To count using lowercase roman numerals, use type="i"



Unordered lists

- Example to create an unordered list:

`<p>Here is a list of unordered items:</p>`

``

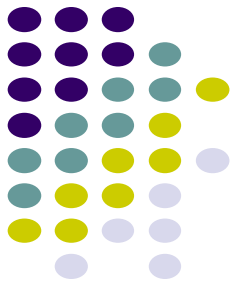
`First item`

`Second item`

`Third item`

``

Style attributes

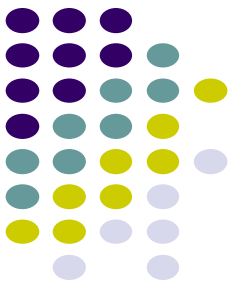


- To change the list style attributes, we can use the CSS attribute called list-style-type.
- The available types are:
 - disc
 - circle
 - square
 - none

<p>Here is a list of unordered items:</p>

<ul style="list-style-type: disc">

First item Second item Third item



Definition lists: <dl> tag

- Create definition lists using <dl>
 - Pairs of text and associated definition; text is in <dt> tag, definition in <dd> tag

```
<dl>  
  <dt>HTML</dt>  
  <dd>A markup language ...</dd>  
  <dt>CSS</dt>  
  <dd>Language used to ...</dd>  
</dl>
```

- Renders without bullets
- Definition is indented

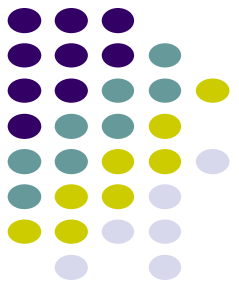
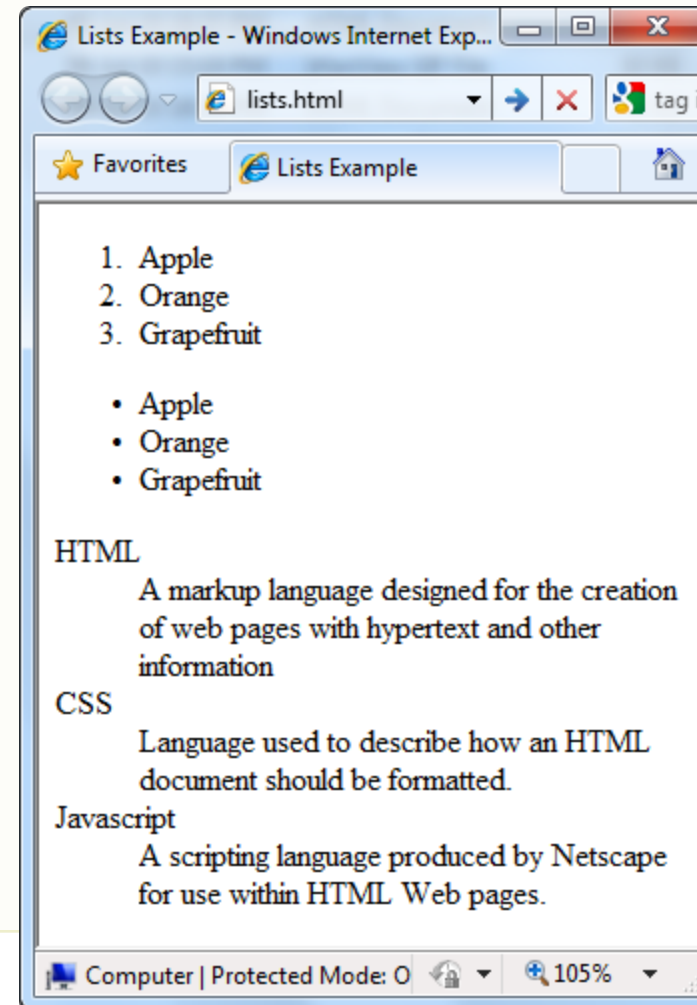
Lists – Example

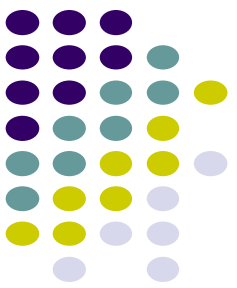
```
<ol type="1">
  <li>Apple</li>
  <li>Orange</li>
  <li>Grapefruit</li>
</ol>

<ul type="disc">
  <li>Apple</li>
  <li>Orange</li>
  <li>Grapefruit</li>
</ul>

<dl>
  <dt>HTML</dt>
  <dd>A markup lang...</dd>
</dl>
```

lists.html





The <div> Tag

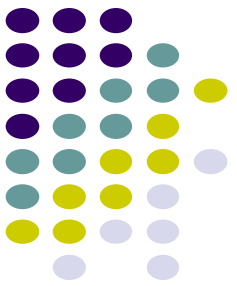
- <div> creates logical divisions within a page
- Block style element
- Used with CSS
- Example:

div-and-span.html



```
<div style="font-size:24px; color:red">DIV  
example</div>
```

```
<p>This one is <span style="color:red; font-  
weight:bold">only a test</span>.</p>
```



The Tag

- Inline style element
- Useful for modifying a specific portion of text
 - Don't create a separate area document
- Very useful with CSS

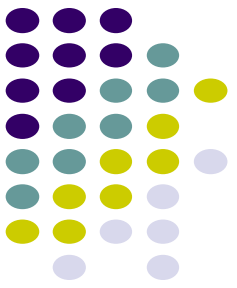
span.html

```
<p>This one is <span style="color:red; font-weight:bold">only a test</span>.</p>
```

```
<p>This one is another <span style="font-size:32px; font-weight:bold">TEST</span>.</p>
```



Images



- Images in HTML are inline elements that can be placed within a paragraph. To add an image, use the `` tag along with the `src` attribute to specify the location of the image.

```

```

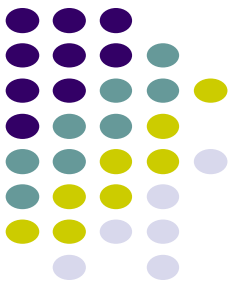
```

```

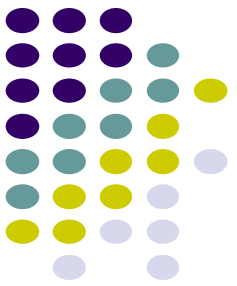
```

```

Image Types



- There are three main types of image formats:
 - Lossless formats - useful for when you need pixel-perfect graphics, for example for logos. The most common format is PNG.
 - Lossy formats - useful for displaying rich images. The most common format used in this category is JPG
 - Animated formats - useful for showing short animated images. The most common format is GIF



Images: **** tag

- ◆ Inserting an image with **** tag:

```

```

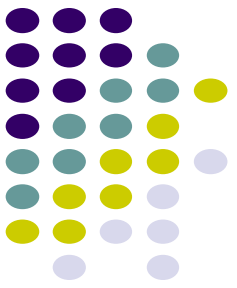
- ◆ Image attributes:

src	Location of image file (relative or absolute)
alt	Substitute text for display (e.g. in text mode)
height	Number of pixels of the height
width	Number of pixels of the width
border	Size of border, 0 for no border

- ◆ Example:

```

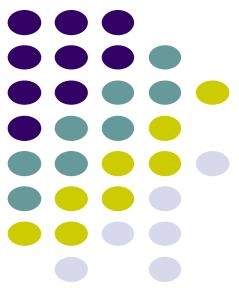
```



HTML- Image Maps

- Image with clickable areas.
- `<map>` tag - define a client-side image-map.
- attribute of the `<map>` element is associated with the ``'s `usemap` attribute and creates a relationship between the image and the map.
- The `<map>` element contains a number of `<area>` elements, that defines the clickable areas in the image map.

Image Map- Example



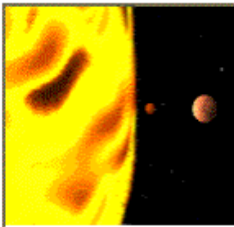
```
<!DOCTYPE html>
<html>
<body>

<p>Click on the sun or on one of the planets to watch it closer:</p>

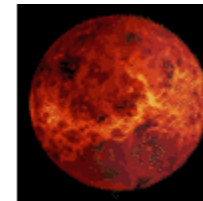


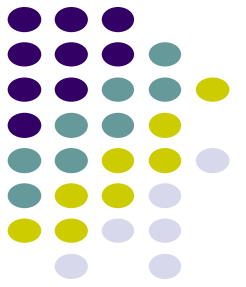
<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" alt="Sun" href="sun.htm">
  <area shape="circle" coords="90,58,3" alt="Mercury" href="mercur.htm">
  <area shape="circle" coords="124,58,8" alt="Venus" href="venus.htm">
</map>
|
</body>
</html>
```

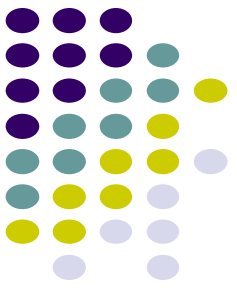
Click on the sun or on one of the planets to watch it closer:



Clicking on Larger Circle --→



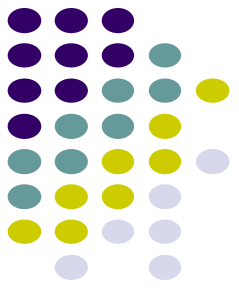




Exercise

- Add a image within the body, tag with the image "photo.jpg".
- Adjust size of img to 100px by 100px
- Add onload command, `onload="alert()"` with the text "task complete"

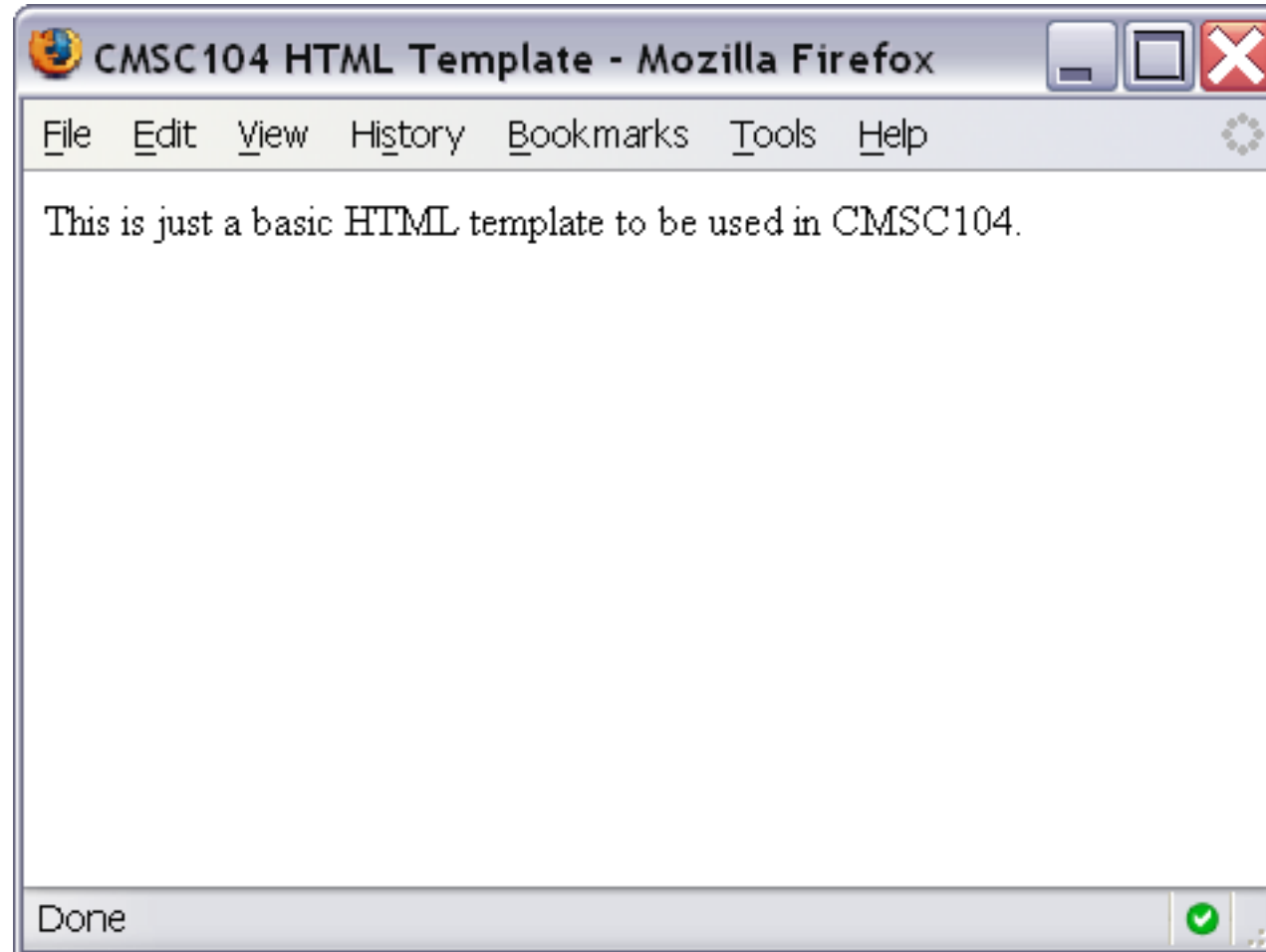
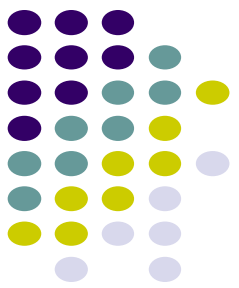
Basic HTML Template



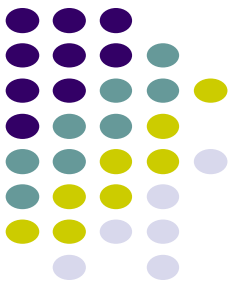
```
<!DOCTYPE html>
<html>
  <head>
    <title>CMSC104 HTML Template</title>
  </head>
  <body>
    This is just a basic HTML template to be used in CMSC104.
  </body>
</html>
```

Example file: template.html

Basic HTML Template Screenshot

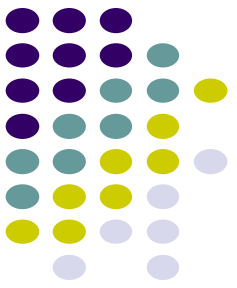


Some Common HTML Tags and Their Meanings



- `<p>...</p>` -- Creates a paragraph
- `
` -- Adds a line break
- `<hr />` -- Separates sections with a horizontal rule
- `<h1>...</h1>` -- Displays a heading (h1-h6)
- `<!--...-->` -- Inserts a comment
- `...` -- Creates an ordered list
- `...` -- Creates an unordered list
- `` -- Inserts an image into the document
- `<a>...` -- Inserts a link into the document

Paragraph Example



<p>

The exam next week will consist of T/F,
multiple choice, short answer and pseudocode
questions. You cannot use a calculator.

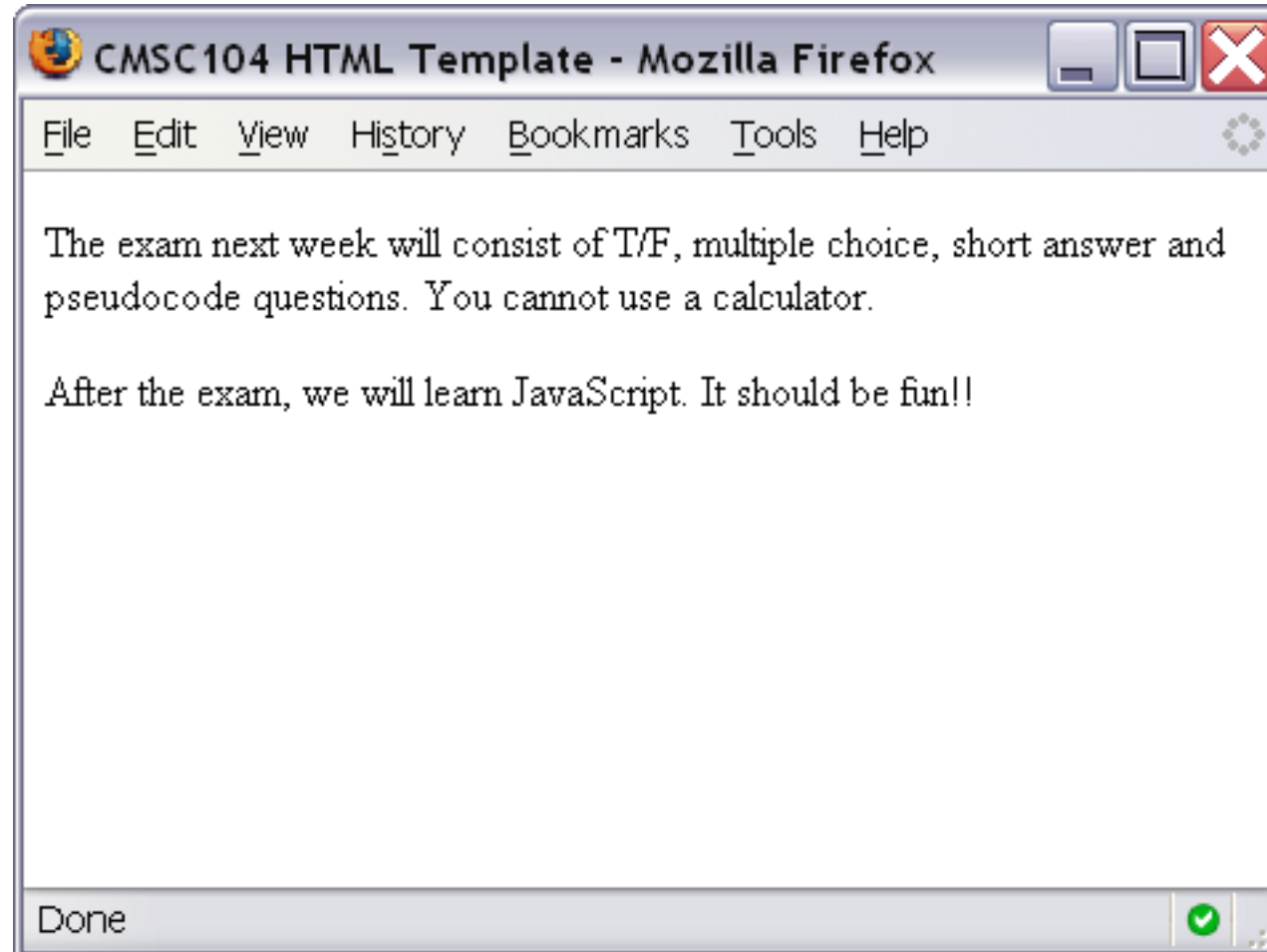
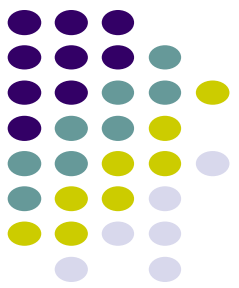
</p>

<p>

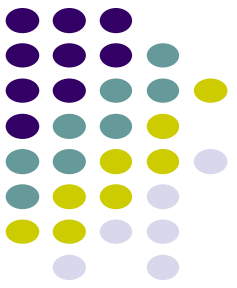
After the exam, we will learn JavaScript.
It should be fun!!

</p>

Paragraph Example Screenshot

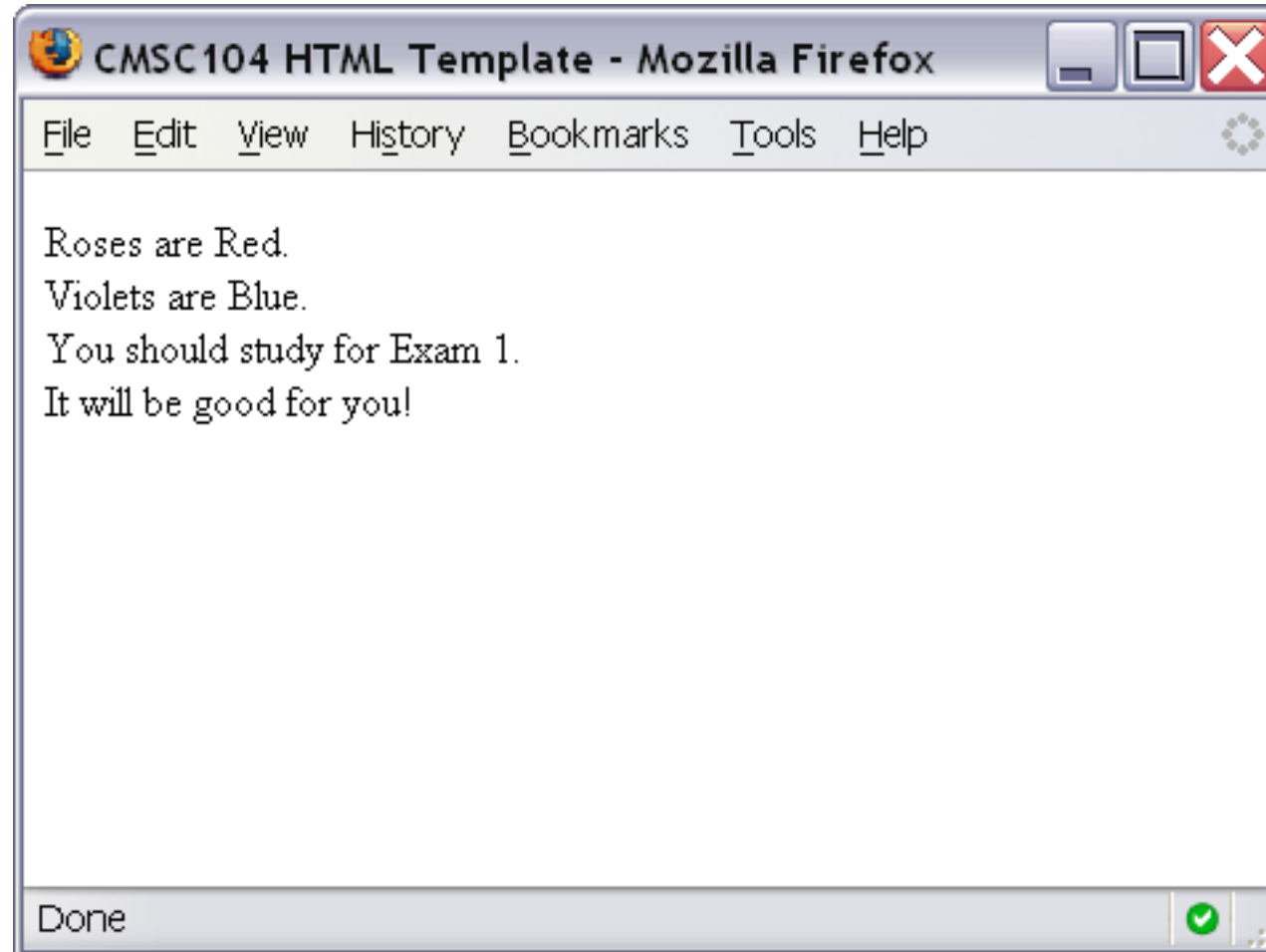
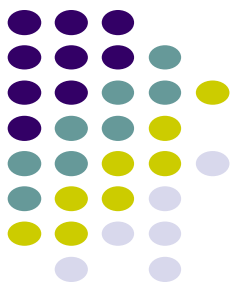


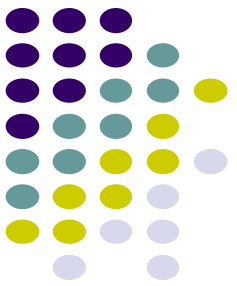
Line Break Example



```
<p>  
  Roses are Red. <br />  
  Violets are Blue. <br />  
  You should study for Exam 1. <br />  
  It will be good for you!  
</p>
```

Line Break Example Screenshot





Horizontal Rule Example

```
<p>
```

```
    The exam next week will consist of T/F,  
    multiple choice, short answer and  
    pseudocode questions.  You cannot use a  
    calculator.
```

```
</p>
```

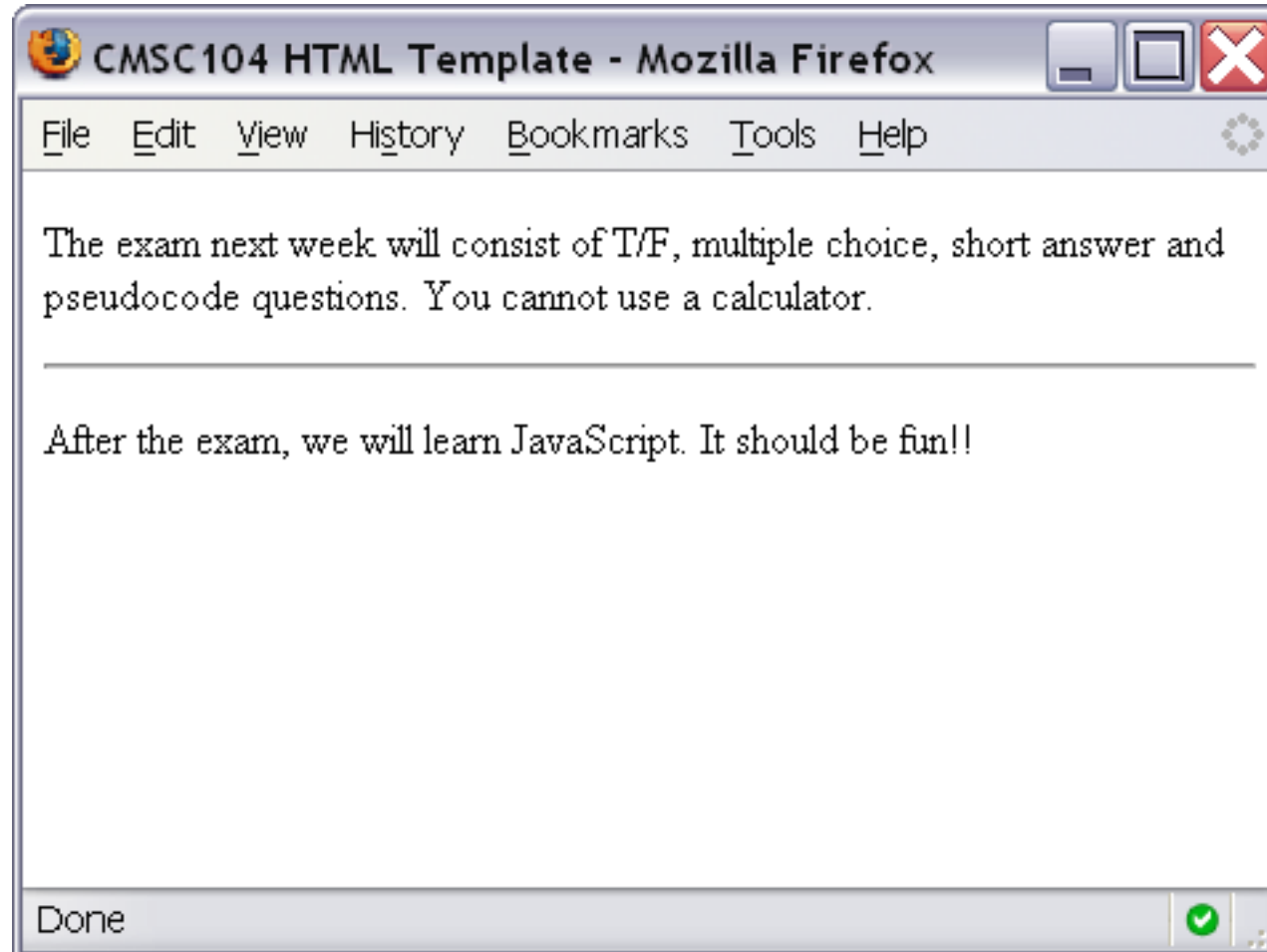
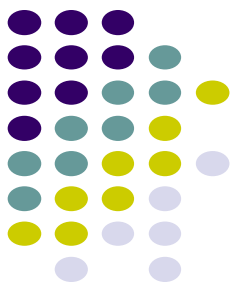
```
<hr />
```

```
<p>
```

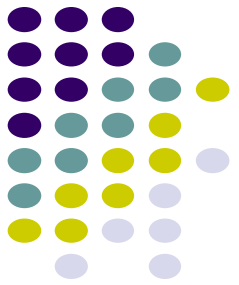
```
    After the exam, we will learn JavaScript.  
    It should be fun!!
```

```
</p>
```

Horizontal Rule Example Screenshot



Heading Example



```
<h1>This is heading 1</h1>
```

```
<h2>This is heading 2</h2>
```

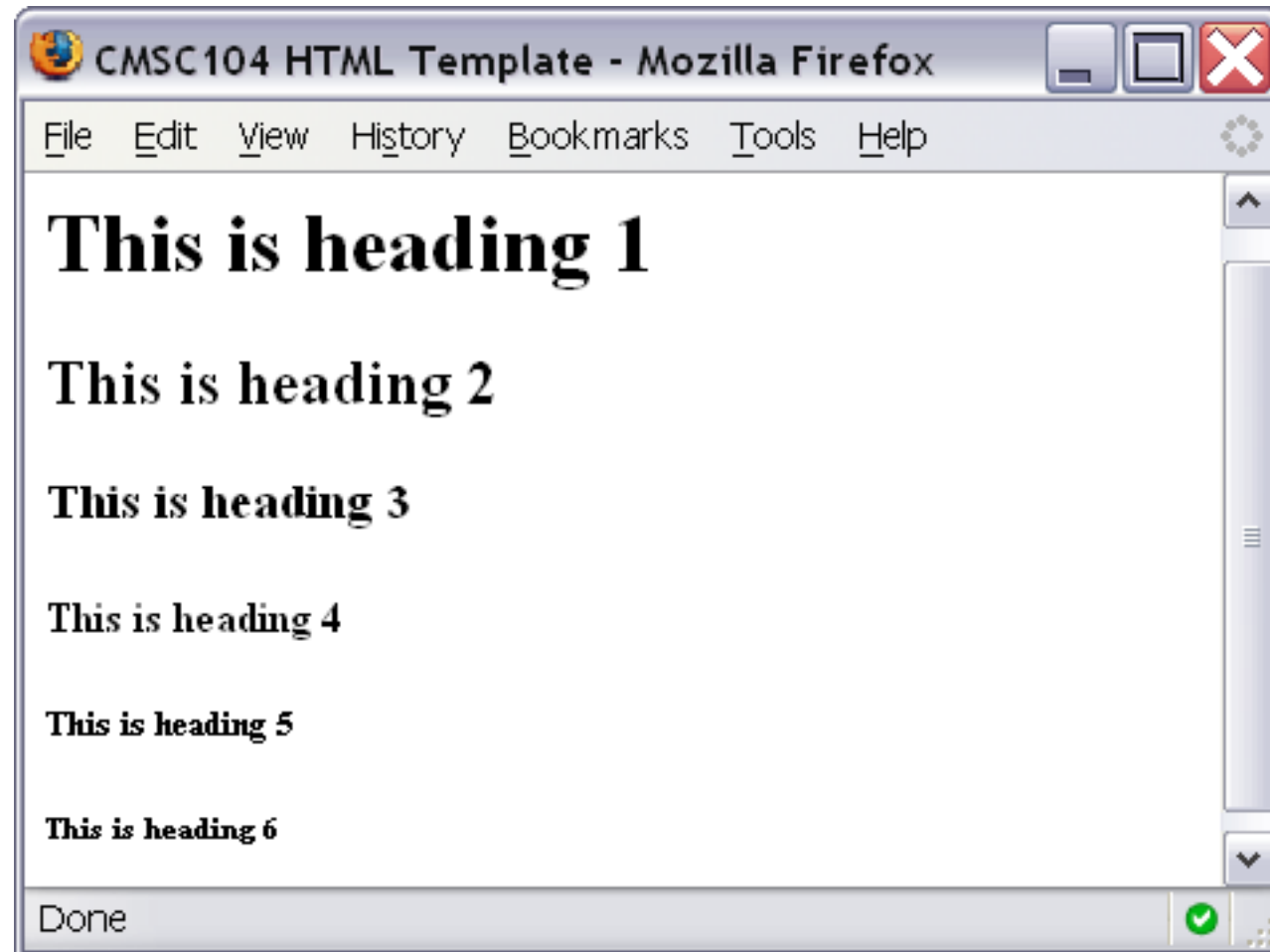
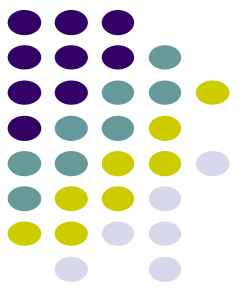
```
<h3>This is heading 3</h3>
```

```
<h4>This is heading 4</h4>
```

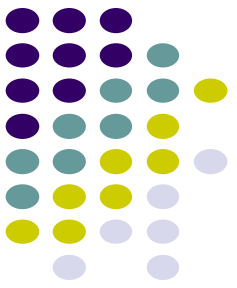
```
<h5>This is heading 5</h5>
```

```
<h6>This is heading 6</h6>
```

Heading Example Screenshot



Comment Example



```
<!-- This is just some sample html  
to illustrate the use of a  
comment -->
```

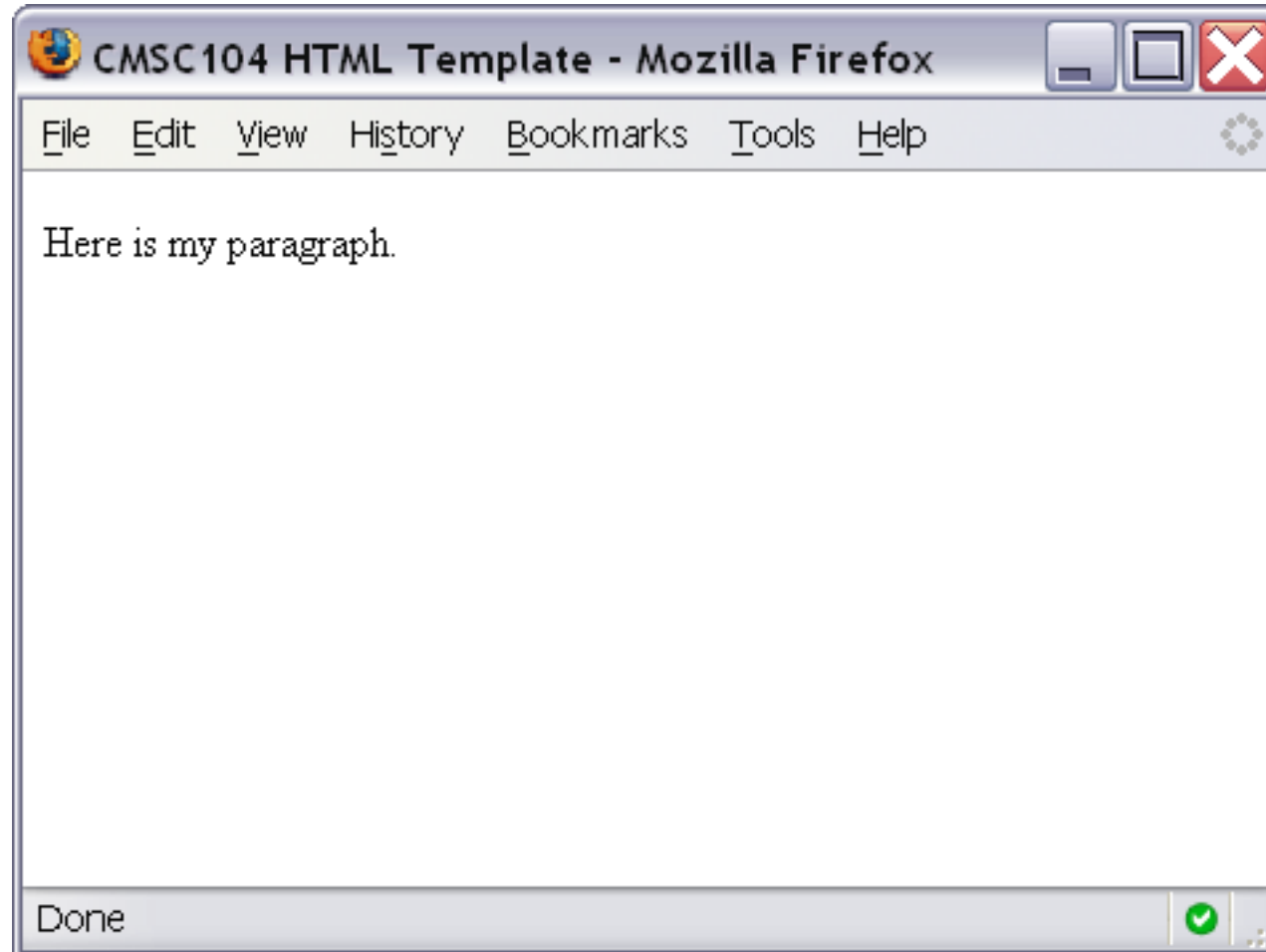
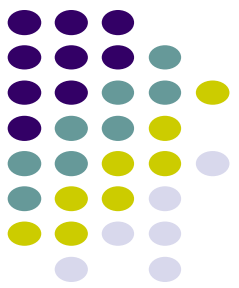
```
<p>
```

```
    Here is my paragraph.
```

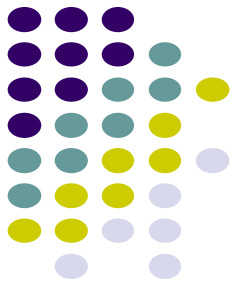
```
</p>
```

```
<!-- Here is another comment -->
```

Heading Example Screenshot

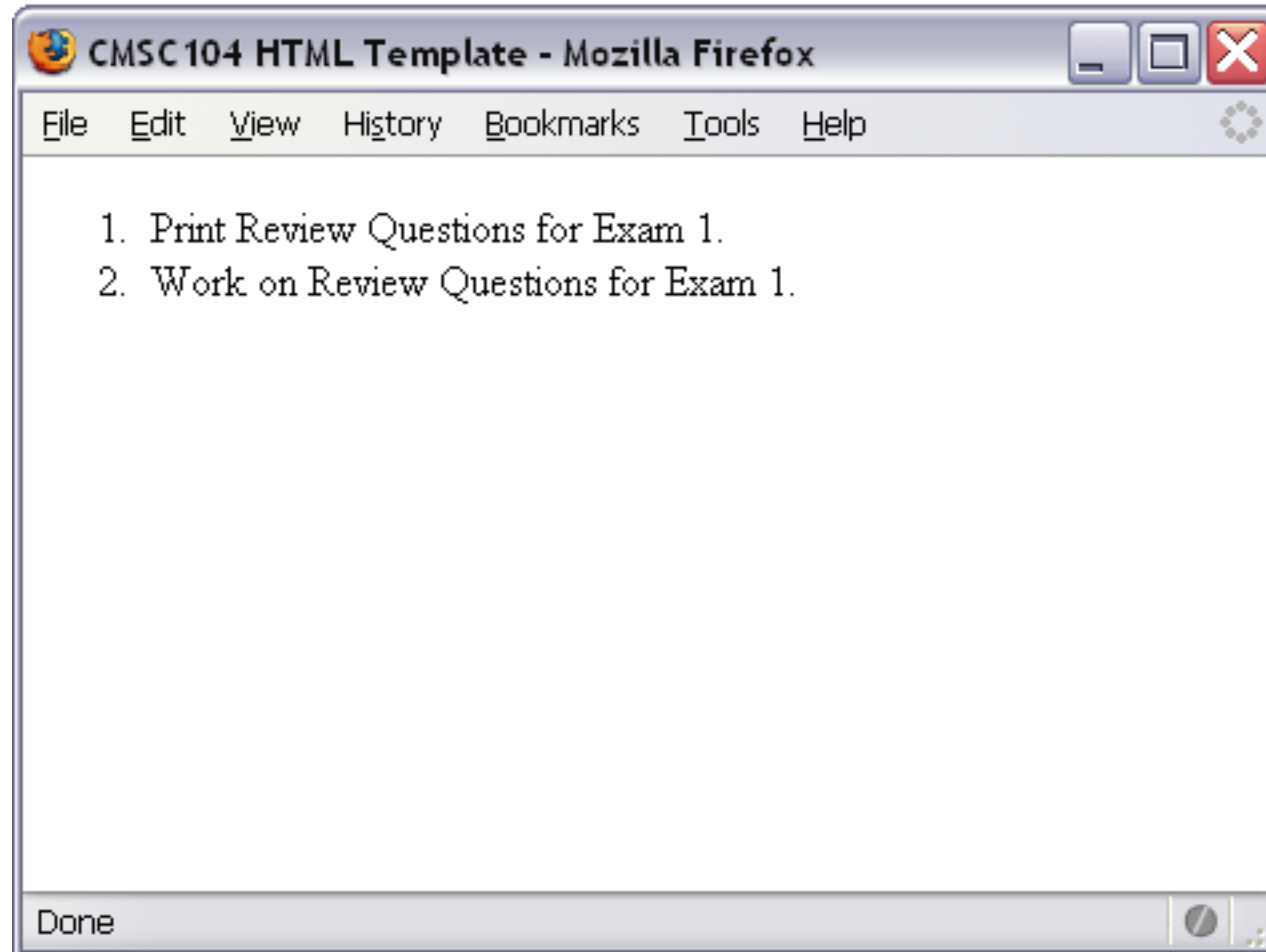
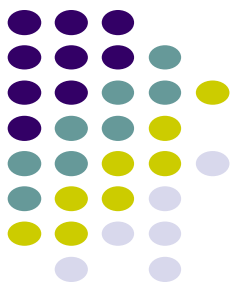


Ordered List Example

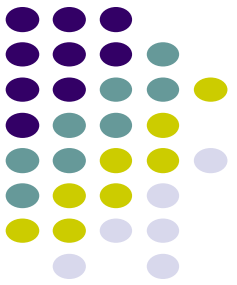


```
<ol>  
  <li>Print Review Questions for Exam 1.</li>  
  <li>Work on Review Questions for Exam 1.</li>  
</ol>
```

Ordered List Screenshot

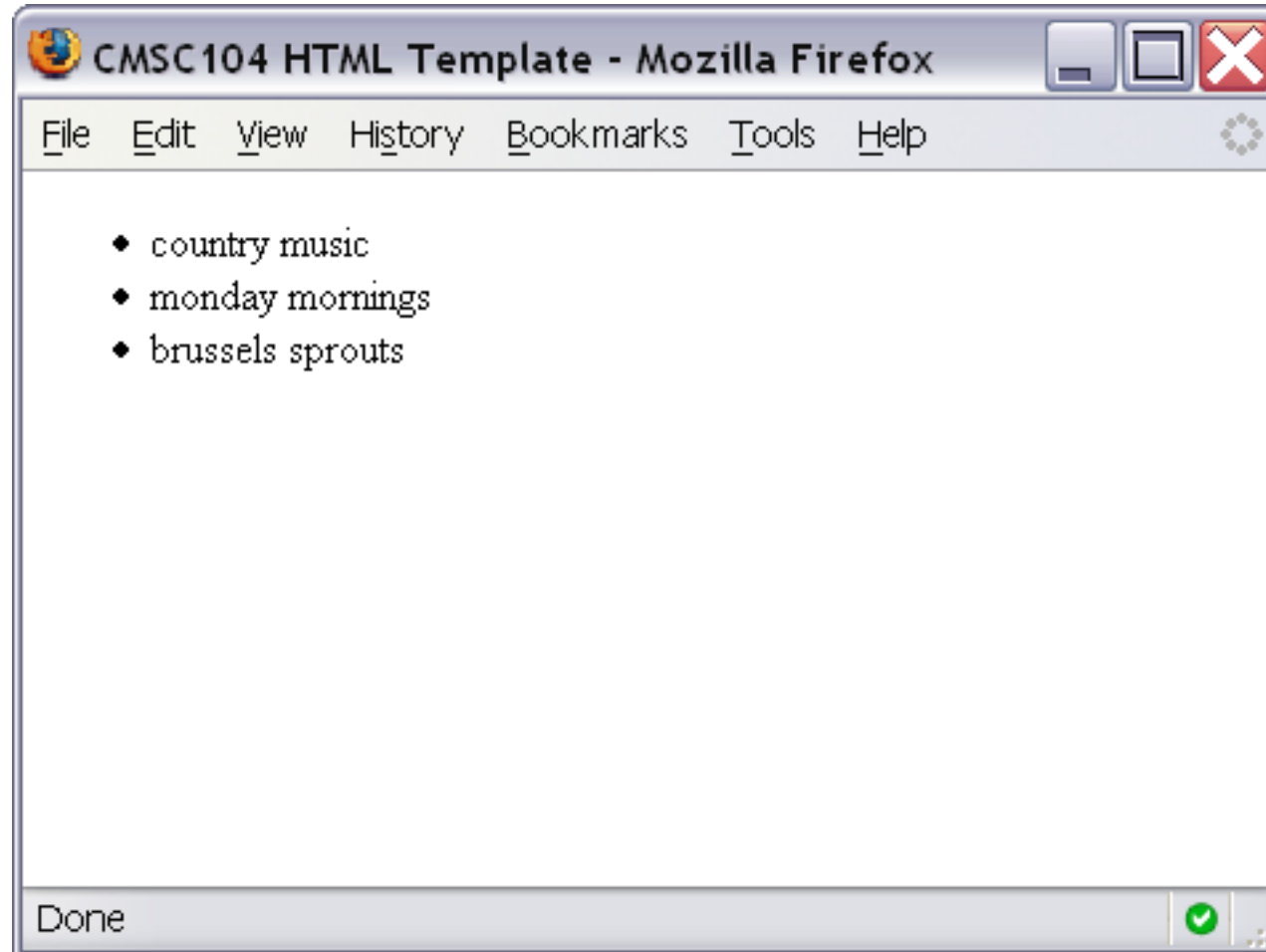
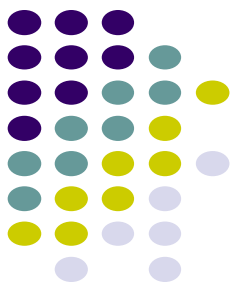


Unordered List Example

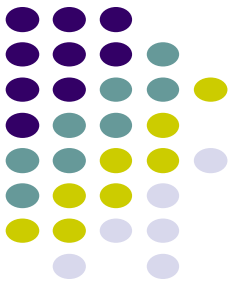


```
<ul>  
  <li>country music</li>  
  <li>monday mornings</li>  
  <li>brussels sprouts</li>  
</ul>
```

Unordered List Screenshot



Link Example



```
<a href="http://www.cs.umbc.edu/104/">CMSC104 Main page</a>
```

Link Screenshot

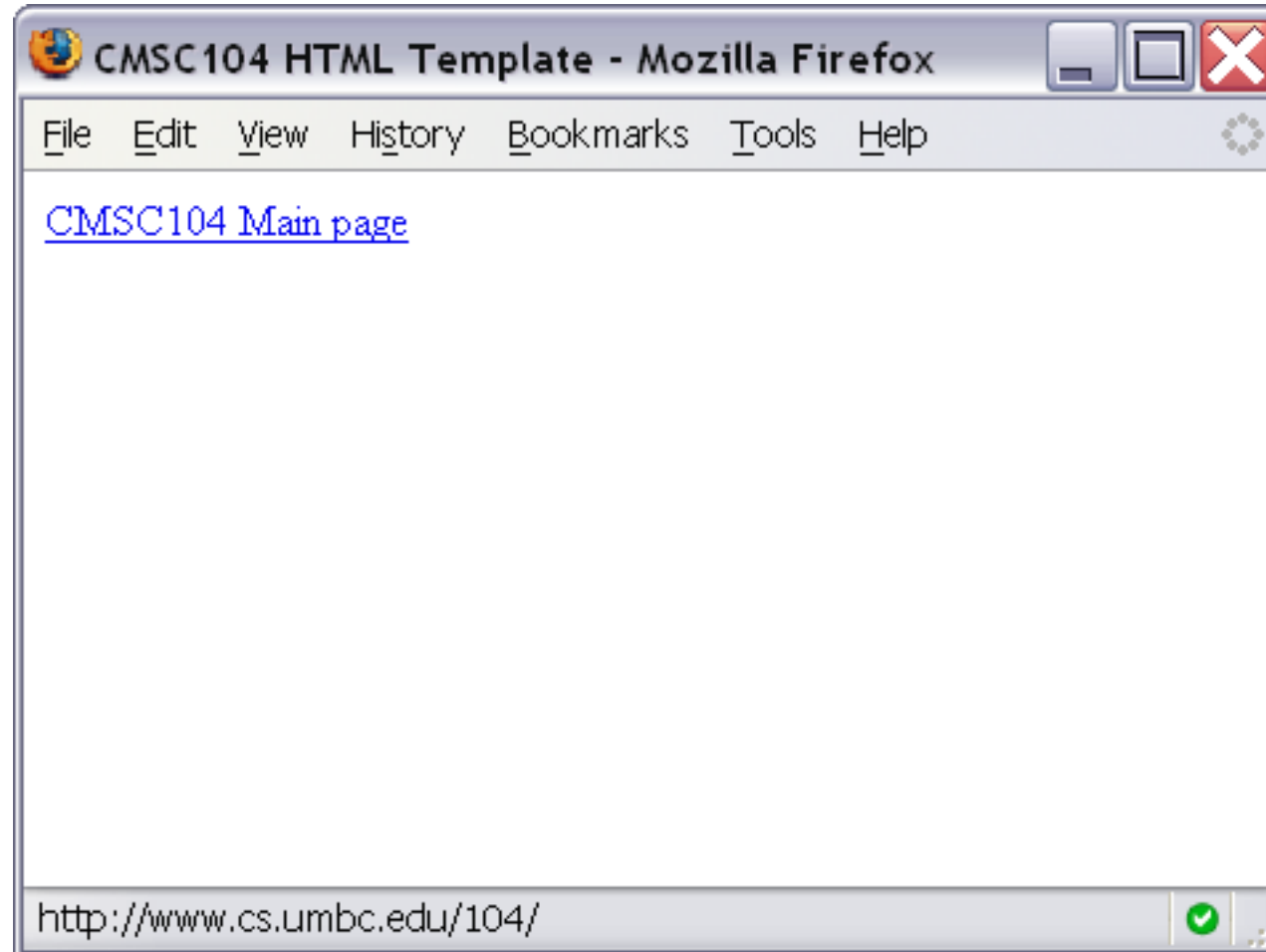
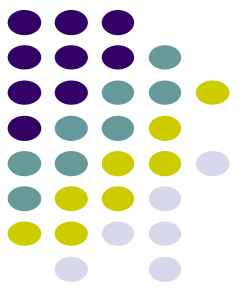
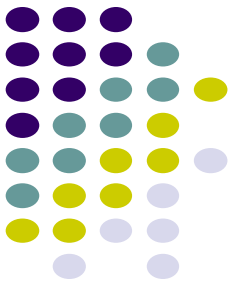


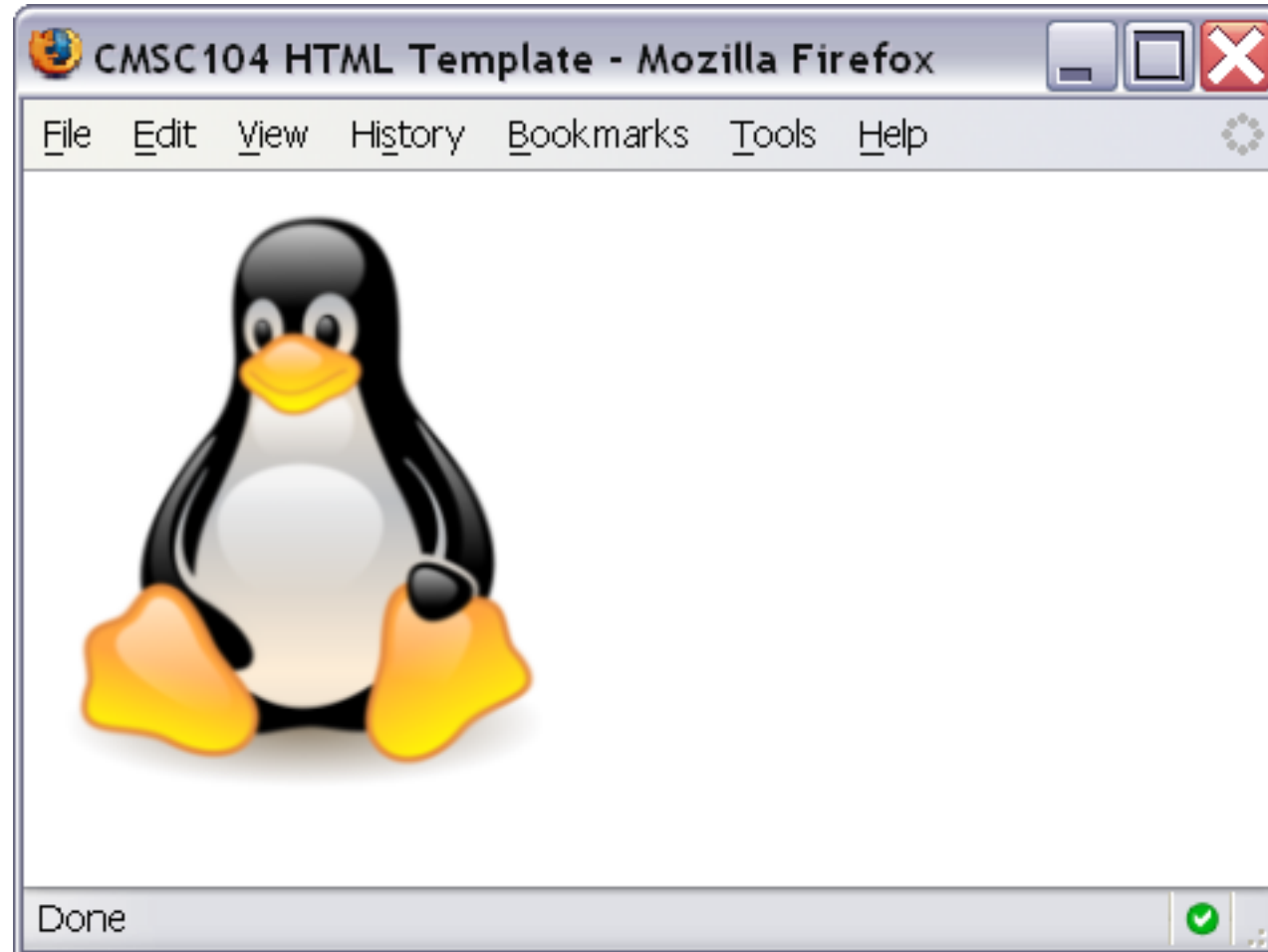
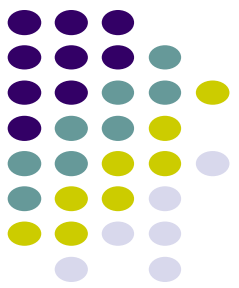
Image Example

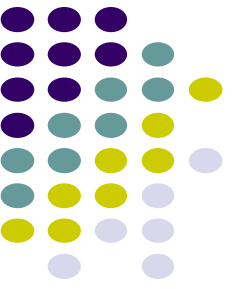


```

```

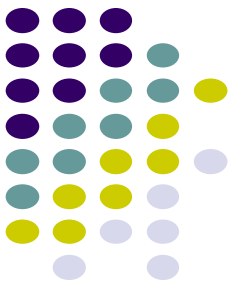
Image Screenshot





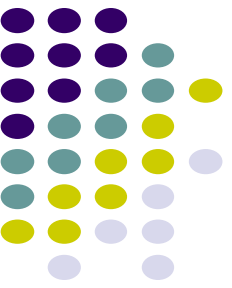
Validating your web pages

- **Introducing the concept of valid pages**
- **Using a doctype**
- **Setting the character set**
- **Meeting the W3C validator**
- **Fixing things when they go wrong**
- **Using HTML Tidy to clean your pages**



Pointers for validation

- All tags have endings.
- Tags can't be overlapped
- Everything's lowercase
- Attributes must be in quotes
- Layout must be separate from markup (css – we shall see later)



Online validator

- You can validate your code using W3C validator at <http://validator.w3.org>
- A validator is actually the front end of a piece of software that checks pages for validity. It looks at your web page's doctype and sees whether the page conforms to the rules of that doctype. If not, it tells you what might have gone wrong.

The W3C Markup Validation Service

validator.w3.org

W3C[®] Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Validate by URI Validate by File Upload Validate by Direct Input

Validate by URI

Validate a document online:

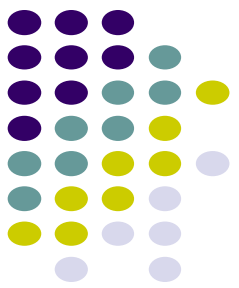
Address:

► More Options

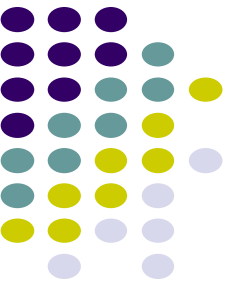
Check

This validator checks the [markup validity](#) of Web documents in HTML, XHTML, SMIL, MathML, etc. If you wish to validate specific content such as [RSS/Atom feeds](#) or [CSS stylesheets](#), [MobileOK content](#), or to [find broken links](#), there are [other validators and tools](#) available. As an alternative you can also try our [non-DTD-based validator](#).

Methods



- **Validate by URI** - this option is used when a page is hosted on a web server. Files stored on local computers can't be checked with this technique.
- **Validate by file upload.** This technique works fine with files you haven't posted to a web server. It works great for pages you write on your computer but that you haven't made visible to the world. This is the most common type of validation for beginners.
- **Validate by direct input.** The validator page has a text box you can simply paste your code into. It works, but I usually prefer to use the other methods because they're easier.



Working Example

- To see an example page that uses all of the tags we discussed today, visit

<http://userpages.umbc.edu/~dblock/lecture6.html>