JavaScript learning path

Perfect 💯 — another round coming up!
This time I'll make the **questions trickier** — mixing **var/let/const**, block scope, TDZ, and hoisting — exactly the kind of traps interviewers love.

⚡ Round 2 — Practice Questions

📝 Q1. Hoisting + Shadowing

```js
var a = 5;
function test() {
  console.log(a);
  var a = 10;
  console.log(a);
}
test();
```

📝 Q2. Const & Reassignment

```js
const arr = [1, 2, 3];
arr.push(4);
console.log(arr);

arr = [5, 6, 7];
console.log(arr);
```

📝 Q3. TDZ Trick

```js
```

```
{
  console.log(x);
  let x = 100;
}
```

📝 Q4. Global vs Block Scope

```js
var x = 1;
let y = 2;

{
  var x = 10;
  let y = 20;
  console.log(x, y);
}

console.log(x, y);
```

📝 Q5. Function Hoisting with Var

```js
sayHi();
var sayHi = function() {
  console.log("Hi!");
};
```

📝 Q6. Loop Trap Again

```js
for (var i = 0; i < 3; i++) {
  setTimeout(() => console.log(i), 100);
}
```

👉 Try to **predict outputs** for each (Q1–Q6).
Then in the **next chat** I'll solve them with full **memory creation + execution simulation** (like an engine).

Would you like me to **add 2–3 interview-level brain-benders** in this round as well (like mixing `var` + `function` + `TDZ`), or keep it at this level?

Perfect 🚀 Let's spice this up with some **interview-level brain-benders** after the standard Q1–Q6. These are the kind of questions that make candidates sweat because they require **internal working knowledge (hoisting, scope, TDZ, closures, memory references, etc.)**.

⚡ Round 2 (Extended) — With Brain-Benders

📝 Q7. Function vs Var Hoisting Clash

```js
console.log(foo);
function foo() { return "I am a function"; }
var foo = "I am a variable";
console.log(foo);
```

📝 Q8. Closure + var/let

```js
function counter() {
  for (var i = 0; i < 3; i++) {
    setTimeout(() => console.log("var:", i), i * 100);
  }

  for (let j = 0; j < 3; j++) {
    setTimeout(() => console.log("let:", j), j * 100);
  }
}
counter();
```

📝 Q9. TDZ + Function Parameter

```js
function test(x = y, y = 2) {
  console.log(x, y);
}
test();
```

📝 Q10. const Object Mutability

```js
const obj = { name: "JS" };
obj.name = "JavaScript";
console.log(obj);

Object.freeze(obj);
obj.name = "ECMAScript";
console.log(obj);
```

✅ So now you have **10 questions (Q1–Q10)**:

- Q1–Q6 = strong basics (var/let/const, scope, hoisting, TDZ, loops).
- Q7–Q10 = **brain-benders** (function vs var hoisting, closures, default params, const objects).

👉 Your task: Try to **predict the output for all 10** (don't worry if you get stuck — that's the point).
👉 In the **next chat**, I'll solve them with **full engine simulation (memory phase + execution phase)** and extra **interview notes**.

Do you want me to give the **solutions for all 10 at once** in the next step, or do you prefer I solve them in **two rounds (Q1–Q6 first, Q7–Q10 next)** so it’s easier to digest?