JavaScript Event Handling - Question Descriptions

File Analysis and Revision Questions

Q2 - Single Click Color Change

Files: `index2.html` + `q2.js`

Question: How would you make a paragraph element change its text color to green when clicked once? What JavaScript method would you use to select the paragraph and what event listener would you attach?

Hint: Focus on the basic click event and DOM element selection.

Q3 - Double Click Color Change

Files: `index3.html` + `q3.js`

Question: How would you modify the previous example to change the paragraph color to blue only when it's double-clicked instead of single-clicked?

Hint: Consider the difference between single and double click events.

Q5 - Input Event Handling

Files: `index5.html` + `q5.js`

Question: How would you capture and log each character as a user types into a text input field? What event would you listen for and how would you access the typed character?

Hint: Think about real-time input monitoring and event data extraction.

Q6 - Select Dropdown Handling

Files: `index6.html` + `q6.js`

Question: How would you update a paragraph element to display the currently selected option from a dropdown menu whenever the user makes a selection?

Hint: Consider the change event and accessing the selected value.

Q7 - Keyboard Event Handling

Files: `index7.html` + `q7.js`

Question: How would you display the pressed key character in an h1 element when any key is pressed on the keyboard? What special handling would you need for the spacebar?

Hint: Focus on keydown events and key detection.

Q8 - File Input Handling

Files: `index8.html` + `q8.js`

Question: How would you create a custom file upload button that triggers a hidden file input element and displays the selected file's name on the button?

Hint: Consider click event delegation and file input manipulation.

Q9 - Form Submission and Dynamic Content

Files: `index9.html` + `q9.js`

Question: How would you create a form that generates a user profile card when submitted, displaying the entered information in a styled card layout?

Hint: Think about form submission handling and dynamic DOM creation.

Q10 - Mouse Movement Tracking

Files: `index3.html` + `q10.js`

Question: How would you make a red square element follow the mouse cursor position on the screen, maintaining its center at the cursor location?

Hint: Consider mousemove events and coordinate calculation.

Q11 - Mouse Position Tracking

Files: `index11.html` + `q11.js`

Question: How would you make a red square element follow the mouse cursor position on the screen, maintaining its center at the cursor location?

Hint: Consider mousemove events and coordinate calculation.

Q12 - Key Release Detection

Files: `index12.html` + `q12.js`

Question: How would you display the character of the key that was just released in an input field?

Hint: Focus on keyup events and accessing the released key.

Q13 - Event Target Identification

Files: `index13.html` + `q13.js`

Question: How would you log event information when clicking on a red square element?

Hint: Consider click events and event object properties.

Q14 - Navigation Click Handling

Files: `index14.html` + `q14.js`

Question: How would you display an alert message when clicking on a navigation element?

Hint: Think about click events and alert functionality.

Q15 - List Item Toggle Styling

Files: `index15.html` + `q15.js`

Question: How would you toggle a line-through style on list items when they are clicked?

Hint: Consider click events and CSS class manipulation.

Q16 - Event Bubbling Demonstration

Files: `index16.html` + `q16.js`

Question: How would you demonstrate event bubbling by logging messages when clicking nested elements (button, inner div, outer div)?

Hint: Consider click events on nested elements and event propagation.

Q17 - Event Capture and Bubbling

Files: `index17.html` + `q17.js`

Question: How would you demonstrate both event capture and bubbling phases by logging messages in different orders when clicking nested elements?

Hint: Consider event phases and the use of the capture parameter in addEventListener.

Q18 - Character Counter

Files: `index18.html` + `q18.js`

Question: How would you create a character counter that shows how many characters are left (out of 20) as the user types in an input field, and change the color to red when the limit is exceeded?

Hint: Focus on input events and dynamic text/content updates.

Revision Strategy

For each question:

1. **Recall the core concept** - What type of event is being handled?
2. **Remember the DOM selection** - How do you select the target element(s)?
3. **Identify the event listener** - What event type and callback function are used?
4. **Understand the logic** - What happens when the event is triggered?
5. **Consider edge cases** - Are there any special conditions or error handling?

Use these questions to test your understanding of JavaScript event handling patterns and DOM manipulation techniques.