

```

<html>
<head><title> Count1 </title>
<script type="text/javascript">
function start()
{
document.getElementById("display").value = count;
count++;
t = setTimeout("start()", 1000);
}

```

```

function stop()
{
clearTimeout(t);
}

```

```

function reset1()
{
count = 0;
document.getElementById("display").value = count;
}

```

```

// -->
</script>
</head>
<body>
<form action="" method="get">
<div style="position: absolute; top: 500px; left: 500px">
<input type="button" value="start" onclick="start()" />
<input type="button" value="stop" onclick="stop()" />
<input type="button" value="reset" onclick="reset1()" />
</div>
</form>
</body>
</html>

```

**Question 1**  
 Dave Roof  
 Thailha N. Pham  
 Ben Aranguran

2)

```
<!ELEMENT name (first, last)>  
<!ELEMENT first (#PCDATA)>  
<!ELEMENT last (#PCDATA)>
```

Igor Polyakov  
Chien-Sen Wang

```
<?xml version="1.0"?>
```

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
```

```
targetNamespace="http://www.w3schools.com"
```

```
xmlns="http://www.w3schools.com"
```

```
elementFormDefault="qualified">
```

```
<xs:element name="name">
```

```
<xs:complexType>
```

```
<xs:sequence>
```

```
<xs:element name="first" type="xs:string" />
```

```
<xs:element name="last" type="xs:string" />
```

```
</xs:sequence>
```

```
</xs:complexType>
```

```
</xs:element>
```

```
</xs:schema>
```

4

```
<?xml version="1.0" encoding="utf-8" ?>
```

```
<xsl:stylesheet
```

```
  xmlns:xsl="XSLT transform namespace"
```

```
  xmlns="XHTML namespace" version="1.0" >
```

```
<xsl:template match="name" >
```

```
  <p><b>Name: </b>
```

```
    <xsl:apply-template />
```

```
  </p>
```

```
</xsl:template >
```

```
<xsl:template >
```

```
  <xsl:template match="first" >
```

```
    <i><xsl:value-of select="," /> </i>
```

```
  </xsl:template >
```

```
<xsl:template >
```

```
  <xsl:value-of select="," />
```

```
</xsl:template >
```

```
</xsl:stylesheet >
```

⑤ readyState property retrieves the current state of the XMLHttpRequest object

Its possible values are:

0 - Object isn't initialized w/ data

1 - Object is loading its data

2 - Object has finished loading its data

3 - <sup>data has started loading</sup> ~~user can interact~~, even though object isn't fully loaded

4 - object is completely initialized

Solution #6

# name.sql

```
CREATE TABLE IF NOT EXISTS Name (  
  First VARCHAR (15) NOT NULL,  
  Last  VARCHAR (15) NOT NULL  
);
```

11/13/06

⑦ #/usr/bin/perl

```
if (-e "numVisitors.txt")
{
    open (DATA, "numVisitors.txt") || die ("cannot open\n");
    $count = <DATA>;
    chomp($count);
    close (DATA);
}

print "Content-type: text/html\n\n";
print "<html><head><title>";
print "</title></head><body>";
print "<p>$count entered</p>";
open(OUT, ">7.txt") || die ("cannot open file\n");

count++;
print OUT $count, "\n";
close(OUT);
```

eight.pl

8

```
# /usr/bin/perl
```

```
use CGI qw/:standard/;
```

```
$fn = param('firstName');
```

```
if ($fn) {
```

```
open (OUT, ">> eight.txt") || die ("cannot open\n");
```

```
print OUT $fn, "\n";
```

```
close (OUT);
```

```
{  
print header, start-html,
```

```
start-form / -action => 'eight.pl',
```

```
-method => 'post';
```

```
-name => 'firstNameForm');
```

```
textfield / -name => 'first-Name',
```

```
submit / -name => 'submit', -value => 'submit');
```

```
end-form,
```

```
end-html;
```



9. Rewrite Problem 8 as a servlet.

```
import javax.servlet.*;
import javax.servlet.http.*;
public class Store extends HttpServlet {
    public void doGet(HttpServletRequest request, HttpServletResponse response)
        throws IOException, ServletException
    {
        File nameDat = new File("name.dat");
        String data;

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        out.println("<head><title>Store Servlet</title></head>");
        out.println("<body>");
        out.println("<h1>Store Servlet</h1>");
        String firstName = request.getParameter("firstname");
        if(firstName != null)
        {
            DataOutputStream outFile = new DataOutputStream ( new
                FileOutputStream(nameDat,true));
            //true means append to file

            outFile.writeUTF(data);
            outFile.flush();
            outFile.close();
        }
        out.print("<form action=\"");
        out.print("Store\" ");
        out.println("method=POST>");
        out.println("<input type=text size=20 name='firstname'>");
        out.println("<br>");
        out.println("<input type='submit'>");
        out.println("</form>");
        out.println("</body>");
        out.println("</html>");
    }
}
```