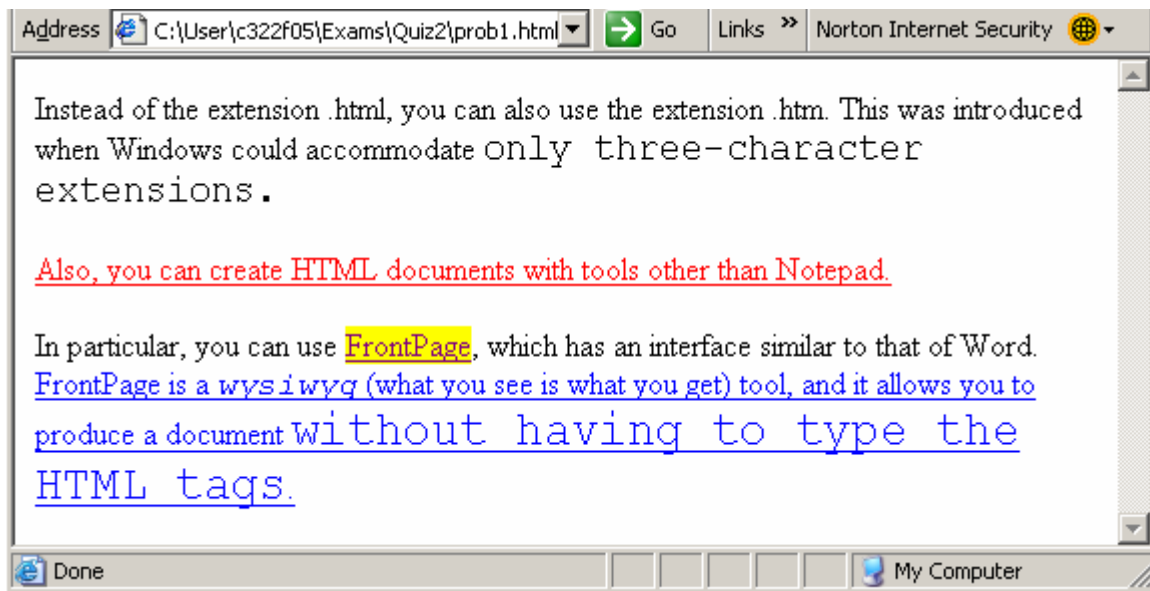


COMP 322 Internet Systems Fall 2007 Quiz 2—Solutions

Each of the two problems is worth 10 points.

1. The rendering below is of file `prob1.html`, which is listed on the next page with gaps. The first line of underlined text is red, and the three lines of underlined text at the bottom are blue. The word “FrontPage” (an anchor for a hyperlink that has been visited) is yellow. The presentation effects are produced by an external stylesheet, `prob1.css`, that imports another stylesheet, `probla.css`. In the listing of `prob1.html`, the gaps are labeled with Greek letters. These letters are repeated after the listing along with descriptions of the code that should go in the corresponding gaps. You there supply the missing code. . You are then asked to write the stylesheets.



```

<html>
<head>
  <title>Problem 1</title>
  α
</head>
<body>
  <p>
    Instead of the extension .html, you can also use the
    extension .htm. This was introduced when Windows could
    accommodate <span>only three-character extensions.</span>
  </p>

  <p β>
    Also, you can create HTML documents with tools other
    than Notepad.
  </p>

  <div>
    In particular, you can use
    <a href="Frontpage.html">FrontPage</a>,
    which has an interface similar to that of Word.
  </div>

  <div γ>
    FrontPage is a <em>wysiwyg</em> (what you see is what you get)
    tool, and it allows you to produce a document
    <span>without having to type the HTML tags</span>.
  </div>
</body>
</html>

```

α (2 pt.): A link to the stylesheet probl.css

Answer

```
<link rel="stylesheet" type="text/css" href="probl.css">
```

β (1 pt.): Apply the class under defined for p elements.

Answer

```
class="under"
```

γ (1 pt.): the class under defined for div elements.

Answer

```
class="under"
```

Now write the stylesheet `probl.css`. It imports the stylesheet `probla.css` and defines a pseudoclass for anchor (`a`) elements so that, after the hyperlink has been followed, the background color of the anchor is yellow. One class `under` is defined for `p` elements and another is defined for `div` elements. The `under` class restricted to `p` elements specifies that they be red and underlined, and the `under` class for `div` elements specifies that they be blue and underlined. Finally, a rule for `em` and `span` elements specifies that they have courier new font; if that font cannot be found, any serif font may be used.

Answer (6 pts.)

```
@import url(probla.css);
a:visited { background-color: yellow }
p.under   { text-decoration: underline; color: red }
div.under { text-decoration: underline; color: blue }
em, span  { font-family: "courier new", serif }
```

The following was removed from the quiz.

~~Finally, write the stylesheet `probla.css`. This has two rules. One specifies that `span` elements that are within `p` elements have content that is 120% the size of the surrounding text. The other specifies that `span` elements that are within `div` elements have content that is 150% the size of the surrounding text.~~

3. For this problem, you define a prototype object `Account`, whose instances represent bank accounts, with instance variables `owner` (a string—the owner’s name) and `balance` (an integer). This (depending on how you do it) may consist of just a constructor function with header

```
function Account( owner, balance )
```

The default value for `owner` is the string “unknown”, and that for `balance` is 0. Write `get` and `set` methods for both instance variables. When an instance of `Account` appears where a string is expected, it should return a string of the form “<owner>’s account”, where <owner> is the value of the `owner` instance variable.

Answer:

```
function Account( owner, balance )
{
  this.owner = owner || "unknown";
  this.balance = balance || 0;
  this.getowner = function() { return this.owner; };
  this.setowner = function( owner ) { this.owner = owner; };
  this.getbalance = function() { return this.balance; };
  this.setbalance = function( balance ) { this.balance = balance; };
  this.toString =
    function() { return this.owner + "'s account"; };
}
```