HTML5 Application Development

Candidates for this exam are seeking to prove core HTML5 client application development skills. This exam focuses on using HTML5, CSS3, and JavaScript ES6 to develop client applications.

Candidates should have at least 150 hours of instruction or hands-on experience with HTML5, CSS3, and JavaScript ES6 and be familiar with the foundational concepts of those technologies.

1. Application Lifecycle Management

1.1 Describe the application lifecycle management stages

- Plan, design, develop, test, deploy, and maintain

1.2 Debug and test web apps

- Input validation errors, runtime errors, breakpoints

2. Graphics and Animation

2.1 Use the canvas element to create graphics and animations

- shape, color, line, translate/move, rotate, scale, interactivity

2.2 Use the svg element to create and display graphics

- Advantages, inline vs. referenced XML, shapes, color, SVG filter effects

2.3 Transform, style, and enhance text and graphics

- Graphics effects (rounded corners, shadows, transparency, background gradients, typography, and Web Open Font Format), 2-D and 3-D transformations (translate, scale, rotate, skew, and 3-D perspective transitions and animations), keyframes

2.4 Apply CSS filters to images

- grayscale, blur, sepia, opacity, drop-shadow, saturate

3. Forms

3.1 Construct and analyze markup that uses form elements

- datalist, fieldset, meter, legend, output

3.2 Configure input validation

- Validation attributes, pattern attribute for regular expressions, correct data type, length, required value

4. Layouts

4.1 Manage content layout, positioning, and flow by using CSS

- Content flow (inline vs. block flow), positioning of individual elements (float vs. absolute positioning), content overflow (scrolling, visible, and hidden), basic CSS styling

INFORMATION TECHNOLOGY
SPECIALIST

© 2025 Certiport, Inc. Certiport and the Certiport logo are registered trademarks of Certiport Inc. All other trademarks and registered trademarks are the property of their respective holders.

4.2 Construct layouts by using responsive design

- grid view, background-size, images, picture, viewport, responsive width, media queries

4.3 Construct flexible responsive layouts by using CSS flexbox

- flex container (flex-direction, flex-flow, flex-wrap), flex items (flex-basis, flex-grow, flex-shrink, order, flex)

4.4 Construct grid-based layouts by using CSS grid

- container, items, templates, gap

5. JavaScript Coding

5.1 Create and use custom classes

- Instantiation, properties, methods, inheritance

5.2 Perform data access by using JavaScript

- Send and receive data, transmit and parse complex objects, load and save files, XML, JSON

5.3 Construct code that responds to events by using event listeners and handlers

- Gesture events, handling multiple events, Event object, bubbling vs. cascading

5.4 Construct code that uses JavaScript APIs

- Google Charts, jQuery, Geolocation

5.5 Manage the state of an application

- Session state vs. app state, where to store state (local vs. session storage)

© 2025 Certiport, Inc. Certiport and the Certiport logo are registered trademarks of Certiport Inc. All other trademarks and registered trademarks are the property of their respective holders.