

Sample Exam

Write your name on the exam. Write something for every question. Students who do not write something for everything lose out over students who write down wild guesses. You will get some points if you attempt a solution but nothing for a blank sheet of paper. Write something down, even wild guesses. Problems take long to read but can be answered concisely.

Problem	Maximum	Score
1	10	
2	10	
3	10	
4	10	
5	10	
Total	50	

Problem 1 – HTML

a) Briefly describe the function of 5 of the following 7 HTML tags: `<h3></h3>`, `<pre></pre>`, `
`, `<tt></tt>`, ``, `<input>`, `<form></form>`

`<h3></h3>` is used for a level 3 heading (smaller than H1 and H2, larger than text).

`<pre></pre>` is used for preformatted text (browser preserves indentation and newlines in HTML source, uses monospaced font).

`
` introduces a line break into the page.

`<tt></tt>` asks browser to use monospaced font (a.k.a. teletype or typewriter text).

`` is used for ordered (numbered) lists.

`<input>` is used for various types of controls (text fields, checkboxes, radio buttons) the user can enter information with.

`<form></form>` is used as a container for input controls that are submitted to the server together.

b) Show how HTML tables can be used to achieve the text layout on the right (there is more than one valid solution). Do not worry about the width and height of cells; just give a table that produces this relative alignment of the text in the five cells.

Text 1	Text 2	Text 4
	Text 3	
Text 5		

This type of layout can be accomplished either using the `colspan` and `rowspan` attributes, or by adding tables inside cells of other tables (this second method does not work for all layouts).

```
<table>
  <tr>
    <td rowspan="3">Text 1</td>
    <td>Text 2</td>
    <td rowspan="2">Text 4</td>
  </tr>
  <tr>
    <td>Text 3</td>
  </tr>
  <tr>
    <td colspan="2">Text 5</td>
  </tr>
</table>
```

Problem 2 – C# and database programming

Write a short C# program that does the following:

Using Odbc commands and based upon the schema of the example database gone over in class, write a program that prints a class roster of names and email addresses for people who took CS 302 in Fall of 2005 (this will be a three way join). Error conditions are to be handled with exceptions. It is not necessary to get the schema exactly right (column and table names, for example – we're looking for concepts. You will be using: OdbcConnection, OdbcCommand, OdbcDataReader. Use parameterized queries for catalog number, term, and year.

```
using System;
using System.Collections.Generic;
using System.Text;
using System.Data.Odbc;
using System.Data;

namespace Demol{
    class Program{
        static string connectionString="don't care about it";

        static void Main(string[] args){
            OdbcCommand c = new OdbcCommand();
            c.Connection = new OdbcConnection(connectionString);

            c.CommandText = "select people.firstname, people.lastname, people.email"+
                "from people,grades,courses "+
                "where people.id=grades.studentid and grades.courseid=courses.id and"+
                " courses.catalognumber=? and grades.semester=? and grades.year=?";
            c.Parameters.AddWithValue("@catalognumber", "302");
            c.Parameters.AddWithValue("@semester", "Fall");
            c.Parameters.AddWithValue("@year", "2005");

            OdbcDataReader reader = null;

            try{
                c.Connection.Open();
                reader = c.ExecuteReader();
                while (reader.Read()){
                    Console.WriteLine(string.Format("{0} {1} {2}", reader[0].ToString(),
                        reader[1].ToString(), reader[2].ToString()));}
            }catch (Exception e){
                Console.Error.WriteLine(e.Message);
            }finally{
                if (c.Connection.State == ConnectionState.Open)
                    c.Connection.Close();
                if (reader != null && !reader.IsClosed)
                    reader.Close();
            }

            Console.WriteLine("Hit return to exit.");
            Console.Read();
        }
    }
}
```

Problem 2 – HTTP and state (server side programming)

a) Explain briefly what the following sentence means: “HTTP is a stateless protocol.”

Individual HTTP requests are handled independently from previous ones by the server. There is no information at the web server built based on previous HTTP requests (“server state”) that would affect the outcome of a given request.

b) Give at least two reasons why a server using dynamic pages with server-side programs would want to keep state across HTTP requests.

- 1) *To remember the identity of the user*
- 2) *To remember previous selections (e.g. shopping cart, user preferences)*
- 3) *To remember user preferences set in previous pages*

c) Explain briefly how session state works in ASP.NET .

The server places a cookie on the client and stores the objects associated with the session in server memory. Each client has a different session. When the browser makes subsequent requests, it sends the cookie and the server uses it to identify the appropriate session data which becomes available when the new requests are processed. Session data can be read and updated. Session data can also be stored at a state server or database server. If the data of a given session is not accessed by any page for a certain time, it is discarded by the server.

Problem 4 – JavaScript

a) What does the keyword `this` refer to if used inside the method of an object? What does it refer to if it is used outside an object method?

Inside an object's method it refers to the object, outside it refers to the "global object" (the `window` variable also points to it).

b) What happens if you assign to an object property that does not exist (the property)? What happens if you read an object property that does not exist?

When assigning to an inexistent property, the property is created and the value is stored in it. When reading an inexistent property, the `undefined` value is read.

Note: reading an inexistent property does not throw an exception, but using the `undefined` value may cause errors later in the script.

c) When does the JavaScript code between `<script></script>` tags get executed?

It is executed as the page is being rendered. The HTML elements from after the tag will not be there in the document yet.

Note: In JavaScript, defining a function is equivalent to assigning a function object to a variable (with the same name as the function). Thus while the function may be executed later, say by an event handler, the binding of the code to the name of the function happens when the page is loaded.

d) Why is it that you shouldn't use `document.write()` inside an event handler?

`document.write()` will insert text (or HTML markup) into the page if used inside a `<script></script>` tag in the body of the document. By the time an event handler is executed the page is already built and calling `document.write()` will overwrite the page (which is probably not what you want).

e) Give the name of a DOM function you can use to find a specific element of an HTML document.

`getElementById()`

Problem 5 – defining various terms

Explain concisely the difference between the following pairs of terms

a) Explain the difference between “content” and “presentation” and explain why it is important to separate them

“Content” refers to the actual information conveyed by the document and “presentation” refers to instructions about how the information should be presented to the user. By separating the two different people can work on content and presentation and it is easy to update one without affecting the other.

b) Lossy compression vs. lossless compression

Lossless compression allows the receiver to reconstruct the exact byte sequence that was compressed; lossy compression, used for images and sounds, allows only the reconstruction of a similar data (similarity being defined by the human senses), but the compressed data is smaller.

c) Explain the difference between persistent and temporary state and give at least an example of a situation where both types of state exist.

Temporary state is discarded quickly whereas persistent state is not. For example the user name and password entered by the user for HTTP authentication are kept only until the browser is closed and are not available when the browser is restarted and thus they are temporary state. If a site uses cookies for authentication, they can be retained by the browser for a long time and survive many browser restarts and thus these cookies are persistent state.

d) AJAX asynchronous requests vs. ASP.NET postback

As a result of an ASP.NET postback, the server re-builds the entire page and sends it back to the client, the response to an AJAX request is typically much smaller and it results in only a portion of the page being rebuilt through DOM manipulation at the client.

e) SOAP vs. REST

SOAP request are in XML, REST requests are a sequence of name-value pairs. SOAP requests are more expressive, but impose more overhead.