

## HTML and CSS

Candidates for this exam should be able to recognize and write syntactically correct HTML and CSS, structure data using HTML elements, and create and apply styles using CSS.

Candidates are expected to have at least 150 hours of instruction or hands-on experience with HTML and CSS, be familiar with their features and capabilities, and understand how to write, debug, and maintain well-formed HTML and CSS code.

To be successful on the test, the candidate is also expected to have the following prerequisite knowledge and skills:

- 8th grade reading skills
- Basic computer skills and operating system functionality
- Critical thinking skills

### 1. HTML Fundamentals

#### 1.1 Construct markup that uses metadata elements

- script, noscript, style, link, meta tags (encoding, keywords, viewport, and description)

#### 1.2 Construct well-formed page markup

- DOCTYPE declaration, html, head, body, proper syntax, closing tags and commonly used symbols

### 2. CSS Fundamentals

#### 2.1 Analyze and implement inline styles, internal (embedded) style sheets, and external style sheets

- When to use inline styles, internal (embedded) style sheets, or external style sheets; precedence when using a combination of inline styles and style sheets

#### 2.2 Construct and analyze rule sets

- Valid syntax for the CSS rule set, selectors (class, id, elements, and pseudo-class)

### 3. Document Structure using HTML

#### 3.1 Construct and analyze markup to structure content and organize data

- Table tags (table, tr, th, td), h1-h6, p, br, hr, div, span, ul, ol, li

#### 3.2 Construct and analyze markup that uses HTML5 semantic elements

- Semantic tags (header, nav, section, article, aside, footer, details, summary, figure, caption)

#### 3.3 Construct and analyze markup that implements navigation

- target, anchor (a href), bookmark, relative vs. absolute links, navigating simple folder hierarchies, map, area

#### 3.4 Construct and analyze markup that uses form elements

- Form attributes, action, method, submission methods, input types and restrictions, select, textarea, button, option, label



# IT SPECIALIST EXAM OBJECTIVES

## 4. Multimedia Presentation using HTML

### 4.1 Construct and analyze markup that displays images

- `img` and `picture` elements and their attributes

### 4.2 Construct and analyze markup that plays video and audio

- `video`, `audio`, `track`, `source`, `iframe`

## 5. Webpage Styling using CSS

### 5.1 Construct and analyze styles that position content

- Positioning (`float`, `relative`, `absolute`, `static`, and `fixed`) `max-width`, `overflow`, `height`, `width`, `align`, `display`, `inline vs. block`, `visibility`, `box model`, `margins` and `padding`

### 5.2 Construct and analyze styles that format text

- `font-family`, `color`, `font-style`, `font-size`, `font-weight`, `font-variant`, `link colors`, `text formatting`, `text alignment`, `text decoration`, `indentation`, `line-height`, `word-wrap`, and `letter-spacing`

### 5.3 Construct and analyze styles that format backgrounds and borders

- `border-color`, `border-style`, `border-width`, `background properties`, `colors`

### 5.4 Construct and analyze styles that create a simple responsive layout

- Units of measurement (`percentages`, `pixels`, `em`, `vw`, `vh`), `viewport` and `media query`, `frameworks` and `templates`, `working with breakpoints`, `grids`

## 6. Accessibility, Readability, and Testing

### 6.1 Construct well-formed HTML and CSS markup that conforms to industry best practices

- Reusing rules and rule sets, `commenting`, `web-safe fonts`, `cross-platform usability`, `separation of structure (HTML)` and `style (CSS)`

### 6.2 Apply accessibility principles and evaluate content accessibility

- `Text alternatives`, `color contrast and usage`, `legibility of typography`, `tab order`, `text resizing`, `text hierarchy`, `translate`

### 6.3 Evaluate the structural integrity of HTML and CSS markup

- `Syntax errors`, `tag mismatch`, `cascading issues`

