**JavaScript - Insane Level**

1. **What will be the output of the following?**

js

CopyEdit

let a = [1, 2, 3];

let b = a;

let c = [...a];

b.push(4);

c.pop();

console.log(a, b, c);

* + (a) [1, 2, 3, 4] [1, 2, 3, 4] [1, 2]
  + (b) [1, 2, 3] [1, 2, 3, 4] [1, 2, 3]
  + (c) [1, 2, 3, 4] [1, 2, 3, 4] [1, 2, 3]
  + (d) [1, 2, 3, 4] [1, 2, 3, 4] [1, 2, 3, 4]

1. **What does this return?**

js

CopyEdit

console.log(typeof (function() { return typeof arguments; })());

* + (a) "function"
  + (b) "object"
  + (c) "undefined"
  + (d) "number"

1. **Which of these is a correct way to create a private variable in JavaScript?**
   * (a) var \_private = 42;
   * (b) let privateVar = 42;
   * (c) function closure() { let secret = 42; return () => secret; }
   * (d) const private = new WeakMap();
2. **What does this return?**

js

CopyEdit

console.log([] == ![]);

* + (a) true
  + (b) false
  + (c) undefined
  + (d) TypeError

1. **How does JavaScript handle floating-point precision internally?**
   * (a) Uses IEEE 754 double-precision floating-point
   * (b) Uses base-10 storage
   * (c) Uses an adaptive precision algorithm
   * (d) Uses a 64-bit integer representation

**HTML - Impossible Level**

1. **What will happen if you place multiple <main> elements in an HTML5 document?**
   * (a) The page will not load
   * (b) The browser will ignore all but the first <main>
   * (c) The browser will treat it as invalid but still render
   * (d) The browser will merge them
2. **What does the shadowRoot API do?**
   * (a) Allows styling elements without affecting global CSS
   * (b) Prevents JavaScript from accessing elements
   * (c) Enables WebSockets
   * (d) Controls browser rendering order
3. **Which of these statements is FALSE about <canvas>?**
   * (a) It does not support event listeners
   * (b) It can be used for real-time graphics rendering
   * (c) It does not support child elements
   * (d) It cannot be styled with CSS
4. **What does the <isindex> tag do?**
   * (a) Defines a search field (obsolete)
   * (b) Embeds a database query
   * (c) Creates an indexable heading
   * (d) Renders an unordered list
5. **What does the sandbox attribute in <iframe> do?**
   * (a) Prevents cross-origin requests
   * (b) Blocks all JavaScript execution
   * (c) Disables all networking
   * (d) Enables WebRTC connections

**CSS - Insane Level**

1. **What will be the final computed height of the following element?**

css

CopyEdit

div {

height: 100px;

padding: 20px;

border: 10px solid black;

box-sizing: border-box;

}

* + (a) 100px
  + (b) 140px
  + (c) 120px
  + (d) 160px

1. **What does this CSS property do?**

css

CopyEdit

mix-blend-mode: difference;

* + (a) Creates a color inversion effect
  + (b) Blends images like Photoshop layers
  + (c) Changes contrast dynamically
  + (d) Only works in canvas elements

1. **How does content-visibility: auto; improve performance?**
   * (a) Prevents rendering of offscreen elements
   * (b) Reduces memory usage
   * (c) Enables lazy-loading for text
   * (d) Forces GPU acceleration
2. **What does contain: layout; do in CSS?**
   * (a) Restricts layout calculations
   * (b) Applies a flex container
   * (c) Prevents element resizing
   * (d) Disables animations
3. **Which of the following is NOT a valid CSS display property?**
   * (a) inline-grid
   * (b) run-in
   * (c) subgrid
   * (d) outside-flex

**React.js - Nightmare Level**

1. **What is the output of this React code?**

jsx

CopyEdit

function App() {

const [count, setCount] = React.useState(0);

useEffect(() => {

setCount((prev) => prev + 1);

}, []);

return <div>{count}</div>;

}

* + (a) 0
  + (b) 1
  + (c) Infinite loop
  + (d) undefined

1. **Why does React use a key in lists?**
   * (a) To optimize the diffing algorithm
   * (b) To enable event delegation
   * (c) To allow rendering in multiple browsers
   * x(d) To disable re-rendering
2. **What does React.memo() do?**
   * (a) Prevents unnecessary re-renders
   * (b) Optimizes event handlers
   * (c) Creates a virtual DOM copy
   * (d) Caches JSX elements
3. **What is the purpose of useDeferredValue() in React?**
   * (a) Delays state updates for smoother UI
   * (b) Prevents concurrent rendering
   * (c) Forces immediate DOM updates
   * (d) Used for animations
4. **How does useSyncExternalStore() improve state management?**
   * (a) Allows subscribing to external stores efficiently
   * (b) Optimizes state updates
   * (c) Prevents memory leaks
   * (d) Forces a re-render on every update