Introduction to Web Programming

Contents

[1 What is Web Programming? 1](#_Toc155600203)

[2 Example 3](#_Toc155600204)

# What is Web Programming?

Web programming is writing *applications* (*web apps*) that run on the web. Static webpages display information. Web apps are dynamic. They allow the user to accomplish a task (search for a topic, download music, buy a book, pay bills, *etc.*)A simplified version of the architecture of web programming is shown below:



A web app exists on a server and is composed of a number of web files. A web file contains HTML and possibly some client-side code (JavaScript), and server-side code intermingled. A user with a browser, for example, requests a file in the web application. The server loads the file and runs a parser. When the parser finds HTML, it sends it to an output buffer. When it finds code, it extracts and runs it. The output of the code is HTML, which is also sent to the output buffer. When the parser is finished, the contents of the output buffer are sent to the browser (Client). The client receives a web page consisting of HTML (and JavaScript), but no server-side code. It runs the Java Script and then displays the page.

We typically divide this paradigm into *client-side* code and *server-side* code. We begin this course with a brief treatment of the client-side code and then spend the majority of the course on server-side code.

The technologies we will use:

**Client-side**

1. HTML – a markup language used to build the structure of web pages and present data
2. CSS – used to provide style to the web page, look-and-feel
3. JavaScript – used to write code that runs in the browser. Does things like validate data before it is sent to the server.
4. jQuery – an API written on top of JavaScript to provide enhanced client-side behaviors
5. Responsive Design – CSS Grid, CSS FlexBox, Media Queries, *etc.*
6. Bootstrap – framework for building responsive designs.

**Client & Server-side**

1. AJAX – a paradigm that uses JavaScript to make asynchronous requests to a server. Not shown in the figure above, the client code can also communicate directly with the server.
2. XML – used to represent, transmit, and present data.
3. JSON – “JavaScript Object Notation is a standard text-based format for representing structured data based on JavaScript object syntax. It is commonly used for transmitting data in web applications.” In other words, you can convert JavaScript objects into JSON objects/arrays and pass them to the server. Next, the server can deserialize these as objects in the server-side language (C#, PHP, Java, *etc.*), process them, package the results up as JSON objects and send back to the client. It is a lighter weight alternative to XML.

**Server-side**

1. ASP.NET – An API that supports the .NET approach to writing web apps.
2. C# – the language we will use to program on the server side
3. MS Access or SQL Server – A database server for storing information. Spring 2022 – we will probably only use MS Access.
4. SQL – The language used to communicate with a database. We use C# to issue SQL statements to the database.
5. Web Service – A set of classes that reside on a server that provide services to web applications. Probably will not discuss this in Spring 2024.

# Example

<!-- HTML-client side -->

<html>

<head>

 <title>Drop Down Example</title>

 <!-- CSS-client side -->

 <style>

 hr {

 color: black;

 border-width: medium;

 }

 #ddIntRate {

 height: 73px;

 width: 105px;

 }

 </style>

 <!-- Javascript-client side -->

 <script type="text/javascript">

 var newBalances = [];

 function demoDropdown() {

 var dd = document.getElementById("ddIntRate");

 alert(dd.selectedOptions.);

 var msg = "Num items in dropdown: " + dd.length + "\n\n";

 msg += "Contents of dropdown's options array: \n";

 for (i = 0; i < dd.length; i++) {

 var opt = dd.options[i]

 msg += " index: " + i + ", ";

 msg += "text: " + opt.text + ", ";

 msg += "value: " + opt.value + ", ";

 msg += "selected: " + opt.selected + "\n";

 }

 }

 </script>

</head>

 <!—Call JavaScript -->

<body onload="setFocus(txtCurBal)">

 <h2>Drop Down Example</h2>

 <hr>

 <!-- PHP-server side -->

 <?php

 $level=5;

 echo ("<ul>");

 for ($i=0;$i<$level;$i++) {

 echo "<li>$i</li>" ;

 }

 echo ("</ul>");

 ?>

 <form id="AccountForm">

 <p>

 Type in your savings account balance:

 <input type="text" id="txtCurBal" size="7" value="0">

 </p>

 <p>

 Select an interest rate.

 <select size="1" id="ddIntRate" multiple>

 <option value="0.04">4.00%</option>

 <option value="0.0425">4.25%</option>

 <option value="0.045">4.50%</option>

 <option value="0.0475">4.75%</option>

 <option value="0.05">5.00%</option>

 </select>

 </p>

 <p>

 <!—Call JavaScript -->

 <input id="btnDDInfo" type="button" value="Demo DropDown" onclick="demoDropdown()" />

 </p>

 <p><textarea rows="4" id="txaNewBal" cols="60"></textarea></p>

 <p id="pNewBal"></p>

 <div id="divNewBal" style="border:2px solid #0066FF; display:inline-block; margin-bottom:10px" "></div>

 <table border="1">

 <tr>

 <td>Current Balance</td>

 <td>New Balance</td>

 </tr>

 <tr>

 <td id="currBal"></td>

 <td id="newBal"></td>

 </tr>

 </table>

 </form>

</body>

</html>