STEP

IT ACADEMY

Syllabus

Front-end

Creating Web Pages Using HTML5 and CSS3

Version 4.0.0

Course duration: 48 classes

1 class = 80 minutes

Course objective

To teach students to develop and lay out static web pages using HTML5 and CSS3. To present a full picture about the technological chain of creating web sites and to form understanding of trends in web technologies. To teach students to pick the most suitable way to develop web pages. To teach them to test and check the code of web pages.

Upon completion of the course, students will:

- know and apply HTML basics: tags, attributes, and methods of structuring the content of web pages in order to create formatted documents;
- know and apply CSS basics: values, lists, colors, fonts, and other formatting metrics;
- have skills of checking and debugging the code of web documents;
- have skills of formatting contents of web documents for various screens: from standard browsers to mobile devices;
- have skills of fast and quality formatting of complex web documents;
- know HTML5 and CSS3 basics.

Any free editor can be used to create a web site: Notepad++, Microsoft Visual Studio Community.

Upon completion of this course, students submit practical task and takes the test of knowledge on the course materials. In order to be allowed for the test, all homework and practical assignments must be submitted. Students must create a website and put it on the Internet. Basic requirements: box layout, valid code.

Topic Plan

Module	**1.** Introduction to Web Technologies. Structure of HTML. Text Formatting with HTML	**4 classes**
Module	**2.** Formatting with CSS. Lists. CSS Paddings and Margins	**3 classes**
Module	**3.** Graphics in web design. Optimizing Images. Hyperlinks. Navigation Principles of Websites	**6 classes**
Module	**4.** Animation in CSS3	**3 classes**
Module	**5.** Tables	**2 classes**
Module	**6.** Positioning. Layout of Web Pages with Blocks	**4 classes**
Module	**7.** Adaptive Design	**2 classes**
Module	**8.** Flexbox	**2 classes**
Module	**9.** Grid Layout	**2 classes**
Module	**10.** Forms. Frames	**2 classes**
Module	**11.** Version Control Systems	**4 classes**
Module	**12.** Bootstrap	**6 classes**
Module	**13.** Materialize	**4 classes**
Module	**14.** Less	**2 classes**
Module	**15.** Exam	**2 classes**

Module 1. Introduction to Web Technologies. Structure of HTML. Text Formatting with HTML

1. Introduction to the subject.
2. Introduction to markup languages. Hypertext Markup Language (HTML).
   - Internet.
   - HTTP.
   - Development of HTML, versions. Version HTML5.
   - Cross browser compatibility issues. Browser war.
   - W3C.
3. Tags as a basic element of the HTML structure. The rules of writing tags and their attributes in the HTML5 standard. Syntax differences of HTML4, XHTML, and HTML5.
4. Main mistakes when writing tags.
   - <!DOCTYPE HTML> specifications.
   - Validating with FireFox—HTML Validator add-on.
   - The concept of well-formed.
   - HTML5 ancestors: SGML and XML.
5. The structure of an HTML5 document.
   - Basic elements and their purposes.
   - New structural elements: <header>, <nav>, <section>, <article>, <aside>, <footer>. Availability of new tags in modern browsers. How new tags are displayed in obsolete browsers.
6. Character encodings of a page and <meta> tags.
   - The <meta> tag—setting information about the page (expires, refresh, autor, copyright, keywords, description).
   - Declaring character encodings with the <meta> tag.
   - Substitute character and encodings.
7. Classification of tags: inline and block.
   - Inline.
   - Block.
8. Text formatting model: headers and paragraphs. The <p>, <h1>..<h6> elements. Text alignment within inline elements: the align attribute.
9. Classification of tags: logical and physical tags.
   - Physical style tags.

- Logical style tags.
- Brief overview of the main logical style tags: <abbr>, <acronym>, <cite>, <code>, <del>, <dfn>, <ins>.

10. Practice: Creating a simple web page.

Module 2. Formatting with CSS. Lists. CSS Paddings and Margins

1. Cascading Style Sheets (CSS).
   - Introduction. Overview of versions. Purpose: HTML sets the structure, and CSS formatting.
   - Adding CSS to HTML with the style attribute. Rules for writing CSS properties.
2. Tags containing no formatting: <div> is a block tag; <span> is an inline tag.
3. Analogy of HTML and CSS by the example of inline and block tags.
4. Additional properties of CSS for text formatting: letter-spacing, line-height, text-intend, text-transform, white-space, word-spacing.
5. The use of class and id attributes to set styles.
   - Creating styles for tags, classes, identifiers inside the <style> tag. The concept of selectors. The rule for writing selectors: tag selector, class selector, id selector, universal selector *.
   - Style priority (tag / class / id / style). The !important rule to change priority.
   - Style inheritance. Standard property values.
   - Tracing styles with firebug (Firefox add-on).
6. The use of the external CSS style files.
   - Adding CSS files with the <link> tag and @import instruction.
   - CSS files and browser cache.
7. Practice: Text formatting by means of CSS.
8. Creating lists.
   - Unordered lists: <ul>, <li> elements.
   - Ordered lists: <ol>, <li> elements.
   - The type, value, start attributes.
9. Creating nested lists.
10. Formatting lists with CSS.
    - Properties: list-style-type, list-style-image, list-style-position.
    - The shorthand list-style property.
    - Styling multilevel lists. Nested selectors.

11. Definition lists: <dl>, <dd>, and <dt> elements.
12. Managing paddings and margins.
    - The margin property and its children margin-left, margin-top, margin-right, margin-bottom.
    - The padding property and its children padding-left, padding-top, padding-right, padding-bottom.
    - The difference between padding and margin and their purposes.
    - Removing indentations for some tags: <body>, <h1>..<h6>, <p>.
13. Practice: Creating lists.

Module 3. Graphics in Web Design. Optimizing Images. Hyperlinks. Navigation Principles of Websites

1. Basics of working with Adobe Photoshop/GIMP.
    - What is Adobe Photoshop/GIMP?
    - Basic terms of Adobe Photoshop/GIMP.
    - Layers and tools.
    - Tools for cutting, filtering, and meshes.
2. Graphics file formats in web.
3. The <img /> tag and its attributes (src, alt, width, height, border).
    - The border property as an analogue of the border attribute.
    - Setting margin, padding, border properties for an image.
    - Image alignment with the align attribute. The float property as an analogue of the align attribute.
4. Background of a page: the background property.
    - Setting color as a background: background-color. Required background for the <body> element.
    - Setting image as a background: background-image, background-repeat, background-position, background-attachment.
    - Images and browser cache.
5. General information about hyperlinks.
    - The <a> tag and its attributes (href, target).
    - Ergonomics, convenient navigation.
6. Absolute and relative addressing.
    - Managing external links.

- Managing internal links with the <a> element. The id and name attributes.
- Managing the mixed transition (to the specified element in an external HTML document).
- Image links. Removing link border.

7. Creating a menu by means of the structure of lists (<ul>, <li>), their formatting. The display property. Converting a link into a block element.
8. Pseudo-classes.
   - Link pseudo-classes: active, hover, link, visited.
   - Pseudo-classes for regular elements: first-child, first-line, first-letter.
9. The cursor property of CSS.
10. Practice: Development of an image gallery.
11. CSS3 properties.
    - Background: setting gradients, resizing background, the background and background-size properties.
    - Borders: rounding edges, the border-radius property.
    - Setting translusency to page elements, the opacity property.
    - Full support of the CSS 2.1 selectors.
12. Multimedia.
    - Insert video in the page with the <video> tag.
    - Insert audio in the page with the <audio> property.
    - Creating images and animation with the <canvas> tag.
    - SVG format.

Module 4. Animation in CSS3

1. 2D transformation functions.
   - matrix()
   - translate(), translateX(), translateY()
   - scale(), scaleX(), scaleY()
   - rotate()
   - skew(), skewX(), skewY()
2. CSS3 filters.
   - blur()
   - brightness()
   - contrast()

- drop-shadow()
- grayscale()
- hue-rotate()
- opacity()
- saturate()
- sepia()

3. The @keyframes rule.
4. The animation property and its components.

Module 5. Tables

1. Creating a basic table. The <table>, <tr>, and <td> tags.
   - The border, cellspacing cellpadding attributes. Possible CSS analogues: border, padding.
   - Specifying the width and height of a cell: the width and height attributes. Rules for setting width and height. CSS analogues: the width and height properties.
   - Data alignment in a table: align and valign. CSS analogues: text-align, vertical-align.
   - Managing background color and border color of a table (of a separate row, separate cell).
   - Images as a table background (of a separate row, separate cell).
2. Joining cells: colspan and rowspan attributes.
3. Tags of the logical table structuring: <thead>, <tbody>, <tfoot>. Tags of logical column grouping: <colgroup>, <col>.
4. Managing table borders: frame and rules attributes.
5. Practice: Creating complex tables.
6. Basics of creating table layout. Example of a table layout: its disadvantages.

Module 6. Positioning. Layout of Web Pages with Blocks

1. The position property.
   - Positioning: relative and absolute.
   - The top, left, bottom, right properties.
2. The visibility and overflow properties.
3. Practice.

4. Basics of the box layout. Layout rules.
   - Block nesting.
   - Setting block width and height with the width and height properties.
   - Block floating. Cancel block floating. The float and clear properties.
   - The rules for paddings and margins.
   - Setting minimum height and width for a block: min-height, min-width properties. Setting these properties in IE6.
   - Inline-block alignment (margin, text-align, line-height, position). Cross-browser alignment.
5. Considering the simplest structures of pages.
   - The structure of a fixed size website.
6. Responsive structure. Blocks with negative margin.

Module 7. Adaptive Design

1. What is adaptive design?
2. The principles of adaptive design.
3. The viewport metatag.
4. Media requests.

Module 8. Flexbox

1. What is Flexbox?
2. Pros and cons of using flexbox.
3. Basic terms.
4. Basic properties of a flex container.
5. Baseline and cross axis.
6. Multiple lines in a flex container.
7. Application examples.

Module 9. Grid Layout

1. What is Grid Layout?
2. How to create a grid container?
3. Row, column.
4. Element positioning.

5. Grid lines.
6. The repeat function.
7. Application examples.

Module 10. Forms. Frames

1. Introduction to forms.
2. Controls of forms.
   - Buttons (Send, Cancel, etc.)
   - Flags.
   - Radio buttons.
   - Pop-up lists.
   - Text input.
   - Choosing files.
   - Hidden controls.
3. Form development with HTML.
   - The <form> element.
   - The <input> element.
   - The <button> element.
   - The <select>, <optgroup>, and <option> elements.
   - The <textarea> element.
   - The <label>.
   - Form structure: <fieldset> and <legend>.
4. HTML5 form elements.
5. Form validation using HTML5.
6. Formatting form elements with CSS.
7. Frames and their structure (theory).
   - The <iframe> tag.
   - Using frames to add external resources (YouTube, Google Maps, etc.)

Module 11. Version Control Systems

1. What is version control?
2. Why do we need version control?

3. Review of version control systems.
   - CVS
   - SVN
   - Git
   - Others.
4. Git.
   - What is Git?
   - Aims and objectives of Git?
   - Basic terms:
     - repository;
     - commit;
     - branch;
     - working directory.
   - Operations with Git:
     - installation;
     - creating a repository;
     - adding a file to the repository;
     - commiting to the repository;
     - getting the current state of the working directory;
     - branch display;
     - operations with the accumulation buffer;
     - git remote;
     - git push;
     - git pull;
     - other operations.
5. External services (GitHub).

Module 12. Bootstrap

1. What is Bootstrap?
2. Origin of Bootstrap.
3. Aims and objectives of Bootstrap.
4. Adding Bootstrap.
5. Grid system.

6. Using various Bootstrap patterns.
7. Navigation and Bootstrap.
8. Using images and videos.
9. Scrolling.
10. Animated controls.
11. Application examples.

Module 13. Materialize

1. What is Materialize?
2. Origin of Materialize.
3. Aims and objectives of Materialize.
4. What is material design?
5. Adding Materialize.
6. Grid system in Materialize.
7. The concept of Helper. Types of Helper.
8. Navigation and Materialize.
9. Using images and videos.
10. Scrolling.
11. Application examples.

Module 14. Less

1. What is Less?
2. Origin of Less.
3. Aims and objectives of Less.
4. Adding and compiling Less.
5. Variables.
6. Mixins.
7. Functions.
8. Application examples.

Module 15. Exam