Schedule

A week-by-week breakdown of the material.

Week 1 (01/05-01/09)

- Day 1
  - History of Javascript[1]
  - Components of a Web Page[2]
  - Survey of HTML[3]
- Day 2
  - Survey of CSS[4]
  - A sample page[5]
  - Lab 1[6]
- Day 3
  - How to run Javascript[7]
  - Numbers in Javascript[8]
  - Local and Global Variables[9]
  - Basics of Arrays[10]

Week 2 (01/12-01/16)

- Day 1
  - Basics of Objects: Object literals[11]
  - Equality in Javascript[12]
  - Strings and Regular Expressions[13]
  - Function basics[14]
  - Lab 2[15]

[1] notes/history.html
[2] notes/page_components.html
[3] notes/html_survey.html
[4] notes/css_survey.html
[5] notes/sample_page.html
[6] labs/1.html
[7] notes/how_to_run.html
[8] notes/numbers.html
[9] notes/local_vs_global.html
[10] notes/array_basics.html
[11] notes/object_basics.html
[12] notes/equality.html
[13] notes/strings_and_re.html
[14] notes/functions_basic.html
[15] labs/2.html

  - Quiz 1[16]

- Day 2
  - No class
- Day 3
  - No class

Week 3 (01/19-01/23)

- Day 1
  - Functions as values[17]
  - Arrow Functions[18]
  - Array Collection methods[19]
- Day 2
  - Function Usage Patterns[20]
  - ES6 Classes[21]
  - Lab 3[22]
- Day 3
  - Function call forms and the value of this[23]
  - Introduction to the DOM (Document Object Model)[24]

Week 4 (01/26-01/30)

- Day 1
  - Introduction to jQuery[25]
  - Introduction to DOM Events[26]
  - Timers[27]
- Day 2
  - Modules

[16] https://moodle.hanover.edu/mod/quiz/view.php?id=15591
[17] notes/function_values.html
[18] notes/function_arrow.html
[19] notes/array_collection_methods.html
[20] notes/function_usage_patterns.html
[21] notes/classes.html
[22] labs/3.html
[23] notes/function_calls_and_this.html
[24] notes/dom_intro.html
[25] notes/jquery_intro.html
[26] notes/events_intro.html
[27] notes/events_timers.html

  - Overview of Software Development Practices[28]
  - Basics of Version Control[29]
  - Quiz 2[30]
- Day 3
  - The DRY principle[31]
  - Object properties[32]
  - Different stack implementations[33]
  - Object creation and prototypes[34]

Week 5 (02/02-02/06)

- Day 1
  - "Class" construction examples[35]
  - Testing Basics[36]
- Day 2
- Day 3

Week 6 (02/09-02/13)

- Day 1
- Day 2
  - TaskApp: Building a Web App[37]
- Day 3

Week 7 (02/16-02/20)

- Day 1
  - TaskApp: Building a Web App[38]
- Day 2

[28] notes/dev_overview.html
[29] notes/git_version_control.html
[30] https://moodle.hanover.edu/mod/quiz/view.php?id=33676
[31] notes/dry.html
[32] notes/object_properties.html
[33] notes/stack_various.html
[34] notes/object_creation_prototypes.html
[35] notes/class_construction.html
[36] notes/testing_basics.html
[37] notes/taskapp_setup.html
[38] notes/taskapp_setup.html

  - TaskApp: Building a Web App[39]

- Day 3

Week 8 (02/23-02/27)

BREAK

Week 9 (03/02-03/06)

- Day 1
  - Assignment of Projects[40]
  - Using Templates: Handlebars[41]
- Day 2
  - Work on Projects
- Day 3

Week 10 (03/09-03/13)

- Day 1
  - Graphics in Javascript, SVG[42]
- Day 2
  - Custom Events: Oberver, Publish/Subscribe[43]
- Day 3

Week 11 (03/16-03/20)

- Day 1
  - Documentation systems[44]
  - Code Reviews[45]
- Day 2

[39] notes/taskapp_setup.html
[40] notes/project_descriptions.html
[41] notes/templates.html
[42] notes/graphics.html
[43] notes/custom_events.html
[44] notes/documentation.html
[45] notes/code_review.html

  - Module Patterns: Node/CommonJS modules[46]
  - Using require.js[47]
  - UMD modules[48]
- Day 3

Week 12 (03/23-03/27)

- Day 1
  - The Visitor Pattern and the need for it[49]
- Day 2
  - Basics of HTTP[50]
  - Introduction to XMLHttpRequest[51]
- Day 3

Week 13 (03/30-04/03)

- Day 1
  - Projects progress
- Day 2
  - Projects progress
- Day 3

Week 14 (04/06-04/10)

- Day 1
  - Projects progress
- Day 2
  - What we left out[52]
  - Final Study Guide[53]
- Day 3

[46] notes/pattern_modules.html
[47] notes/requirejs.html
[48] notes/umd.html
[49] notes/patterns_visitor.html
[50] notes/http_intro.html
[51] notes/xhr_intro.html
[52] notes/left_out.html
[53] notes/final_study_guide.html

Other topics we did not cover

- MVC: Model-View-Controller[54]
- Intro to Patterns: Iterator[55]
- Lab 5: Collections[56]
- Design Patterns[57]
- Patterns of code reuse[58]
- Pattern: Command[59]
- Pattern: Proxy[60]
- Pattern: Visitor[61]
- Pattern: Visitor (cont)[62]
- Lab 6: Double-Linked Lists, History[63]
- Pattern: Composite[64]
- Patterns: Adapter and Facade[65]
- Example of Designing a Model[66]

Old notes

- Functions as closures[67]
- Functions as a means for creating a scope/context[68]
- Git basics[69]
- Review of Git commands[70]
- GitHub basics[71]
- Remotes and Branches[72]
- GitHub Workflow[73]
- Overall Project Workflow[74]
- Methods in objects and the need for this[75]

[54] notes/pattern_mvc.html
[55] notes/patterns_iterator.html
[56] labs/5.html
[57] notes/design_patterns.html
[58] notes/code_reuse.html
[59] notes/patterns_command.html
[60] notes/patterns_proxy.html
[61] notes/patterns_visitor.html
[62] notes/patterns_visitor.html
[63] labs/6.html
[64] notes/patterns_composite.html
[65] notes/patterns_adapter_facade.html
[66] notes/design_example.html
[67] notes/function_closures.html
[68] notes/functions_for_scope.html
[69] notes/git_basics.html
[70] notes/git_commands_review.html
[71] notes/github_basics.html
[72] notes/git_remotes_branches.html
[73] notes/github_workflow.html
[74] notes/project_workflow.html
[75] notes/object_methods.html

- Setting up a second remote[76]

[76] notes/second_remote.html