COMP 2406/2006: Fundamentals of Web Applications

Winter 2013 Mid-Term Exam

February 14, 2013

Below there are 28 questions on 3 pages (not including this one). Each question is worth one point except for those marked otherwise; all together there are 34 points. Answer all questions. For each question either fill in the blank, circle the correct answer, or write below the question. You have 50 minutes. Good luck!

Page	Received Points	Max. Points
2		18
3		8
4		8
Total		**34**

1. _____ What is the name of your team?

2. What is the name(s) of your teammate(s)?

3. **(true / false)** HTTP cookies are only sent to a web server when explicitly requested.

4. **(true / false)** Cookies are normally parsed by client-side JavaScript.

5. **(true / false)** Arbitrary (even undefined) properties of regular defined object can be accessed in JavaScript without generating a runtime error.

6. **(true / false)** The `prototype` property is used when retrieving an object's properties.

7. **(true / false)** The `for in` construct in JavaScript is problematic because it loops over all of the accessible properties of an object.

8. **(true / false)** `var` causes a symbol to be defined within the inner most function context.

9. **(true / false)** Write access to object properties are resolved by looking at the object's properties and then by looking at the chain of objects pointed to by the objects on the `prototype` chain.

10. **(true / false)** The `apply` function invocation pattern allows you to set the value of `this`.

11. **(true / false)** The use of `that` is normally to get around the limitations of the function invocation pattern—namely, that `this` gets bound to the global object.

12. **(true / false)** In general, HTML forms can be submitted successfully even if the client has disabled JavaScript.

13. **(true / false)** In an HTML form, if I want to return a value called "country", I set the `id` parameter to "country".

14. _____ In the constructor invocation pattern, `this` is bound to which object (at the start of the function)?

15. _____ What is the biggest constraint on the performance of web applications?

16. _____ What HTTP command is normally used to request the contents of a page?

17. _____ In the method invocation pattern, a function returns what value if there is no explicit `return` statement?

18. _____ In `git`, which is the term used for making a logical local copy of the current files in a local repository, generally for the purpose of implementing a feature that may not be ready for incorporation into the mainline of the code? `branch`, `clone`, or `fork`?

```
1  // Client side JavaScript example
2  var FormExample = (function(FormExample, $, undefined){
3
4      var COUNTRIES = ["America", "Canada", "Mexico", "Russia", "France"];
5
6      $(function(){
7          $("#country").autocomplete({
8              source: COUNTRIES
9          });
10
11          var sendEmail = $("#send-email");
12          sendEmail.on("change", function(event){
13              if(sendEmail.is(":checked")){
14                  $("#email-info").show();
15              }else{
16                  $("#email-info").hide();
17              }
18          }).change();
19
20      return FormExample;
21  })(FormExample || {}, $);
```

19. If we assume `#country` refers to an input field, then the autocomplete in lines 7-9 still allows arbirtrary text to be entered into it (assuming the use of jQuery-UI). (true/false)

20. $ is in general the short name for what function in modern JavaScript code?

21. If we assume that `FormExample` is not defined on line 1, then what is the value of FormExample on line 3?

22. Under what conditions is the anonymous function defined on line 12 called?

23. (2) What change would omitting the `var` keyword in line 4 cause? Would this change effect the correct execution of this function? Explain.

24. (2) What is the purpose of the call to `.change()` on line 18?

```
 1  // Server-side JavaScript example
 2  // app.js from express
 3  var express = require('express')
 4    , routes = require('./routes')
 5    , http = require('http')
 6    , path = require('path');
 7
 8  var app = express();
 9
10  app.configure(function(){
11    app.set('port', process.env.PORT || 3005);
12    app.set('views', __dirname + '/views');
13    app.set('view engine', 'jade');
14    app.use(express.favicon());
15    app.use(express.logger('dev'));
16    app.use(express.bodyParser());
17    app.use(express.methodOverride());
18    app.use(app.router);
19    app.use(require('stylus').middleware(__dirname + '/public'));
20    app.use(express.static(path.join(__dirname, 'public')));
21  });
22
23  app.configure('development', function(){
24    app.use(express.errorHandler());
25  });
26
27  app.get('/', routes.index);
28
29  app.post('/add', routes.add);
30  app.post('/delete', routes.add);
31
32  http.createServer(app).listen(app.get('port'), function(){
33    console.log("Express server listening on port " + app.get('port'));
34  });
```

25. (**true / false**)   The only HTTP GET command allowed by this application is for /.

26. In the `http` module, `createServer()` should be originally assigned as a method of what object?

27. (2) What port will this server listen on? Explain how you know.

28. (4) List all functions—including methods—that are being *invoked* (directly called) on lines 32-34 and explain very briefly what each does. (Do not include any functions that are just being defined, are being called implicitly, or will be called during a later function invocation.)