

DOM - Advanced Coding Questions

1. Using `getElementById()`

1. Write a program that selects an element by ID, changes its content, and appends another element dynamically beneath it.
-

2. Using `getElementsByName()`

2. Create a function that selects multiple elements by their `name` attribute and removes all their background colors.
-

3. Using `getElementsByTagName()`

3. Write a program that selects all `<a>` tags and modifies their `href` attributes to redirect to a new URL.
-

4. Using `getElementsByClassName()`

4. Create a function that changes the background color of every element with the class `.highlight`, based on the index of the element in the collection.
-

5. Using `querySelector()`

5. Write a program that uses `querySelector()` to select the first element matching the `.active` class and then adds a border around it.
-

6. Using `querySelectorAll()`

6. Write a program that selects all `` elements with the class `completed`, and changes their font size to 18px.
-

7. Traversing Parent Elements

7. Create a function that selects a button and traverses upward to find the parent `div` element, changing its background color to blue.
-

8. Traversing Child Elements

8. Write a program that selects the first child element of a parent and appends a new `<p>` element inside it with the text "New Child Added".
-

9. Traversing Sibling Elements

9. Create a function that selects an element, and traverses to its previous sibling and next sibling, logging both to the console.
-

10. Traversing All Siblings

10. Write a program that traverses all siblings of an element and appends a new element to each sibling's parent node.
-

11. Using `createElement()`

11. Create a program that dynamically creates a `<div>` and inserts it before an existing element with the class `.container`.
-

12. Using `appendChild()`

12. Write a program that dynamically creates an unordered list and appends it to the body of the document, adding 5 list items inside it.
-

13. Using `innerHTML`

13. Create a program that sets the `innerHTML` of a container element to include both `div` and `span` elements with specific content inside.
-

14. Using `textContent`

14. Write a program that sets the `textContent` of a `p` tag to display the current date in `YYYY-MM-DD` format.
-

15. Using `after()`

15. Write a program that adds a new `<p>` element after a heading (`<h1>`) with the text "Added After H1".
-

16. Using `prepend()`

16. Write a function that prepends a new `strong` element with the text "Important" at the beginning of a container.
-

17. Using `insertAdjacentHTML()`

17. Write a program that uses `insertAdjacentHTML()` to insert a new section of content before the footer element.
-

18. Using `replaceChild()`

18. Write a program that replaces an old image with a new one inside a container.

19. Using `cloneNode()`

19. Write a function that clones an entire `<article>` element and appends it to another parent element.

20. Using `removeChild()`

20. Write a program that removes a specific child element from the DOM when a button is clicked.

21. Using `insertBefore()`

21. Write a program that inserts a new list item before the last item in an existing list of items.

22. Attribute Manipulation: `getAttribute()`

22. Write a program that reads the `src` attribute of all images in a gallery and logs them to the console.

23. Attribute Manipulation: `setAttribute()`

23. Write a program that updates the `href` attribute of a link to a new URL when clicked.

24. Attribute Manipulation: `hasAttribute()`

24. Write a program that checks if an element with a specific class has the `data-id` attribute and alerts the result.
-

25. Attribute Manipulation: `removeAttribute()`

25. Write a program that removes a `disabled` attribute from all buttons in a form when a reset button is clicked.
-

26. Style Manipulation: `style` Property

26. Create a program that changes the background color of an element to a random color on each button click.
-

27. Style Manipulation: `cssText`

27. Write a function that changes multiple styles (like `font-size`, `color`, `background`) for a single element using `cssText`.
-

28. Style Manipulation: `getComputedStyle()`

28. Write a program that logs the computed `height` and `width` of a given element to the console.
-

29. Class Manipulation: `className` Property

29. Write a program that toggles a class `.highlighted` on all `<p>` tags when a button is clicked.
-

30. Class Manipulation: `classList` Property

30. Write a function that adds the class `.active` to an element and removes it after 2 seconds using `classList`.
-

31. Event Handling: Bubbling and Capturing

31. Create an example that demonstrates both event bubbling and capturing by logging events at each phase.
-

32. Event Handling: DOM Level 0

32. Write a program that attaches an event handler to a button using the DOM Level 0 method (`onclick`), and logs a message when clicked.
-

33. Event Handling: `addEventListener()`

33. Create a function that listens for the `click` event on a button and logs the event object.
-

34. Event Handling: `removeEventListener()`

34. Write a program that listens for a `click` event and then removes the event listener after it is triggered once.
-

35. Event Handling: Event Objects

35. Create a function that logs the mouse coordinates whenever a `click` event occurs on a page.
-

36. Event Handling: Mouse Events

36. Write a program that changes the background color of a div when the mouse is moved over it (use `mouseover` and `mouseout`).
-

37. Event Handling: Keyboard Events

37. Write a function that logs the key pressed whenever the user types inside an input field.
-

38. Event Handling: Focus Events

38. Create a program that logs a message when an input field gains or loses focus (`focus` and `blur`).
-

39. Event Handling: Input Events

39. Write a program that dynamically displays the value of an input field as the user types it (use `input` event).
-

40. Event Handling: Submit Event

40. Create a form submission handler that prevents the default action and logs the values of the input fields.
-

41. Event Handling: Touch Events

41. Write a program that detects a touch event on mobile devices and changes the color of the element being touched.
-

42. Using `setInterval()` and `clearInterval()`

42. Write a program that creates a countdown timer using `setInterval()` and stops it using `clearInterval()` once the timer reaches zero.
-

43. Using `setTimeout()`

43. Write a program that changes the content of an element after a 3-second delay using `setTimeout()`.
-

44. Handling Multiple Event Types

44. Write a function that listens for both `click` and `dblclick` events on an element and logs a message based on the type of event.
-

45. Event Delegation for Dynamic Elements

45. Create a program that uses event delegation to listen for clicks on dynamically added list items and logs the text of the clicked item.
-

46. Modifying `data-*` Attributes

46. Write a program that adds a `data-id` attribute to an element dynamically and then logs the value of that attribute when clicked.
-

47. DOM Manipulation with Dynamic User Input

47. Create a program that allows the user to input an item name, and dynamically adds it to a list.
-

48. Creating and Removing Dynamic Elements

48. Write a function that creates a new element, appends it to a parent, and removes it after a delay of 5 seconds.
-

49. Using `requestAnimationFrame()`

49. Write a program that animates a box moving across the screen using `requestAnimationFrame()`.
-

50. Creating a Dynamic Form

50. Write a function that dynamically creates a form with multiple input fields, submits the form, and logs the input values.
-

51. Cloning Multiple Nodes

51. Write a program that clones a set of elements and appends the clones at the end of the body.
-

52. Detecting Changes to DOM (MutationObserver)

52. Write a program that uses `MutationObserver` to watch for changes in the DOM and logs them to the console.
-

53. Preventing Propagation of an Event

53. Create a program that prevents the propagation of a `click` event from a child element to its parent.
-

54. Handling Form Submission Using DOM

54. Write a form handler that prevents form submission if any required fields are empty.
-

55. Using `focus()` and `blur()` Methods

55. Write a program that focuses an input field automatically when the page loads, and logs a message when the input loses focus.
-

56. Dynamically Changing Styles Based on User Interaction

56. Create a program where clicking a button changes the background color of a container based on user input.
-

57. Using `Element.scrollToView()`

57. Write a function that scrolls an element into view when it is clicked.
-

58. Advanced Event Handling with Event Delegation

58. Write a program that uses event delegation to handle click events on dynamically created list items inside a container.
-

59. Handling Multiple Events on a Single Element

59. Write a function that handles multiple events like `click`, `mouseover`, and `mouseout` on a single element and applies different actions to each.
-

60. Creating a Complex Interactive Page with DOM Manipulation

60. Create a complex interactive page (e.g., a to-do list) that allows the user to add, remove, and edit items dynamically.

61. DOM Traversing: Accessing Multiple Levels of Parents

61. Write a program that selects a child element and traverses up the DOM tree to access multiple levels of parents, logging their `className` properties.
-

62. Manipulating Multiple Elements Simultaneously

62. Write a function that changes the `textContent` of all `<h2>` elements inside a container with the class `.section`.
-

63. Using `hasChildNodes()`

63. Write a program that checks if a `div` has any child elements using the `hasChildNodes()` method and alerts the result.
-

64. Advanced Class Manipulation with `classList.toggle()`

64. Write a program that uses `classList.toggle()` to add or remove a class `.active` based on whether a button is pressed an even or odd number of times.
-

65. Remove All Attributes from an Element

65. Write a program that removes all attributes from an element with the class `.remove-attributes` when a button is clicked.
-

66. Clone an Element with Deep Copy

66. Create a program that clones an element along with all its child elements and appends the clone to the body.
-

67. Using `setAttribute()` with Custom Attributes

67. Write a program that adds a custom attribute `data-status="active"` to all elements with the class `.status` and logs the attribute value.
-

68. Modifying Multiple CSS Properties

68. Write a program that dynamically changes the `color`, `border`, and `font-size` of an element by applying the CSS properties in an object.
-

69. Implementing Dynamic Content Search

69. Write a function that listens for keypresses in an input field and dynamically filters a list of items, showing only those that match the input text.
-

70. Event Delegation with `addEventListener()`

70. Create a program that listens for click events on any list item inside a `ul`, and logs the text content of the clicked item.
-

71. Dynamically Create a Table and Append Rows

71. Write a function that dynamically generates a table and adds five rows, each containing two columns.
-

72. Working with `DOMContentLoaded` Event

72. Write a program that listens for the `DOMContentLoaded` event and logs a message once the DOM is fully loaded.
-

73. Modify Styles of a Parent Element Dynamically

73. Write a program that changes the `border` of a parent element when a child element is clicked.
-

74. Add an Event Listener to Multiple Elements

74. Write a program that adds a click event listener to all buttons inside a `div` and changes their background color when clicked.
-

75. Handling Form Validation and Submitting

75. Write a program that checks if all required fields in a form are filled before submitting, and alerts the user if any are missing.
-

76. Using `document.createTextNode()`

76. Create a program that uses `document.createTextNode()` to create a new text node and appends it to a specific element in the DOM.
-

77. Dynamically Remove Class from Child Elements

77. Write a program that removes a `.highlight` class from all child elements of a container when a button is clicked.
-

78. Using `document.querySelectorAll()` to Modify Multiple Elements

78. Write a program that selects all elements with the class `.item` and changes their font size to `20px` using `querySelectorAll()`.
-

79. Change `textContent` of Multiple Elements Dynamically

79. Write a program that allows the user to change the text of all `<p>` elements by inputting new text in a text box.
-

80. Creating a Modal Using DOM Manipulation

80. Write a function that creates a modal dialog dynamically, appends it to the body, and provides functionality to close it when the user clicks the close button.
-

81. Add New Paragraph Based on User Input

81. Write a program that allows the user to input a string and dynamically adds a new paragraph element with the input text into the DOM.
-

82. Toggle Visibility of Multiple Elements

82. Create a program that toggles the visibility of all `<div>` elements with the class `.toggle-visibility` when a button is clicked.
-

83. Using `addEventListener()` to Change Text on Hover

83. Write a function that changes the text of a `div` when the user hovers over it and resets the text when the hover ends.
-

84. Track Mouse Position and Display Coordinates

84. Write a program that tracks the mouse's **x** and **y** coordinates and displays them in real-time inside a **div** element.
-

85. Create a Dynamic Navigation Menu

85. Write a program that creates a dynamic navigation menu with links and appends it to the body. Add functionality to change the active link when clicked.
-

86. Modify **style** Using a Condition

86. Write a program that checks the **data-type** attribute of an element, and based on its value, changes the background color dynamically.
-

87. Add a New Item to a List Dynamically

87. Write a function that adds a new item to an existing list and appends it to the **ul** element.
-

88. Attach **mouseenter** and **mouseleave** Event Handlers

88. Write a program that changes the background color of a **div** when the mouse enters it (**mouseenter** event) and resets the color when the mouse leaves it (**mouseleave** event).
-

89. Apply **requestAnimationFrame()** for Smooth Animation

89. Write a program that animates a **div** to move horizontally across the screen using **requestAnimationFrame()**.

90. Create a Dynamic Countdown Timer

90. Write a program that creates a countdown timer, dynamically displaying the countdown value in an element, and stops when it reaches zero.

91. Use `MutationObserver` to Detect DOM Changes

91. Write a program that uses `MutationObserver` to detect when a new child element is added to a parent and logs the change to the console.

92. Create a Table with Sortable Columns

92. Write a program that dynamically generates a table with sortable columns, allowing the user to click on column headers to sort the data.

93. Toggle Between Two Themes Dynamically

93. Write a function that allows the user to toggle between a light and dark theme for the page by changing CSS classes dynamically.

94. Create a Simple To-Do List Application

94. Write a program that allows the user to add, remove, and mark items as completed in a to-do list, storing the list dynamically in the DOM.

95. Dynamically Change `z-index` of Elements

95. Write a program that dynamically changes the `z-index` of an element when the user clicks a button, bringing the element to the front.

96. Track Window Resize and Adjust Layout Dynamically

96. Write a function that tracks window resize events and adjusts the layout of an element based on the new window size.

97. Handle Keyboard Events for Navigation

97. Write a program that listens for **ArrowLeft** and **ArrowRight** keypresses and navigates between sections of a page.

98. Detect and Handle Clicks Outside an Element

98. Create a program that detects when a user clicks outside a specific element and hides that element.

99. Implement Infinite Scroll

99. Write a program that dynamically loads more content when the user scrolls to the bottom of a page.

100. Create a Dynamic Gallery with Filters

100. Write a program that displays a gallery of images and allows the user to filter the gallery by categories dynamically.