

Dynamic HTML

Cascading Style Sheets™

(CSS)

# CSS

CSS allows you to specify the style of your page elements (spacing, margins, etc.) separately from the structure of your document (headers, body text, links, etc.). This separation of structure from content allows greater manageability and makes changing the style of your document easier.

- 1) Define how documents should be presented on screens or in print
- 2) Put a structure on formatting
- 3) Defined once and used many times
- 4) Nested styles are permitted for flexibility
- 5) HTML tags can be redefined with the desired presentation
- 6) Custom styles can be defined as needed

# Inline Styles (Local)

## Inline styles

1. Individual element's style declared using the **STYLE** attribute

2. Each CSS property followed by a colon and the value of that attribute

3. Multiple properties separated by semicolons

**<P STYLE = "font-size: 20 pt; color: #0000FF">**

4. Inline styles override any other styles

<p >inline styles </p>

<p style="color: green; font-size:20pt">inline styles </p>

<p style="color: red; font-size:30pt">inline styles </p>

<H1 style="color:blue; font-size:40pt">inline styles </H1>

<DIV style="color: lime; font-size:50pt">inline styles </DIV>



# Font Lengths

there are lots of style properties whose value is specified as some kind of length, including font sizes, indentation, spacing, etc. Lengths are specified using a number followed by a 2 letter code that indicates the units. Some of the common unit codes are:

Unit Code	Units	Examples
em	Overall height of current font	Hello <code>&lt;DIV style="font-size:2em"&gt;World&lt;/DIV&gt;</code> Hello <b>World</b>
px	Pixels	Hello <code>&lt;DIV style="font-size:8px"&gt;World&lt;/DIV&gt;</code> Hello <small>World</small>
in	Inches	Hello <code>&lt;DIV style="font-size:.5in"&gt;World&lt;/DIV&gt;</code> Hello <b>World</b>
pt	Points size (72.27 points/inch)	Hello <code>&lt;SPAN style="font-size:8pt"&gt;World&lt;/SPAN&gt;</code> Hello <small>World</small>

# Font Properties

Property Name	Use	Possible Values (examples)
<b>font-family</b>	Sets the font used	Courier Times monospace sans-serif cursive fantasy
<b>font-size</b>	Sets the size of characters	12pt (12 point size) 200% (twice as big as normal) +5pt (5 points bigger than normal) small, medium, large, larger
<b>font-style</b>	italic vs. normal	italic normal
<b>font-weight</b>	Sets the boldness of characters	normal bold bolder (bolder than current setting) lighter (lighter than current setting)

# Color

Each element of a document has both a foreground and background color. These can be changed with the following properties:

Property Name	Use	Possible Values (examples)
<code>color</code>	Sets the foreground color This sets the text color!	Red, Green, Orchid #CECECE, #0000FF any valid color specifier
<code>background-color</code>	Sets the background color	any valid color specifier
<code>background-image</code>	puts an image behind part of a document	url ( <a href="http://www.yahoo.com/images/blah.gif">http://www.yahoo.com/images/blah.gif</a> ) url(cookie.gif)

# Text Properties

Font properties determine the size, color and style of characters. Text properties determine the spacing and alignment. Here are some of the text properties:

Property Name	Use	Possible Values (examples)
<b>letter-spacing</b>	adds space between characters.	normal 2px (2 pixels)
<b>line-height</b>	space between lines	12pt 120% (1.2 times current value) normal
<b>text-align</b>	where to align the text	left (this is the default) center right

# Box Properties

Box properties control how things are placed within a box by the browser. All HTML elements are placed in a box by the browser to determine where to start and stop drawing them. These (invisible) boxes control the layout of an entire page. Here are some of the box properties:

Property Name	Use	Possible Values (examples)
<b>border-color</b>	controls the color of a border around a (otherwise invisible) box.	any color specifier
<b>border-width</b>	controls the border thickness	any length specifier 1px .25in
<b>margin</b>	space between box and any containing elements	any length specifier 10px Note: also check out margin-left, margin-right, margin-top, margin-bottom

# Relative Length Values

Name	Type of Unit	What It Is	Example
<b>em</b>	EM dash	Width of the letter M for that font (and height)	3em
<b>ex</b>	x-height	Height of the UpperCase letter is the same as of the lowercase x of that font	5ex
<b>px</b>	Pixel	Based on the monitor's resolution	125px

**xx-small x-small small medium large x-large xx-large**

# Absolute Length Values

Name	Type of Unit	What It Is	Example
<b>cm</b>	Centimeters		5.1cm
<b>pc</b>	Picas	Generally used to describe font size. 1pc = 12pt	3pc
<b>mm</b>	Millimeters		25mm
<b>pt</b>	Point	Generally used to describe font size. 36pt = 1/2 inch	12pt
<b>in</b>	Inches	1 inch = 2.54cm	2.25in

```
<P style="font-family:sans-serif;font-weight:bold;font-size:20pt; background-color:lime">
```

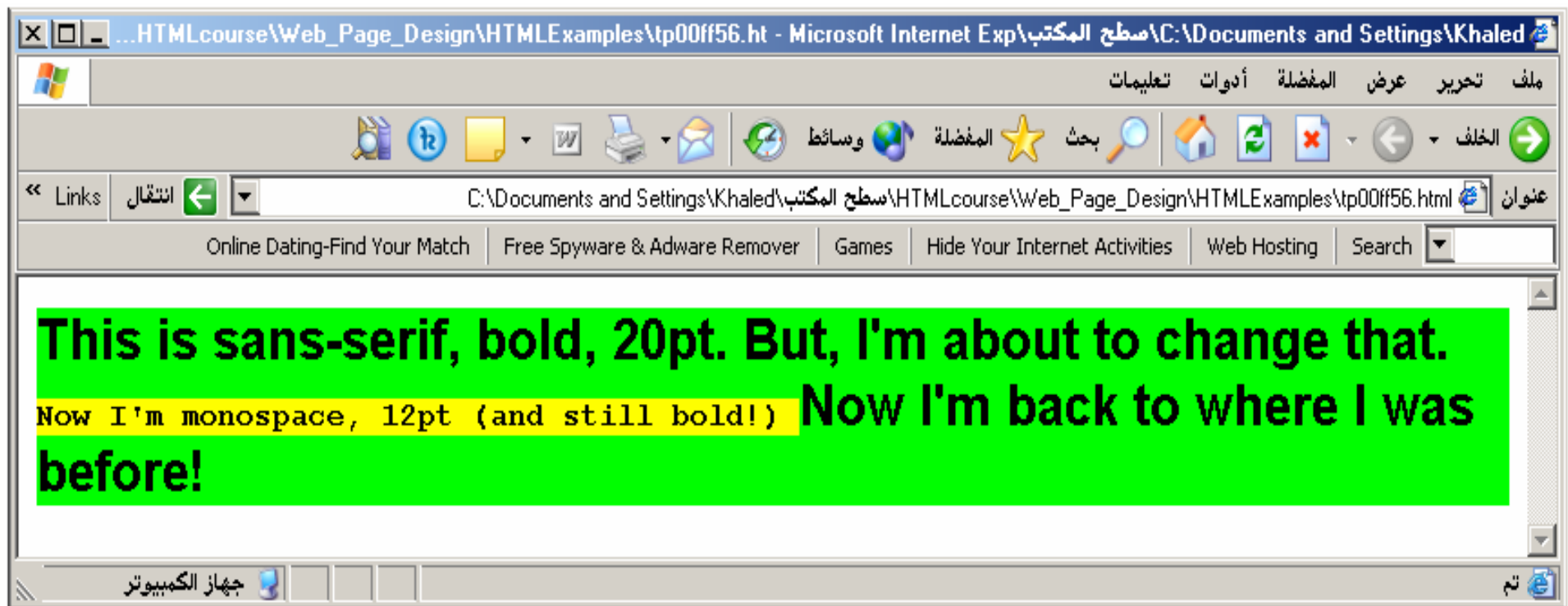
This is sans-serif, bold, 20pt. But, I'm about to change that.

```
<SPAN style="font-family:monospace;font-size:12pt; background-color:yellow">
```

Now I'm monospace, 12pt (and still bold!)

```
</SPAN>
```

```
Now I'm back to where I was before!</P>
```



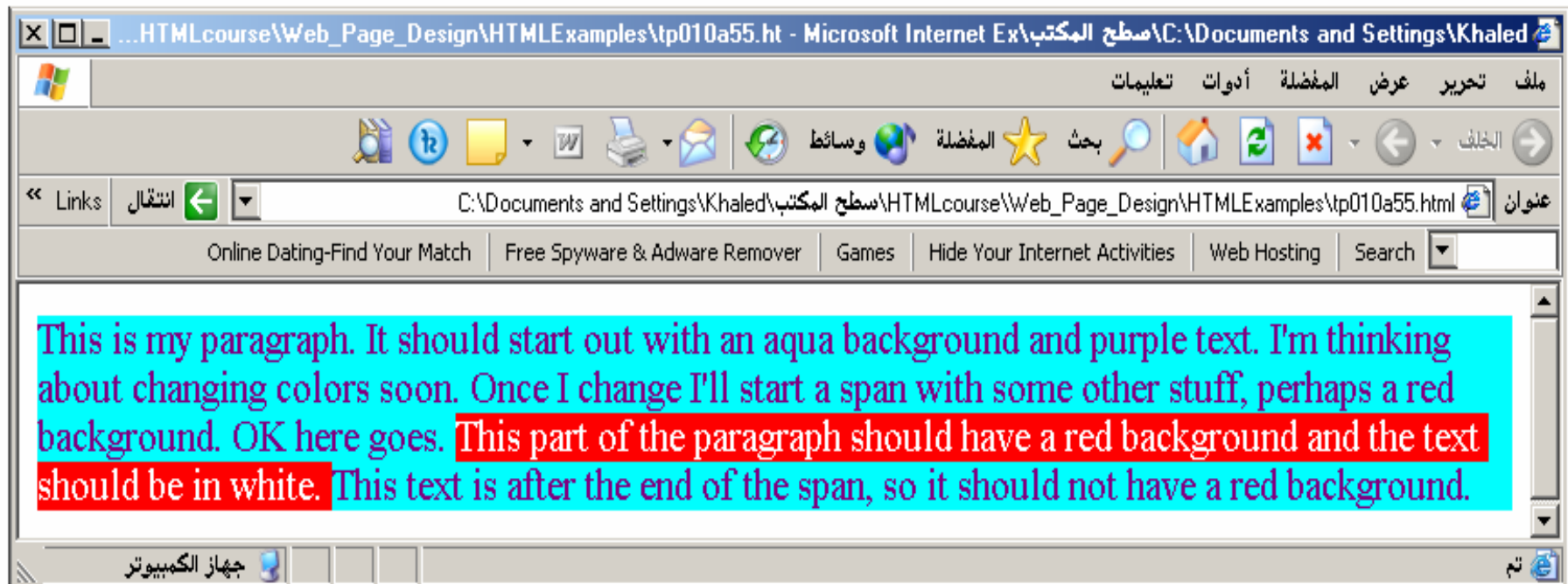
<P style="background-color:aqua;color:purple">

This is my paragraph. It should start out with an aqua background and purple text. I'm thinking about changing colors soon. Once I change I'll start a span with some other stuff, perhaps a red background. OK here goes.

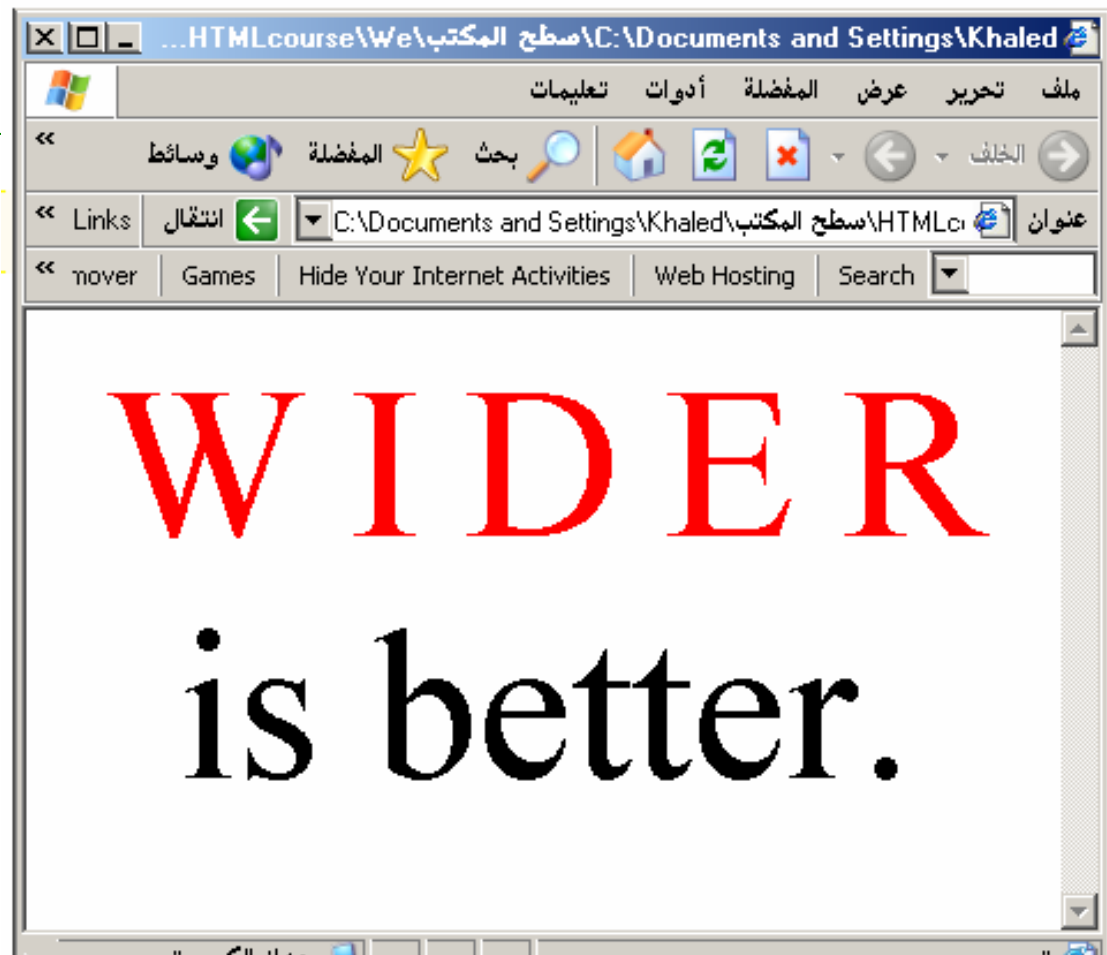
<SPAN style="background-color:red; color:white">

This part of the paragraph should have a red background and the text should be in white.</SPAN>

This text is after the end of the span, so it should not have a red background.</P>

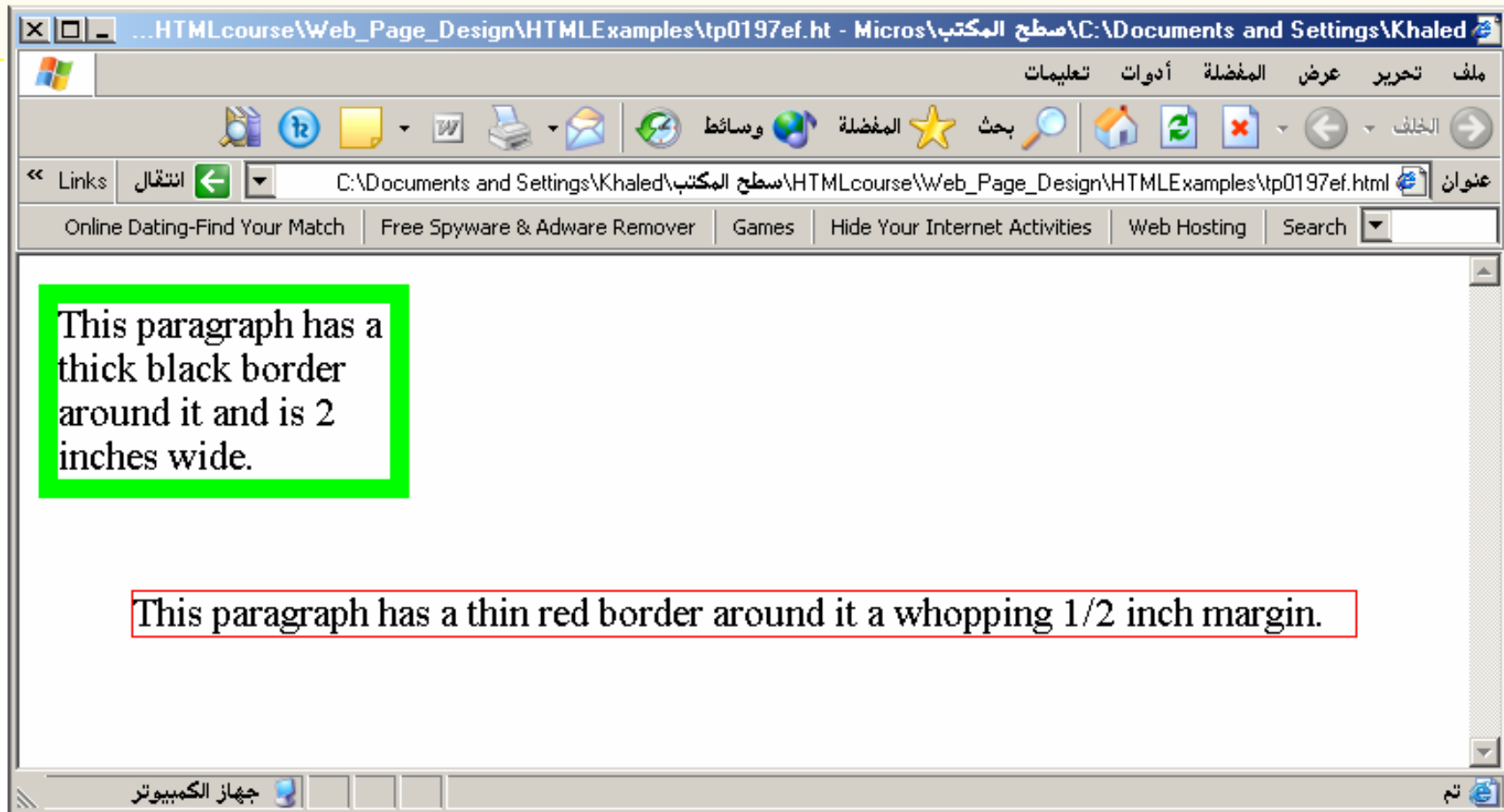


```
<P style="text-align:center;font-size:70pt">  
<SPAN style="letter-spacing:15pt;  
color:rgb(255,0,0)" >  
WIDER</SPAN>  
<BR>  
is better.</P>
```



`<P style="border-style: solid;border-color:lime; border-width:10px; width:2in">` This paragraph has a thick green border around it and is 2 inches wide.`</P>`

`<P style="border-style:solid; border-color:red; border-width:1px;margin: .5in">` This paragraph has a thin red border around it a whopping 1/2 inch margin. `</P>`



```
<P STYLE="margin-left: .5in;
margin-right: 0.5in;          margin-top: 1.75in;
text-align: center;          font-family: comic sans ms;
font-size: 26pt;   color: blue; background-color:yellow;
font-weight:bold;   border-style:solid;
border-color:maroon; border-width:15px">
We all need to <SPAN style="letter-spacing:10pt;font-
family: Arial; background-color:white;color: lime;">LEARN
</SPAN>and to understand how to use
<SPAN style=" font-size=2em;color=white; background-
color:#f70ff0" > CSS </SPAN></p>
```

We all need to **LEARN** and to  
understand how to use **CSS**

# Cursor Types

Cursor Type	
auto	Shows it according to how the viewer has it set (plain)
crosshair	Looks like a cross
default	Makes the cursor stay the same
move	Looks like you should be moving something
hand	The hand you usually see over links
help	A question mark beside the arrow
text	The bar you see when the mouse is over text
wait	The "waiting" hourglass!
n-resize	An arrow- North
s-resize	An arrow- South
e-resize	An arrow- East
w-resize	An arrow- West
ne-resize	An arrow- NorthEast
nw-resize	An arrow- NorthWest
se-resize	An arrow- SouthEast
sw-resize	An arrow- SouthWest

# What will this do?

```
<A HREF="#" style="cursor:hand">A handLink</A><br>
<A HREF="#" style="cursor:crosshair">A Cross Link</A><br>
<A HREF="#" style="cursor:help">A help Link</A><br>
<A HREF="#" style="cursor:wait">A wait Link</A><br>
<A HREF="#" style="cursor:s-resize">A south resize
Link</A><br>
<A HREF="#" style="cursor:e-resize">A east resize Link</A><br>
<A HREF="#" style="cursor:sw-resize">A south west
Link</A><br>
<A HREF="#" style="cursor:se-resize">A south east Link</A><br>
<A HREF="#" style="cursor:text">A text Link</A><br>
<A HREF="#" style="cursor:move">A move Link</A><br>
<A HREF="#" style="cursor: default">A default Link</A><br>
```

# Embedded Styles (Global)

Embedded CSS (*the Intended way*)

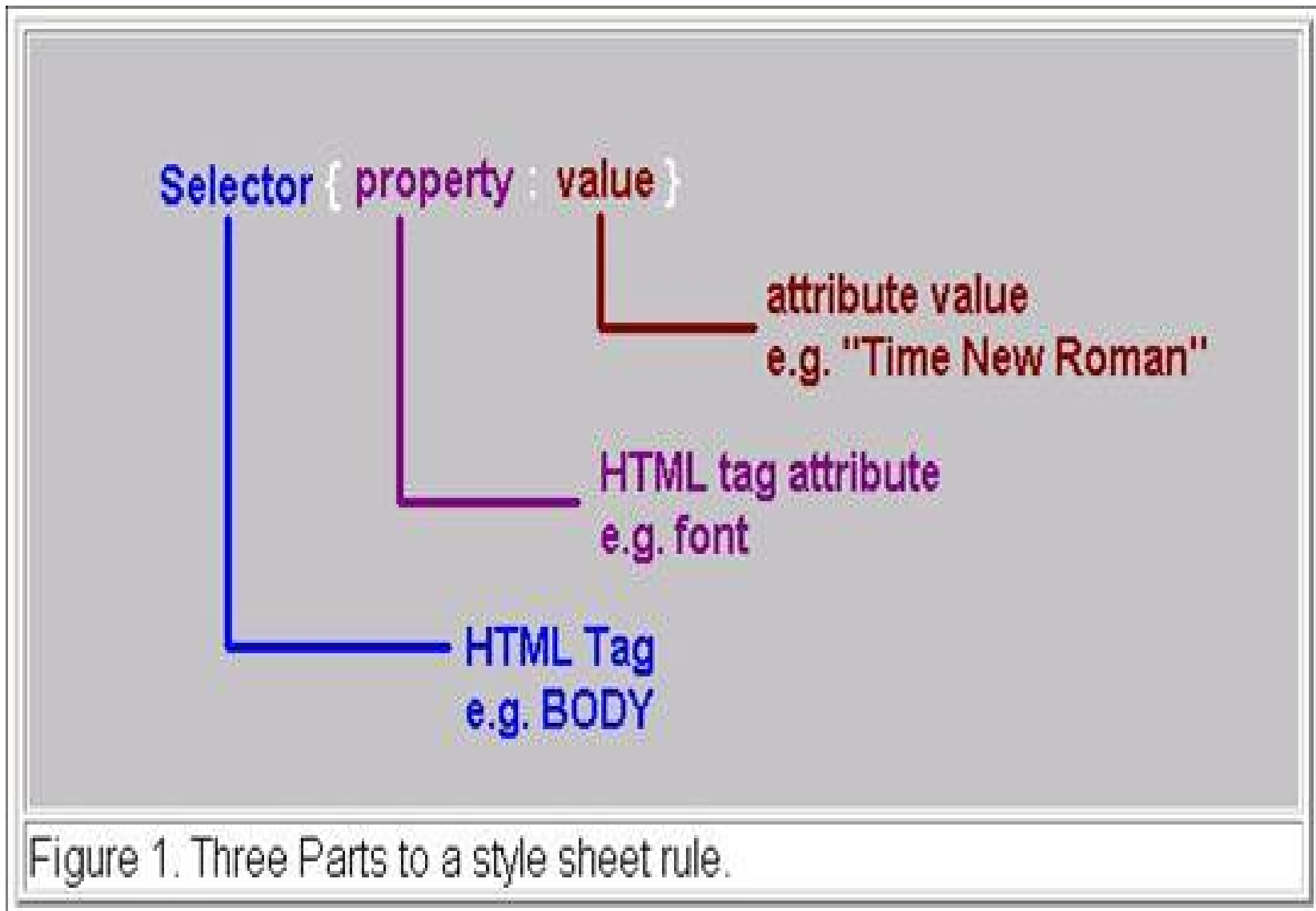
1. Style Rules defined only once
2. Create one place to store all Styles
3. Linked to many Web pages
4. Embedded Styles require less work!
  - Set Background colors or images
  - Set one or more margins
  - Draw Borders in different sizes and colors.
5. Embedded Styles are different sizes and colors.
6. Embedded Styles use Curly Braces.

To embed your **<STYLE>** block.

In the **<HEAD>** after the **<TITLE>**, type:

```
<STYLE TYPE="text/css"> Here I embed my  
styles with joy </STYLE>
```

# Embedded Styling



# What do the following do?

1) **A {text-decoration: none;}**

This tag would allow you to eliminate the underscore on the hypertext links

2) **B {font-family: arial; color: red; font-size: 125%;}**

The Bold tag produces darker and thicker text, now you can add a few more attributes

3) **H1 {font-family: arial black; color: blue; font-size: 200%;}**

It adds Family, Color and Size at 100% larger than the base text for the page.

4) **P {font-family: arial; font-size: 13pt; color: blue; margin-left: 1in; margin-right: 1in; text-align: right; }**

The paragraph tag has both margins by 1 inch.

5) `P B { background: yellow; }`

Now *if, and only if*, the **BOLD** tag occurs within a paragraph, a yellow background will appear behind the text marked between the `<B>` and `</B>` tags.

6) `H1, H2, H3 {text-align: center}`

does exactly the same thing as:

`H1 {text-align: center}`

`H2 {text-align: center}`

`H3 {text-align: center}`

7) `OL LI {list-style: upper-alpha}`

`OL OL LI {list-style: upper-roman}`

`OL OL OL LI {list-style: lower-alpha}`

`OL OL OL OL LI {list-style: decimal}`

## 8) P UL STRONG {color: blue}

and use the sequence `<p><ul><strong>` in the document, the second, more specific rule applies, coloring the contents of the `<strong>` tag blue.

## 9) .italic {font-style: italic}

creates a generic class named italic. To use it, simply include its name with the class attribute. So, for instance, use

`<p class=italic>` or `<h1 class= italic>` to create an italic paragraph or header.

## 10) H1 #blue { color : rgb(0,0,255)}

`#yellow { color : yellow}`

Within the document, we use `<h1 id=blue>` to create a blue heading, or add `id=yellow` to almost any tag to turn it yellow.

## 11) table {color:blue} // tr {color:red} // td {color:green}

## 12) Drop down list

```
select {  
    border: thin dotted #FF0000;  
    font-family: "Comic Sans MS";  
    background-color: #FFCCCC;}
```

## 13) Text Field / Button

```
input {  
    background-color: #FFFFFFCC;  
    border: thin dotted #FF9900;  
    font-family: "Comic Sans MS";  
    background-image: url(picture.jpg);}
```

## 14) TextArea

```
textArea{  
    background-color: #FFFFFFCC;  
    font-family: "Comic Sans MS";  
    background-image: url(picture.gif);  
}
```

<STYLE>

H1 {color:blue; font-size:1.5cm}

P {color: green; font-size: 24pt;}

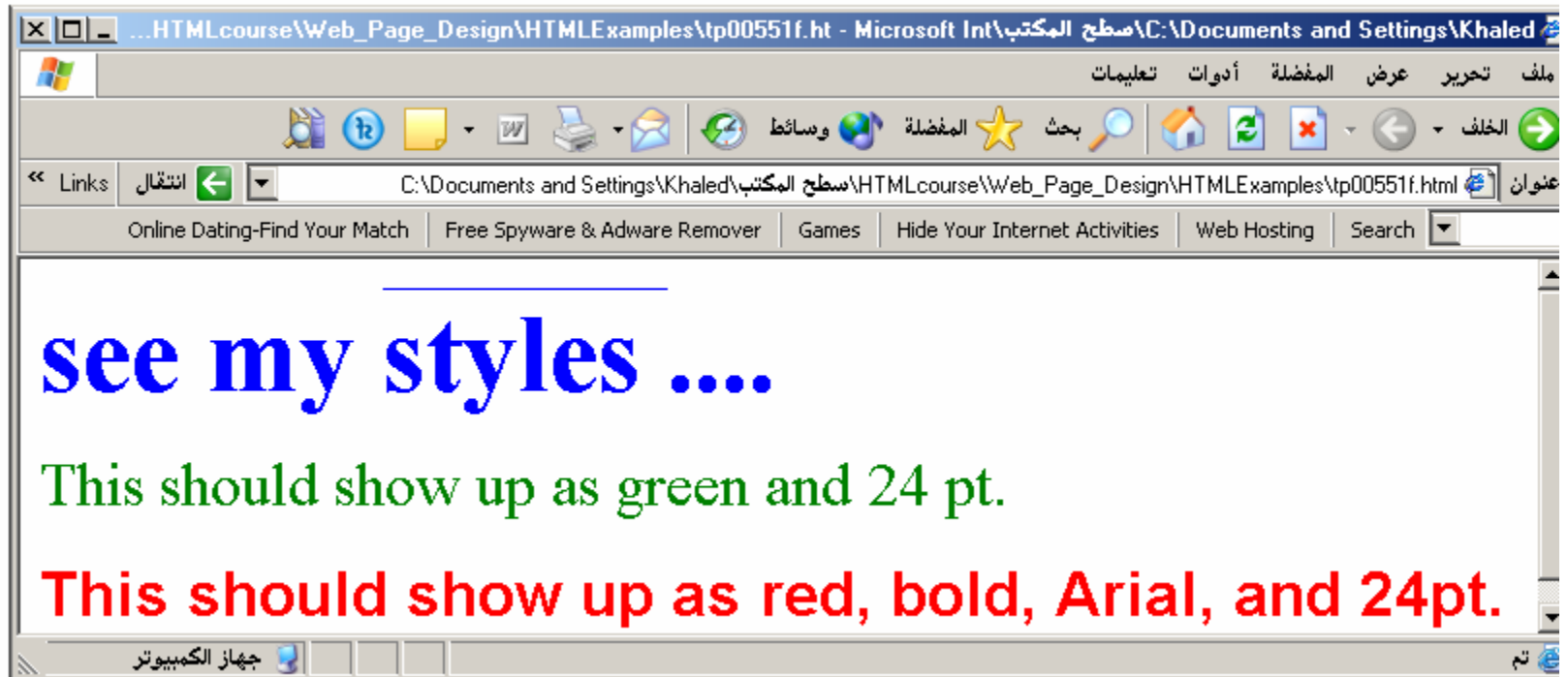
P.x {color: red;font-family:Arial;font-weight:bold }

</STYLE>

<H1> see my <span style="text-decoration:overline">  
styles</span>.... </H1>

<P>This should show up as green and 24 pt.</p>

<P CLASS="x"> This should show up as red, bold, Arial, and  
24pt.</p>



```
<style>
body {
    font-family: "Comic Sans MS";
    font-size: x-large;    font-style: italic;
    font-weight: bold;    text-transform: uppercase;
    color: #0000ff;
    text-decoration: underline overline line-through;
} </style>
```

```
<body> <pre>
```

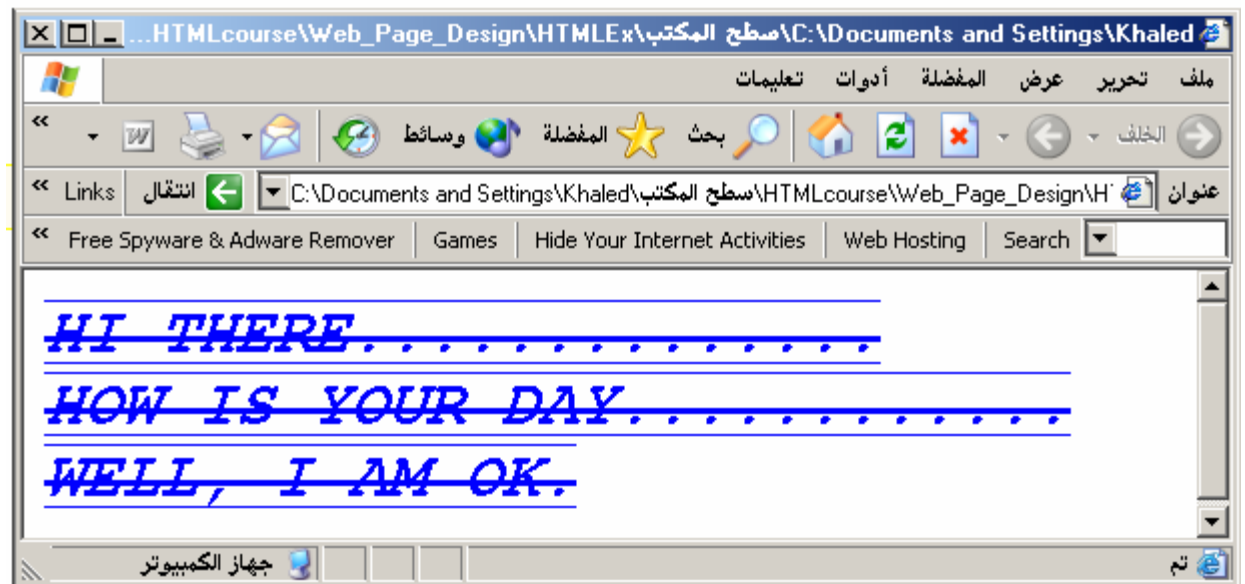
Hi there.....

how is your day.....

Well, I am ok.

```
</pre>
```

```
</body>
```



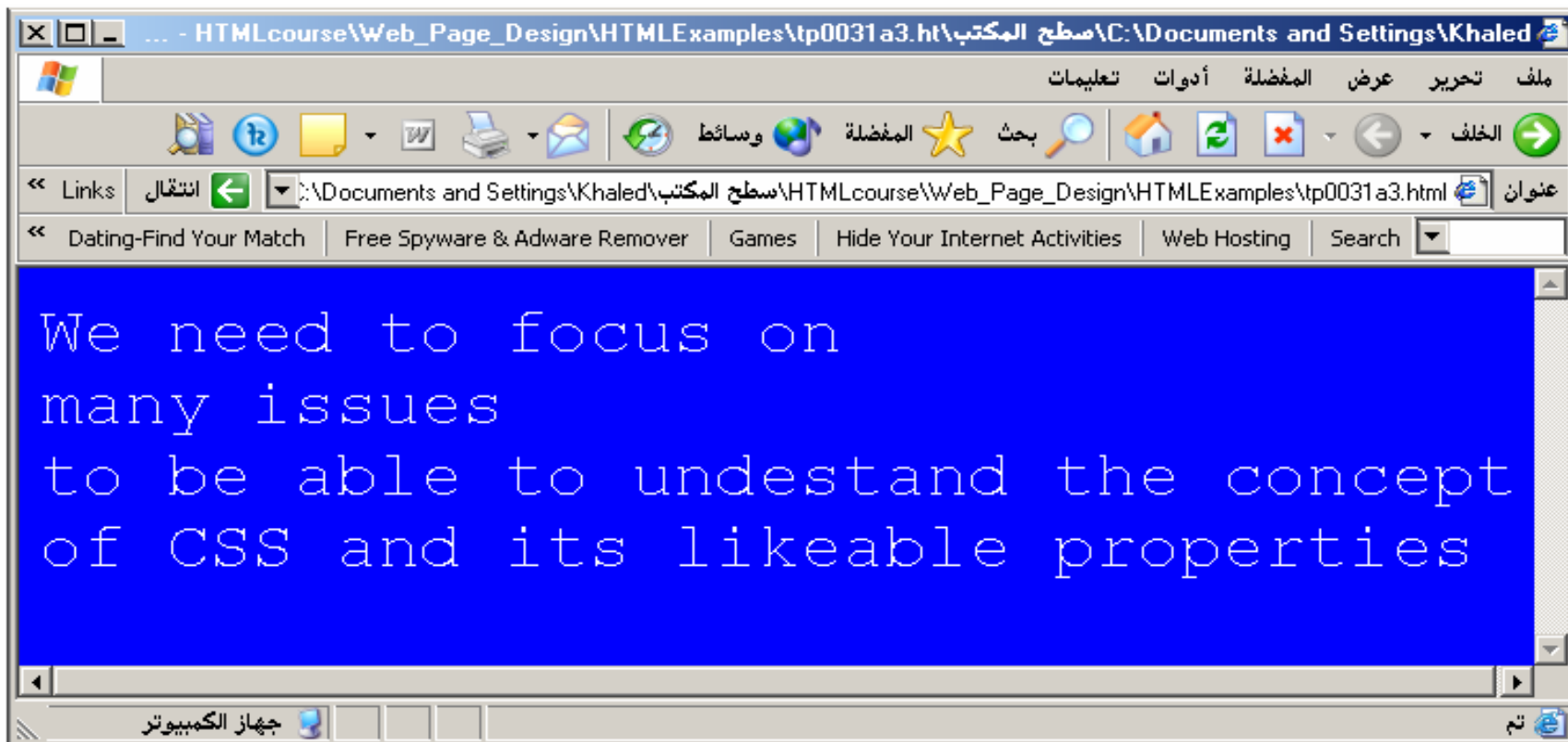
```
<style type="text/css">
```

```
  body { color: white; background: blue;font-size:1.5em }  
</style>
```

```
<body>
```

```
<pre>We need to focus on  
many issues
```

```
to be able to understand the concept  
of CSS and its likeable properties</pre></BODY>
```



<style>

```
input { background-color: #995555;  
        border: thin double dotted #FF0000;  
        font-family: "Comic Sans MS";  
        background-image: url(jordan.gif);  
        width:80px; height=60px;  
        font-weight:bold; font-size:1em}</style>
```

```
<form><input type="button" value="ClickMe"  
name="buttonX" onClick="alert('Hi there');"></form>
```



**<style>**

**select {font-family:Arial; background-color: #FFAA00;  
width:200px; font-size:2em; color:#00bb22; }**

**</style>**

**<form>**

**select size=3>**

**<option>F5**

**<option>F15**

**<option>F16**

**<option>C5**

**<option>C141**

**<option>Cobra**

**</select></form>**



```
<style>
```

```
p {color:rgb(45,100,255); font-size:20pt}
```

```
TEXTAREA {width: 500px; height: 400px;
```

```
color:#f8f888;background-color:blue;
```

```
font-size:30pt;font-weight:bold; margin-left: 150px;}
```

```
input{color:red; font-size:25pt;}
```

```
</style>
```

```
<script>function change()
```

```
{myForm.txtArea.value=myForm.txt.value;}
```

```
</script>
```

```
<form name="myForm">
```

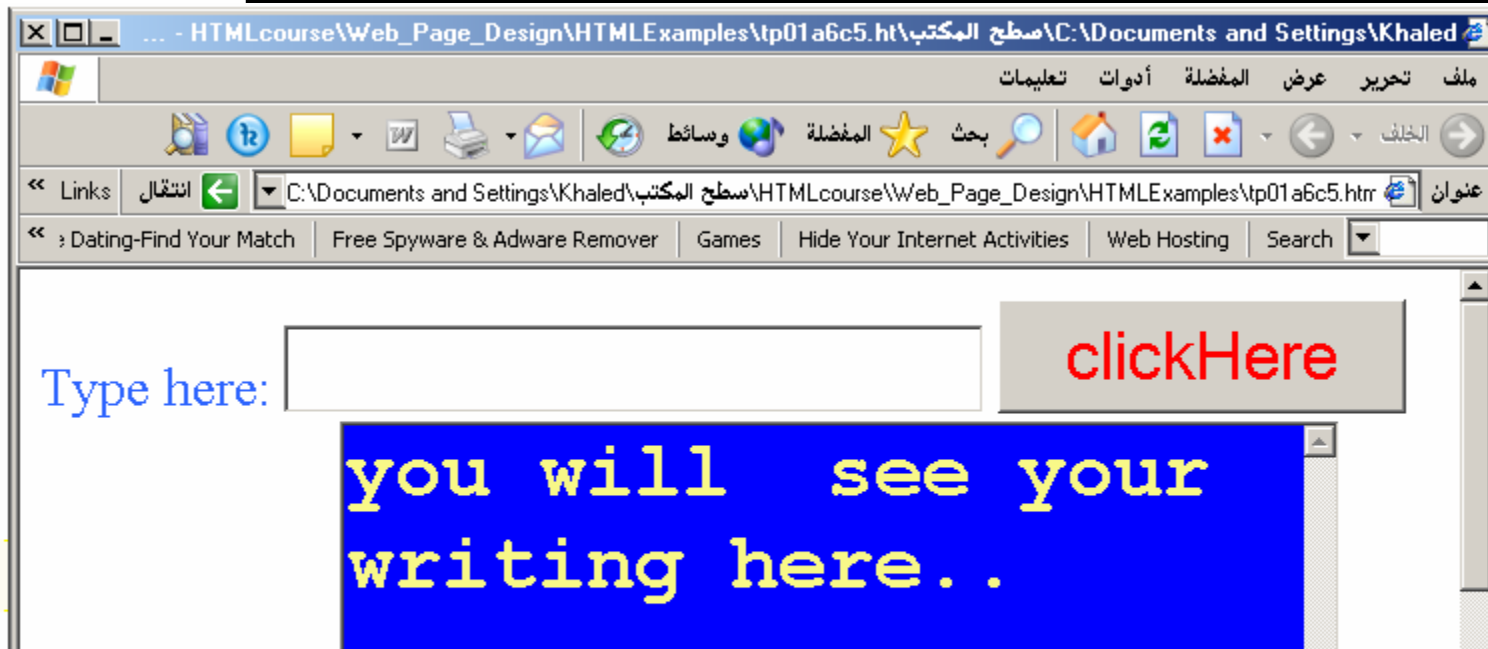
```
<p>Type here: <input type="text" name="txt" size=20>
```

```
<input type="button" name="theButton" value="clickHere"  
onClick="change();">
```

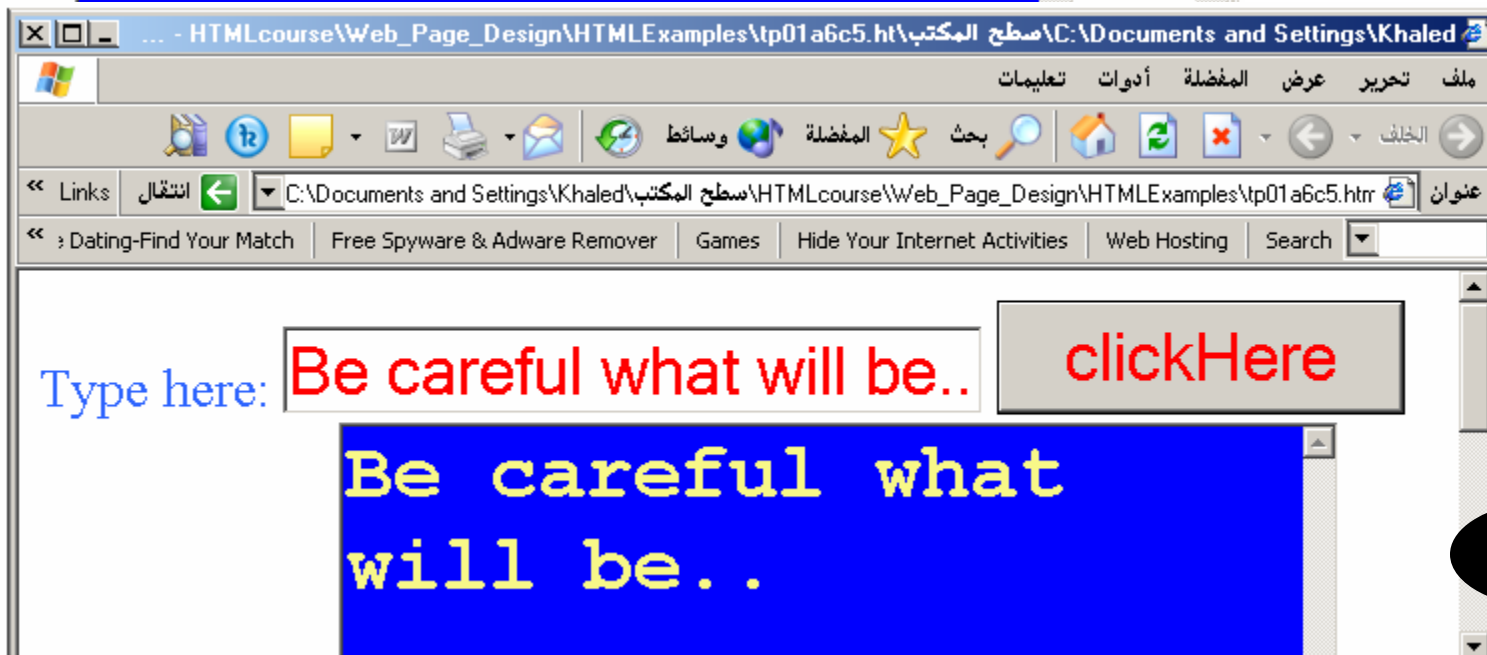
```
<TEXTAREA name="txtArea">you will see your writing  
here..</TEXTAREA>
```

```
</form>
```

# output



جهاز الكمبيوتر



```
<style>
```

```
P {font-size:2.3em;color=forestgreen;margin-left:30px;  
margin-right:240px; background-color:yellow}
```

```
P.float {float: right; font-weight: bold; width: 200px;  
padding: 20px; color:white;background-color: navy;  
margin-right:5px;}
```

```
</style>
```

```
<P CLASS="float"> <IMG SRC="jordan.gif" height=100  
width=100> CSS lets you flow text around graphics
```

```
</p><p>
```

**CSS lets you flow text around graphics**

**or text plus text around tables. It uses a Float property**

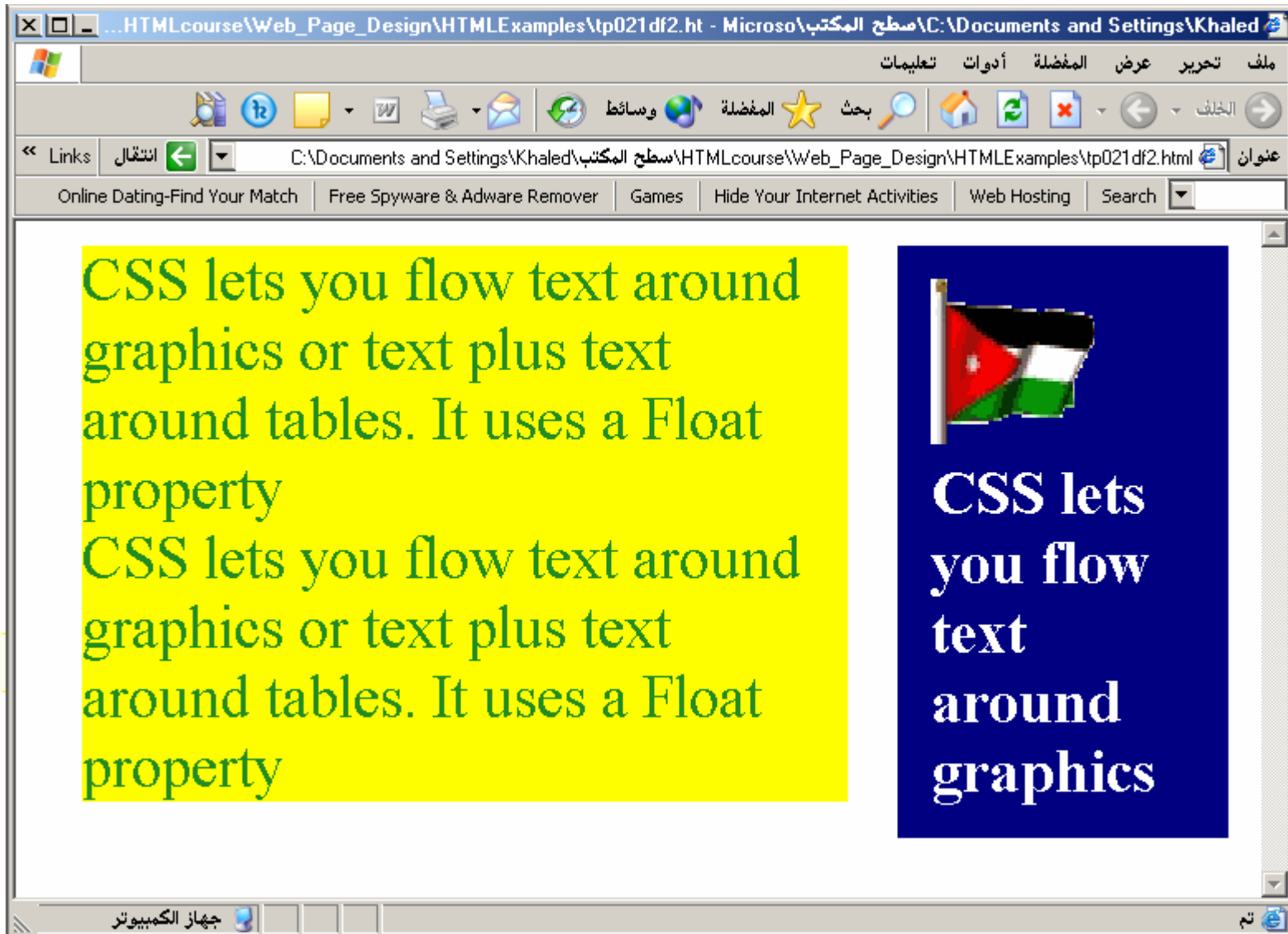
```
<br>
```

**CSS lets you flow text around graphics**

**or text plus text around tables. It uses a Float property**

```
</p>
```

# output



<style>

**B {color:red;}**

**B B {color:blue;}**

**B.foo {color:green;}**

**B B.foo {color:yellow;}**

**B.bar {color:maroon;}**

</style>

<b class=foo>Hey!</b>

<b>Wow!!

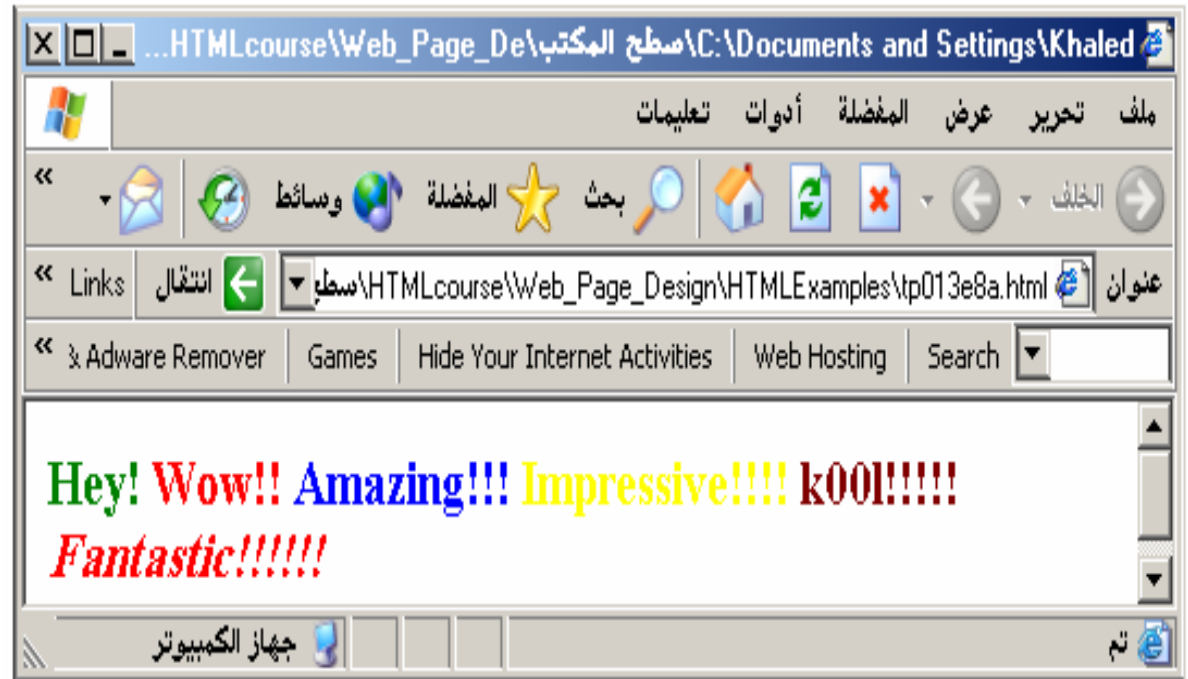
<b>Amazing!!!</b>

<b class=foo>Impressive!!!!</b>

<b class=bar>k00l!!!!!!</b>

<i>Fantastic!!!!!!</i>

</b>



# Positioning Elements

**CSS Positioning (CSS-P) is more accurate than either graphics or tables and the results are displayed much faster. CSS introduces the position property and a capability called *absolute positioning*, which gives us greater control over how documents are displayed.**

**In other words, CSS-P allows us to position elements on the screen either exactly where we want them or in relation to other elements on the screen.**

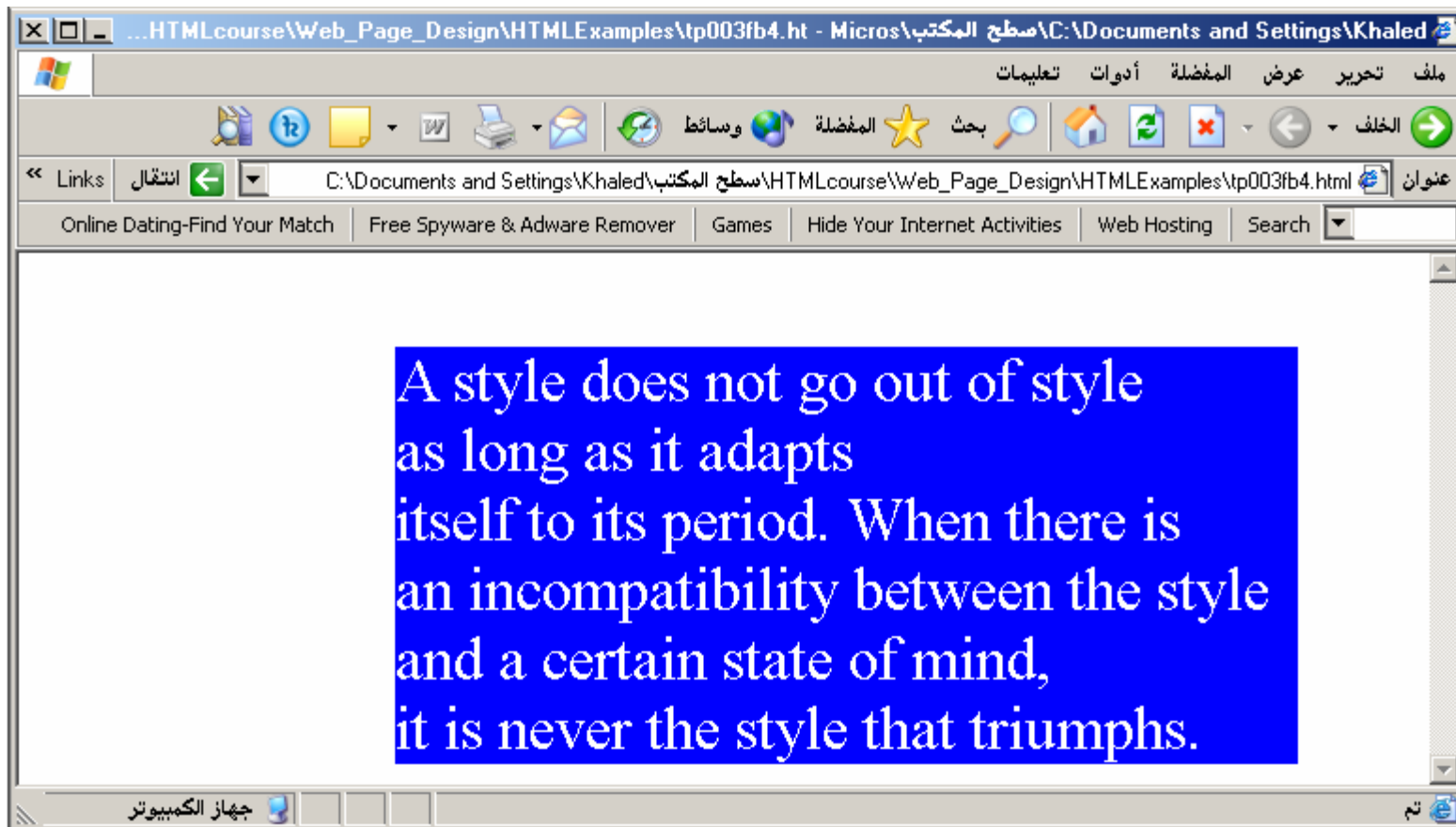
```
<IMG SRC="jordan.gif" width=75 height=75
STYLE="position:absolute; top:50px;left:50px">
<IMG SRC="jordan.gif" width=75 height=75
STYLE="position:absolute; top:100px;left:200px">
```

**Left:** defines the amount of space between the element and the left edge of the browser window.

**Top** defines the space between the element and the top of the window.



**<STYLE> DIV { position: absolute; left: 200px; top: 50px; height: 250px; width=280; background-color=blue; color:white;font-size:25pt} </STYLE><DIV>A style does not go out of style<br>as long as it adapts<br>itself to its period. When there is <br>an incompatibility between the style <br>and a certain state of mind, <br>it is never the style that triumphs. </DIV>**



**<STYLE>**

**DIV { position: absolute; left: 200px; top: 50px; height: 250px; width=200; background-color=blue; color:white;}**

**P { position: absolute; left: 250px; top: 200px; height: 250px width=200; background-color=black; color:yellow; }**

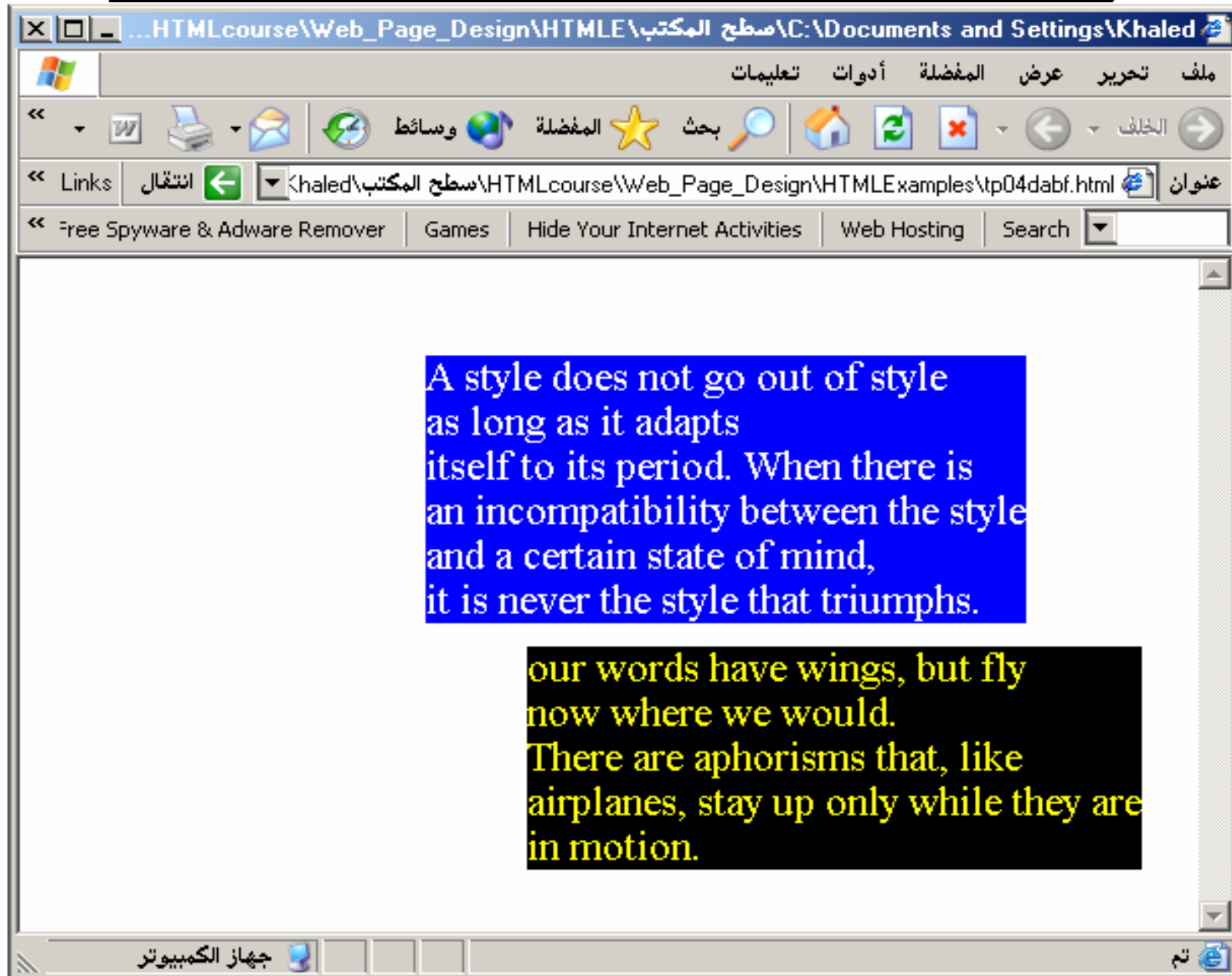
**</STYLE>**

**<DIV>A style does not go out of style<br>as long as it adapts<br>itself to its period. When there is <br>an incompatibility between the style <br>and a certain state of mind, <br>it is never the style that triumphs.**

**</DIV>**

**<p>our words have wings, but fly<br>now where we would. <br>There are aphorisms that, like <br>airplanes, stay up only while they are in motion. </p>**

# output



**<STYLE>**

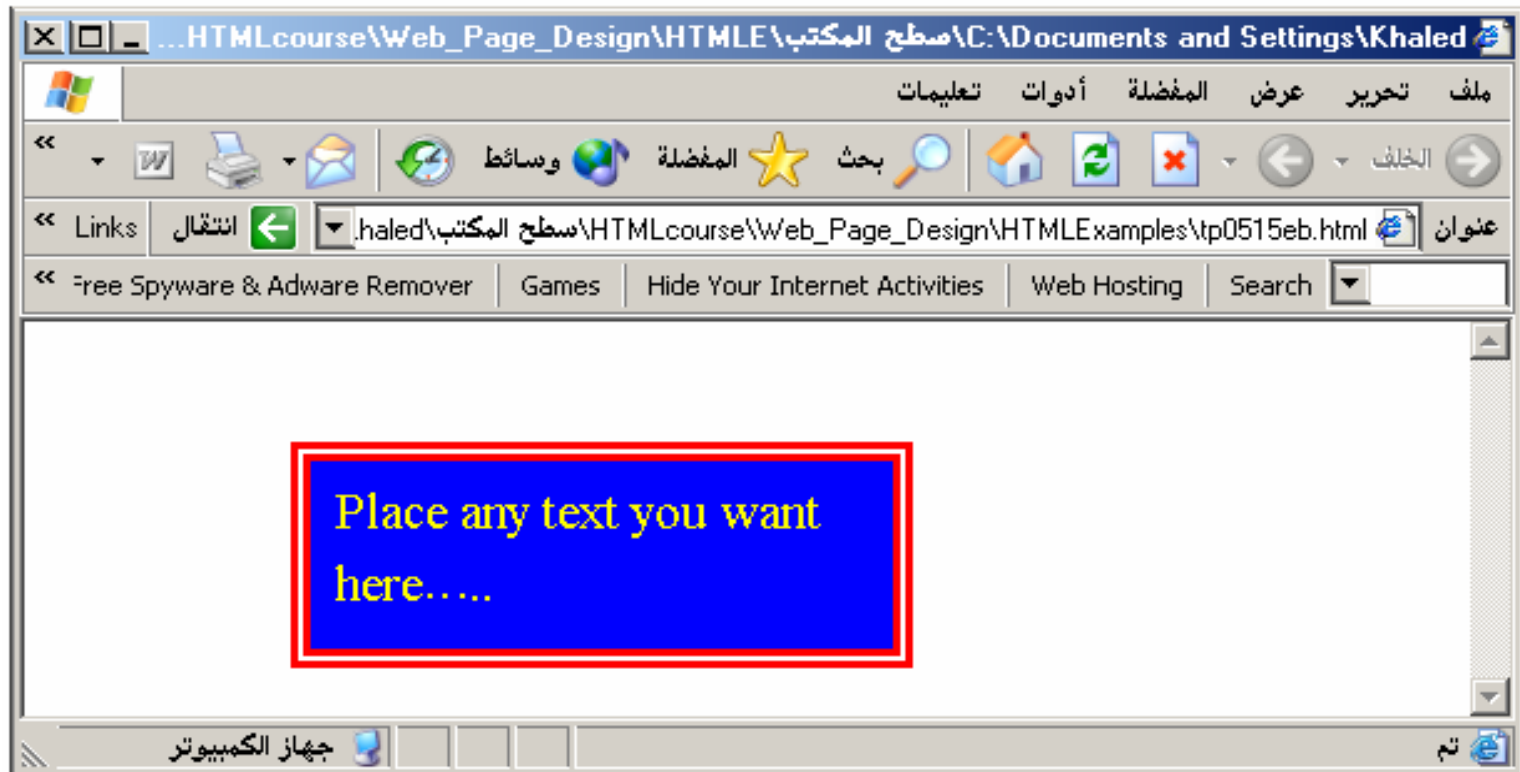
**#pos1 {position: absolute; left: 5em; top: 50px;}**

**</STYLE>**

**<p ID="pos1" STYLE="width: 250px; border: red 8px double; background: blue; color:yellow; padding: 10px;">**

**Place any text you want here.....**

**</p>**



# Backgrounds

```
<style>body {  
background-color: #00FFFF;  
background-image: url(jordan.gif);  
background-repeat: no-repeat;  
background-position: top right;  
background-attachment: fixed;}  
body { color:red; font-size:1.4 em}  
</style>  
<BODY>  
This is Jordan's Flag  
</BODY>
```



# Element Dimensions

```
<style>DIV {background-color: #00FFFF;  
margin-bottom: 0.4em;padding:0.5cm}</style>
```

```
<BODY>
```

```
<DIV STYLE="width: 40%">Browsers that don't  
support images will still be able to read text that has  
been added as CSS. </DIV>
```

```
<DIV STYLE="width:75%; text-align:center">
```

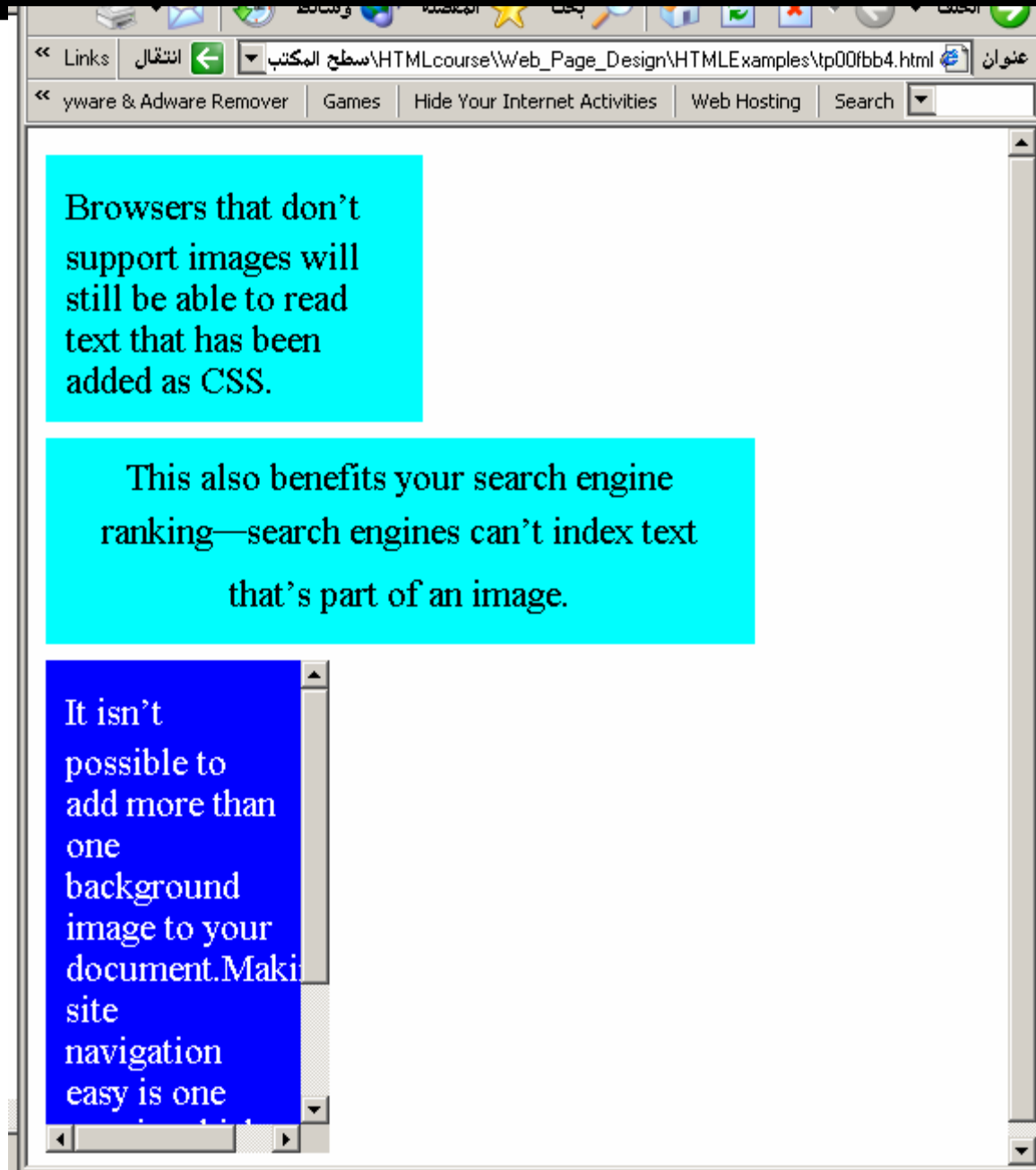
```
This also benefits your search engine ranking—search  
engines can't index text that's part of an image. </DIV>
```

```
<DIV STYLE="width: 20%; height:50%;background-  
color: #0000FF;color:white; overflow:scroll">
```

```
It isn't possible to add more than one background  
image to your document. Making site navigation easy  
is one area in which CSS really comes into its own.
```

```
</DIV></BODY>
```

# output



```
<style>
```

```
#pos1 {position: relative; left: 5em; top: 50px;}
```

```
</style>
```

```
<p ID="pos1" STYLE="color:red;width:200px;  
height:100px; border:green 10px double; background-  
color:pink; font-size:25pt;padding:15px">
```

```
Hi there..... </p>
```

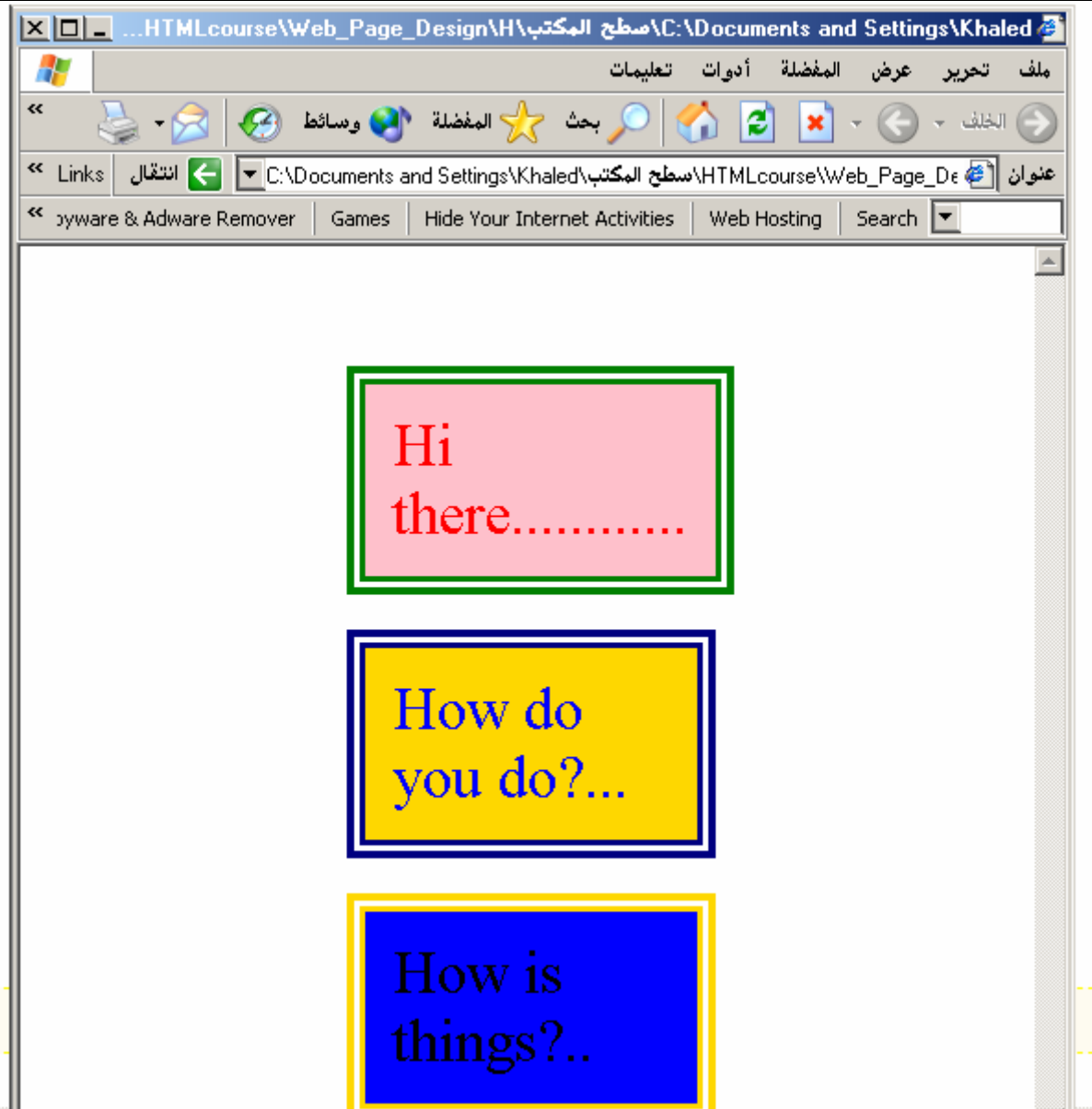
```
<p ID="pos1" STYLE="color:blue;width:200px;  
height:100px; border:navy 10px double; background-  
color:gold;font-size:25pt; padding:15px">
```

```
How do you do?... </p>
```

```
<p ID="pos1" STYLE="color:black; width:200px;  
height:100px; border:gold 10px double; background-  
color:blue; font-size:25pt; padding:15px">
```

```
How is things?.. </p>
```

# output



```
<style>
```

```
#pos1 {position: absolute; left:50px; top: 50px;z-index:1}
```

```
#pos2 {position: absolute; left:80px; top: 80px;z-index:2}
```

```
#pos3 {position: absolute; left:110px; top: 110px;z-index:3}
```

```
#pos4 {position: absolute; left:140px; top: 140px;z-index:4}
```

```
</style>
```

```
<IMG SRC="cat94.JPG" width=100 height=100 ID="pos1">
```

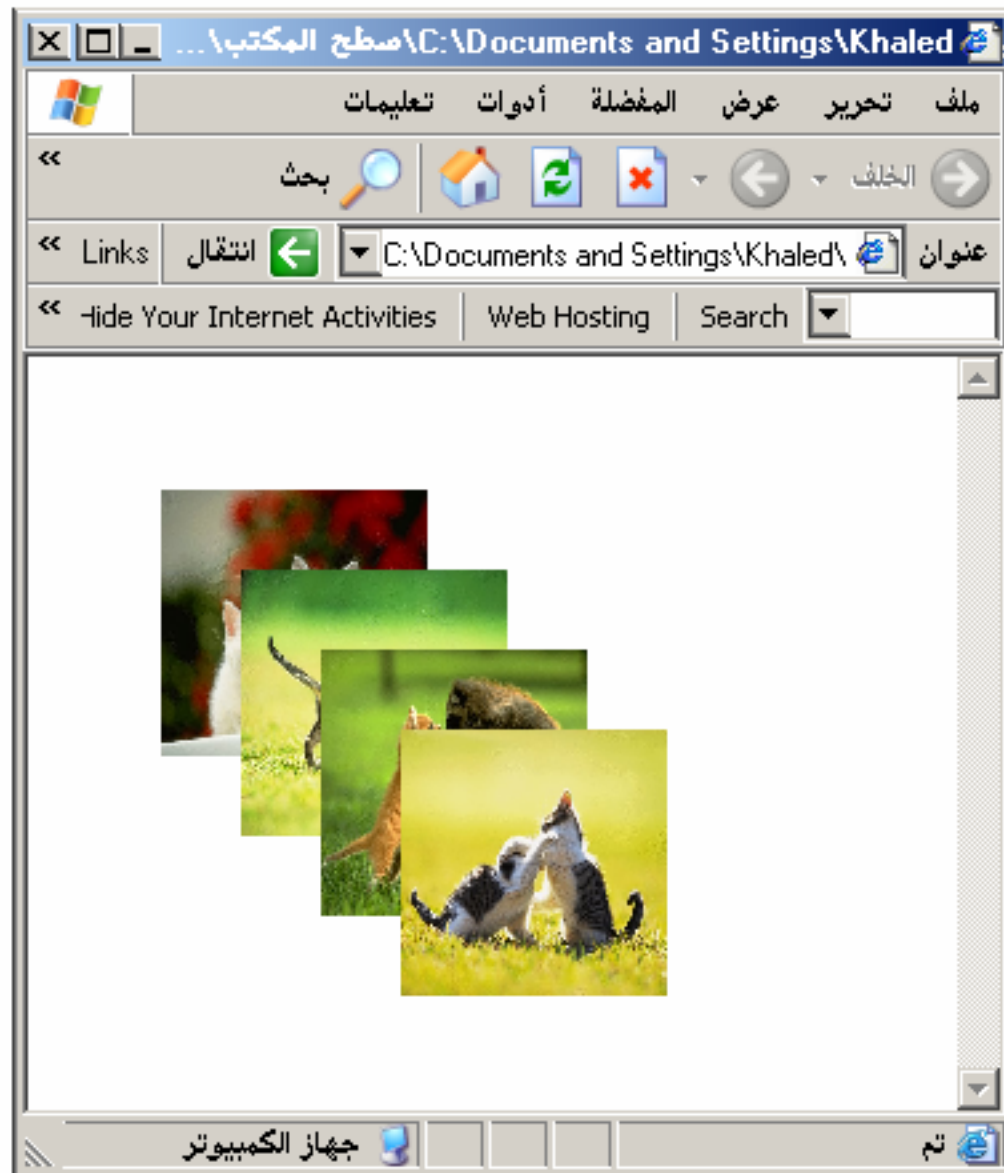
```
<IMG SRC="cat76.JPG" width=100 height=100 ID="pos2">
```

```
<IMG SRC="cat82.JPG" width=100 height=100 ID="pos3">
```

```
<IMG SRC="cat86.JPG" width=100 height=100 ID="pos4">
```

The **z-index** attribute allows you to properly layer overlapping elements. Elements that have higher **z-index** values are displayed in front of elements with lower **z-index** values.

# output



```
<style TYPE="text/css">
```

```
BODY{ background-color:#CCFFDD}
```

```
DIV {text-align:center; margin-bottom: 0.3em;  
width: 30%; position: relative;  
left:25%; padding:0.3em}
```

```
</style>
```

```
<BODY>
```

```
<DIV STYLE="border-style:solid"> Solid border</DIV>
```

```
<DIV STYLE="border-style:double"> Double border</DIV>
```

```
<DIV STYLE="border-style:groove"> Groove border</DIV>
```

```
<DIV STYLE="border-style:ridge"> Ridge border</DIV>
```

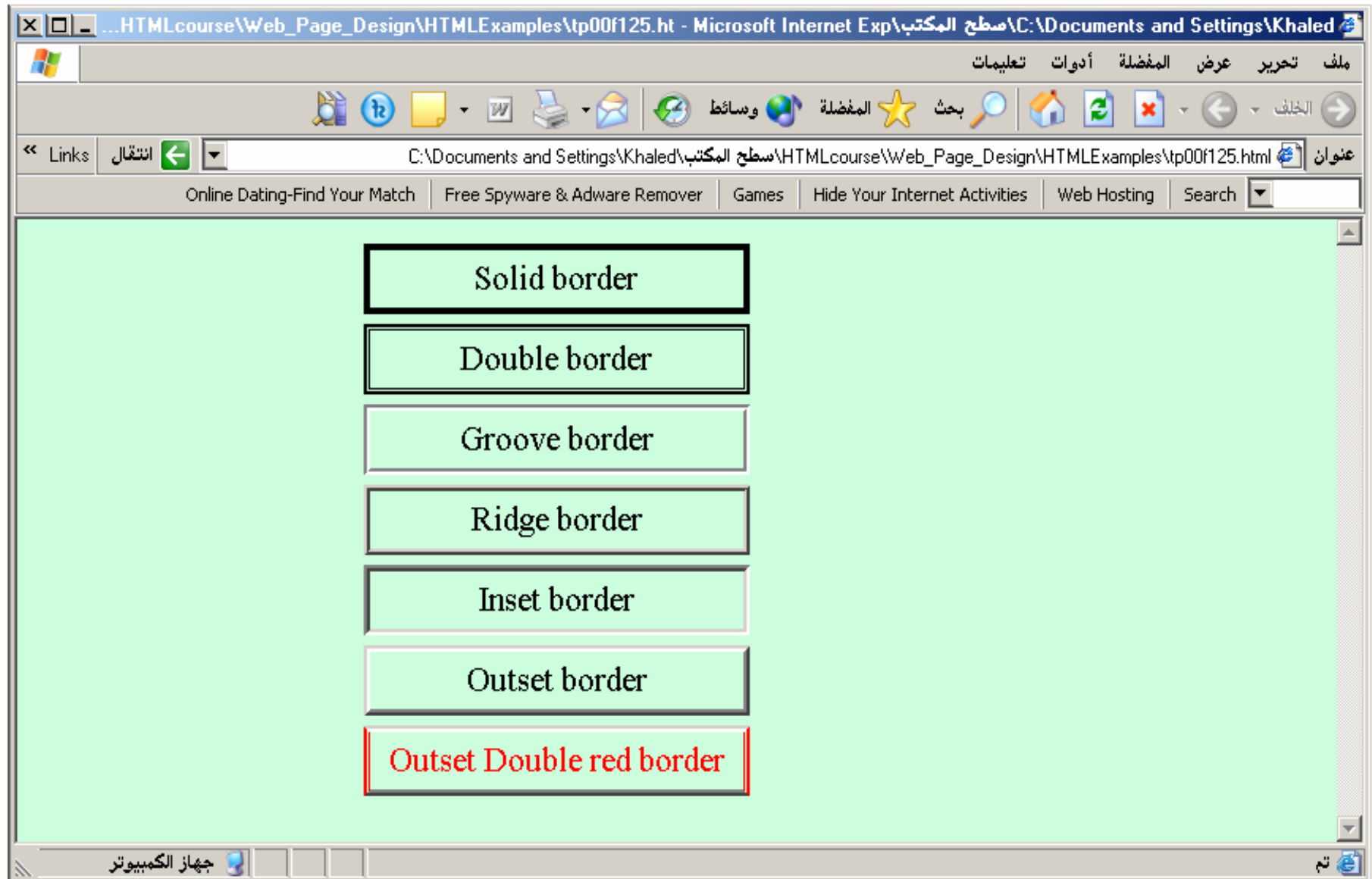
```
<DIV STYLE="border-style:inset"> Inset border</DIV>
```

```
<DIV STYLE="border-style:outset"> Outset border</DIV>
```

```
<DIV STYLE="border-style:outset double; color:red">  
Outset Double Blue border</DIV>
```

```
</BODY>
```

# output



# Dynamic Positioning

```
<Title> Dynamic Positioning</Title>
```

```
<Script>
```

```
var speed=5;
```

```
var count=10;
```

```
var firstLine="Text Growing";
```

```
var fontStyle=["monospace", "arial", "serif"];
```

```
var fontStyleCount=0;
```

```
function start()
```

```
{
```

```
  window.setInterval ("run()",100);
```

```
}
```

```
function run() {  
count +=speed;
```

```
if ((count%400)==0) {  
speed*=-1;  
pText.style.color=(speed<0)? "red":"blue";  
firstLine=(speed<0)? "Text Shrinking": "Text Growing";  
pText.style.fontFamily = fontStyle[++fontStyleCount %3];  
}
```

```
pText.style.fontSize=count/3;  
pText.style.left=count;  
pText.innerHTML=firstLine+ "<BR> Font Size: "+count+"px";  
}
```

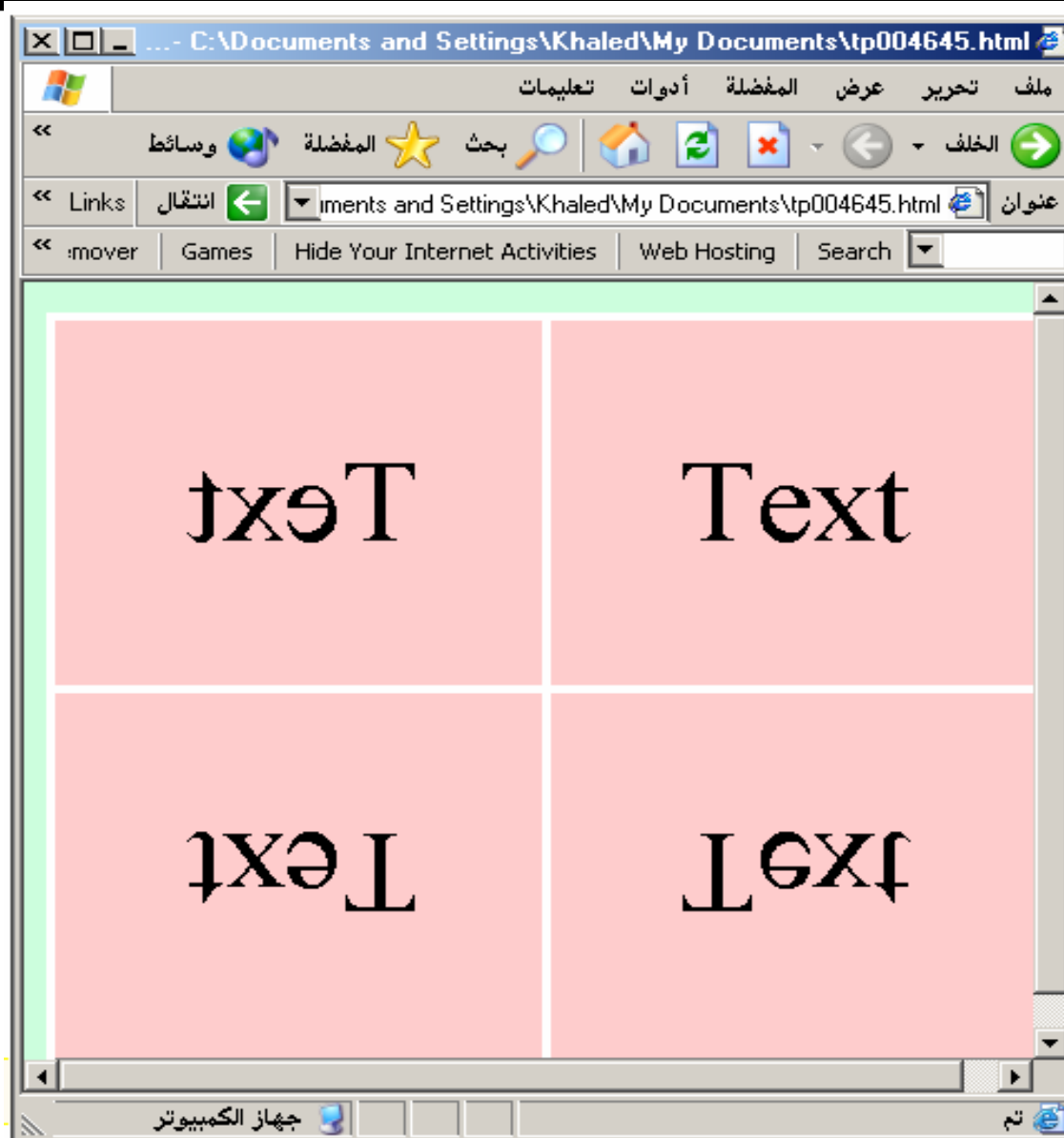
```
</SCRIPT>
```

```
<BODY onLoad="start()">  
<P id="pText" STYLE="position:absolute; left=0;  
font-family: Times ;color:lime">  
Welcome!</p>  
</BODY>
```

# Dynamic HTML: Filters

```
<style>
BODY{ background-color:#CCFFDD}
TABLE {font-size:2em; font-family:Arial;
background-color:#FFCCCC; border-style:outset;
border-collapse:collapse; }
TD { border-style:inset; text-align:center; padding:2ex; }
</style>
<BODY><TABLE>
<TR> <TD STYLE="filter:fliph">
Text</TD><TD>Text</TD></TR>
  <TR> <TD STYLE="filter:flipv fliph">Text</TD>
  <TD STYLE="filter:flipv">Text</TD></TR>
</TABLE>
</BODY>
```

# output



**Good Luck**