✅ JavaScript Tough Assignment – 40 Questions (Coding Heavy)

🔁 Switch Statement (10 Questions — 7 Coding + 3 Conceptual)

💻 Coding Practice (Q1–Q7)

1. Write a switch-based calculator supporting `+`, `-`, `*`, `/`, `%`, and `**`.
2. Use switch to print the full name of a month when a number (1–12) is given.
3. Build a switch block to convert grades (A-F) to performance messages.
4. Create a switch that checks browser type (`Chrome`, `Firefox`, etc.) and prints support message.
5. Check whether a number is even or odd using switch and modulo `%`.
6. Nested switch: Check user role (`admin`, `user`) and their action (`create`, `edit`, `delete`).
7. Use switch with `true` to categorize age groups: Child, Teen, Adult, Senior.

🧠 Conceptual (Q8–Q10)

8. What happens when break is missing in a switch? Explain with example.
9. Can we place `default` at the top of the switch block? Show with code.
10. Explain fall-through in switch with a multi-case example.

⚙️ Higher Order Functions (15 Questions — 12 Coding + 3 Conceptual)

💻 Coding Practice (Q11–Q22)

11. Use `map()` to convert an array of numbers into their squares.

12. Use `filter()` to return only the names longer than 5 characters.
13. Use `reduce()` to find the sum of all odd numbers in an array.
14. Implement your own version of `map()` called `customMap()`.
15. Create a function `repeat(fn, n)` to call any function `n` times.
16. Use `forEach()` to print the index and value of each array element.
17. Use `reduce()` to convert an array of strings into a sentence.
18. Filter out all palindromes from a string array.
19. Map through an array of numbers and return strings like “Even” or “Odd”.
20. Use `reduce()` to count how many times each number appears in an array.
21. Create a function that accepts another function and applies delay using `setTimeout()`.
22. Filter an array of user objects (with `name`, `age`) to return only adults (age > 18).

🧠 Conceptual (Q23–Q25)

23. What's the difference between `map()` and `forEach()`? Show code example.
24. Why is `reduce()` called powerful in functional programming? Explain with one-liner code.
25. Explain what makes a function “Higher Order” with real code example.

🧩 Object Methods and `this` (15 Questions — 11 Coding + 4 Conceptual)

💻 Coding Practice (Q26–Q36)

26. Create a `bankAccount` object with `deposit()` and `withdraw()` methods using `this`.

27. Write an object with `name` and a method `sayHello()` that uses `this` correctly.
28. Use `Object.keys()` to print only the keys of an object dynamically.
29. Use `Object.values()` to sum all values of a numeric object.
30. Merge two objects using `Object.assign()` and print the merged result.
31. Freeze an object using `Object.freeze()` and try modifying it — verify the result.
32. Seal an object using `Object.seal()` and test adding & updating keys.
33. Check if a key exists in an object using `hasOwnProperty()`.
34. Build an object with custom methods: `add()`, `subtract()`, and call them.
35. Create a `student` object with method `getDetails()` that returns a formatted string.
36. Make a method that removes keys with `null` or `undefined` values from an object.

🧠 **Conceptual (Q37–Q40)**

37. Explain the difference between `Object.freeze()` and `Object.seal()` with examples.
38. What happens if we use arrow function as object method and access `this`? Show output.
39. Show an example where method is passed as callback and loses `this` context.
40. Explain `Object.entries()` and its use in converting object to array of key-value pairs.