

Schedule

A week-by-week breakdown of the material.

Week 1 (01/05-01/09)

- Day 1
 - History of Javascript¹
 - Components of a Web Page²
 - Survey of HTML³
- Day 2
 - Survey of CSS⁴
 - A sample page⁵
 - Lab 1⁶
- Day 3
 - How to run Javascript⁷
 - Numbers in Javascript⁸
 - Local and Global Variables⁹
 - Basics of Arrays¹⁰

Week 2 (01/12-01/16)

- Day 1
 - Basics of Objects: Object literals¹¹
 - Equality in Javascript¹²
 - Strings and Regular Expressions¹³
 - Function basics¹⁴
 - Lab 2¹⁵

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 1¹⁶
- Day 2
 - No class
- Day 3
 - No class

Week 3 (01/19-01/23)

- Day 1
 - Functions as values¹⁷
 - Arrow Functions¹⁸
 - Array Collection methods¹⁹
- Day 2
 - Function Usage Patterns²⁰
 - ES6 Classes²¹
 - Lab 3²²
- Day 3
 - Function call forms and the value of this²³
 - Introduction to the DOM (Document Object Model)²⁴

Week 4 (01/26-01/30)

- Day 1
 - Introduction to jQuery²⁵
- Day 2
 - Practice with jQuery
- Day 3
 - Introduction to DOM Events²⁶

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](#)

Week 5 (02/02-02/06)

- Day 1
 - Timers²⁷
 - Overview of Software Development Practices²⁸
- Day 2
 - Basics of Version Control²⁹
- Day 3
 - Module Patterns: Node/CommonJS modules³⁰
 - TaskApp: Building a Web App³¹

Week 6 (02/09-02/13)

- Day 1
 - Testing Basics³²
 - TaskApp (cont)³³
- Day 2
 - Custom Events: Observer, Publish/Subscribe³⁴
- Day 3
 - Observer pattern in TaskApp.

Week 7 (02/16-02/20)

- Day 1
 - Boundaries: controller vs UI/view.
 - Test doubles in the controller.
- Day 2
 - ES6 features: String Templates, Symbols
 - ES6 features: Promises
 - Quiz 2³⁵

²⁷[notes/events_timers.html](#)

²⁸[notes/dev_overview.html](#)

²⁹[notes/git_version_control.html](#)

³⁰[notes/pattern_modules.html](#)

³¹[notes/taskapp_setup.html](#)

³²[notes/testing_basics.html](#)

³³[notes/taskapp_setup.html](#)

³⁴[notes/custom_events.html](#)

³⁵<https://moodle.hanover.edu/mod/quiz/view.php?id=17430>

- Day 3
 - MIDTERM³⁶

Week 8 (02/23-02/27)

BREAK

Week 9 (03/02-03/06)

- Day 1
 - Lab 4³⁷
- Day 2
 - Assignment of Projects³⁸
- Day 3
 - Using Templates: Handlebars³⁹
 - Object creation and prototypes⁴⁰

Week 10 (03/09-03/13)

- Day 1
 - Object properties⁴¹
- Day 2
 - Locking things down⁴²
- Day 3
 - Graphics in Javascript, SVG⁴³

³⁶[midtermStudyGuide.html](#)

³⁷[labs/4.html](#)

³⁸[notes/project_descriptions.html](#)

³⁹[notes/templates.html](#)

⁴⁰[notes/object_creation_prototypes.html](#)

⁴¹[notes/object_properties.html](#)

⁴²[notes/locking_things_down.html](#)

⁴³[notes/graphics.html](#)

Week 11 (03/16-03/20)

- Day 1
 - Projects progress
- Day 2
 - Projects progress
- Day 3
 - Documentation systems⁴⁴

Week 12 (03/23-03/27)

- Day 1
 - Code Reviews⁴⁵
- Day 2
 - Basics of HTTP⁴⁶
 - Introduction to XMLHttpRequest⁴⁷
- Day 3
 - Asynchronous loading example⁴⁸

Week 13 (03/30-04/03)

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

⁴⁴[notes/documentation.html](#)

⁴⁵[notes/code_review.html](#)

⁴⁶[notes/http_intro.html](#)

⁴⁷[notes/xhr_intro.html](#)

⁴⁸[notes/asynchronous_loading_example.html](#)

Week 14 (04/06-04/10)

- Day 1
 - Projects progress
- Day 2
 - What we left out⁴⁹
 - Final Study Guide⁵⁰
- Day 3

Other topics we did not cover

- MVC: Model-View-Controller⁵¹
- Intro to Patterns: Iterator⁵²
- Lab 5: Collections⁵³
- Design Patterns⁵⁴
- Patterns of code reuse⁵⁵
- Pattern: Command⁵⁶
- Pattern: Proxy⁵⁷
- Pattern: Visitor⁵⁸
- Pattern: Visitor (cont)⁵⁹
- Lab 6: Double-Linked Lists, History⁶⁰
- Pattern: Composite⁶¹
- Patterns: Adapter and Facade⁶²
- Example of Designing a Model⁶³

Old notes

- Functions as closures⁶⁴

⁴⁹[notes/left_out.html](#)

⁵⁰[notes/final_study_guide.html](#)

⁵¹[notes/pattern_mvc.html](#)

⁵²[notes/patterns_iterator.html](#)

⁵³[labs/5.html](#)

⁵⁴[notes/design_patterns.html](#)

⁵⁵[notes/code_reuse.html](#)

⁵⁶[notes/patterns_command.html](#)

⁵⁷[notes/patterns_proxy.html](#)

⁵⁸[notes/patterns_visitor.html](#)

⁵⁹[notes/patterns_visitor.html](#)

⁶⁰[labs/6.html](#)

⁶¹[notes/patterns_composite.html](#)

⁶²[notes/patterns_adapter_facade.html](#)

⁶³[notes/design_example.html](#)

⁶⁴[notes/function_closures.html](#)

- Functions as a means for creating a scope/context⁶⁵
- Git basics⁶⁶
- Review of Git commands⁶⁷
- GitHub basics⁶⁸
- Remotes and Branches⁶⁹
- GitHub Workflow⁷⁰
- Overall Project Workflow⁷¹
- Methods in objects and the need for this⁷²
- Setting up a second remote⁷³
- AMD Modules⁷⁴
- Using require.js⁷⁵
- UMD modules⁷⁶
- The Visitor Pattern and the need for it⁷⁷
- The DRY principle⁷⁸
- Different stack implementations⁷⁹
- “Class” construction examples⁸⁰

⁶⁵[notes/functions_for_scope.html](#)

⁶⁶[notes/git_basics.html](#)

⁶⁷[notes/git_commands_review.html](#)

⁶⁸[notes/github_basics.html](#)

⁶⁹[notes/git_remotes_branches.html](#)

⁷⁰[notes/github_workflow.html](#)

⁷¹[notes/project_workflow.html](#)

⁷²[notes/object_methods.html](#)

⁷³[notes/second_remote.html](#)

⁷⁴[notes/amd_modules.html](#)

⁷⁵[notes/requirejs.html](#)

⁷⁶[notes/umd.html](#)

⁷⁷[notes/patterns_visitor.html](#)

⁷⁸[notes/dry.html](#)

⁷⁹[notes/stack_various.html](#)

⁸⁰[notes/class_construction.html](#)