### 1. What will be the output of the following code?

var a = 10;  
if (true) {  
 var a = 20;  
}  
console.log(a);

* 1. 10
  2. 20
  3. undefined
  4. Error

### 2. What will be the output of the following code?

let a = 10;  
if (true) {  
 let a = 20;  
}  
console.log(a);

* 1. 10
  2. 20
  3. undefined
  4. Error

### 3. In the following code, which scope does the variable x belong to?

function foo() {  
 var x = 100;  
 if (true) {  
 let y = 200;  
 }  
}

* 1. Block scope
  2. Function scope
  3. Global scope
  4. Undefined

### 4. What will be the output of the following code?

const a = 5;  
{  
 const a = 10;  
 console.log(a);  
}  
console.log(a);

* 1. 5, 5
  2. 10, 5
  3. 10, 10
  4. 5, 10

### 5. What will be the output of the following code?

var x = 1;  
function test() {  
 var x = 2;  
 if (true) {  
 var x = 3;  
 }  
 console.log(x);  
}  
test();

* 1. 1
  2. 2
  3. 3
  4. undefined

### 6. Which of the following statements about JavaScript variable scopes is correct?

* 1. Variables declared with var have block scope.
  2. Variables declared with let and const have block scope.
  3. Variables declared with let have function scope.
  4. Variables declared inside functions have global scope.

### 7. What will be the output of the following code?

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

* 1. 10
  2. 20
  3. undefined
  4. Error

### 8. What is the output of the following code?

let a = 5;  
{  
 let a = 10;  
 console.log(a);  
}  
console.log(a);

* 1. 10, 10
  2. 10, 5
  3. 5, 5
  4. 5, 10

### 9. What is the scope of a variable declared with let inside a for loop?

for (let i = 0; i < 3; i++) {  
 console.log(i);  
}  
console.log(i);

* 1. Global scope
  2. Block scope
  3. Function scope
  4. Undefined

### 10. What is the scope of a variable declared with var inside a for loop?

for (var i = 0; i < 3; i++) {  
 console.log(i);  
}  
console.log(i);

* 1. Global scope
  2. Block scope
  3. Function scope
  4. Undefined

### 11. Which scope does the variable name have in the following code?

let name = "John";  
function greet() {  
 console.log(name);  
}  
greet();

* 1. Block scope
  2. Global scope
  3. Function scope
  4. Undefined

### 12. What is the result of the following code?

var x = 10;  
function test() {  
 var x = 20;  
 console.log(x);  
}  
test();  
console.log(x);

* 1. 20, 10
  2. 10, 20
  3. 20, 20
  4. 10, 10

### 13. What will the following code output?

let x = 5;  
function outer() {  
 let x = 10;  
 function inner() {  
 let x = 15;  
 console.log(x);  
 }  
 inner();  
}  
outer();

* 1. 5
  2. 10
  3. 15
  4. Error

### 14. In which scope is the variable y accessible in the following code?

function test() {  
 let y = 30;  
 if (true) {  
 let y = 40;  
 }  
 console.log(y);  
}  
test();

* 1. Global scope
  2. Block scope
  3. Function scope
  4. Undefined

### 15. Which scope does the variable a have in the following code?

let a = 100;  
if (true) {  
 let a = 200;  
 console.log(a);  
}  
console.log(a);

* 1. Global scope, Block scope
  2. Block scope, Global scope
  3. Global scope, Global scope
  4. Block scope, Block scope

### 16. What will be the output of the following code?

var a = 5;  
function myFunction() {  
 if (true) {  
 var a = 10;  
 }  
 console.log(a);  
}  
myFunction();

* 1. 10
  2. 5
  3. undefined
  4. Error

### 17. What will be the output of this code?

let x = 5;  
function outer() {  
 const x = 10;  
 function inner() {  
 console.log(x);  
 }  
 inner();  
}  
outer();

* 1. 5
  2. 10
  3. undefined
  4. Error

### 18. What is the scope of i in the following code?

for (let i = 0; i < 5; i++) {  
 console.log(i);  
}  
console.log(i);

* 1. Block scope
  2. Function scope
  3. Global scope
  4. Undefined

### 19. What is the scope of a variable declared with const inside a block?

{  
 const a = 10;  
 console.log(a);  
}  
console.log(a);

* 1. Block scope
  2. Global scope
  3. Function scope
  4. Undefined

### 20. What will the following code output?

let x = 100;  
{  
 const x = 50;  
 console.log(x);  
}  
console.log(x);

* 1. 100, 100
  2. 50, 100
  3. 50, 50
  4. 100, 50

### 21. In the following code, where is the variable x accessible?

let x = "global";  
function test() {  
 let x = "function";  
 if (true) {  
 let x = "block";  
 console.log(x);  
 }  
}  
test();  
console.log(x);

* 1. Block scope, Function scope
  2. Global scope, Function scope
  3. Global scope, Block scope
  4. Block scope, Global scope

### 22. What will be the output of this code?

function test() {  
 var x = 1;  
 if (true) {  
 var x = 2;  
 console.log(x);  
 }  
}  
test();

* 1. 1
  2. 2
  3. undefined
  4. Error

### 23. What is the scope of a variable declared inside a function?

function foo() {  
 let z = 30;  
}  
console.log(z);

* 1. Global scope
  2. Function scope
  3. Block scope
  4. Undefined

### 24. In which scope is the variable y in the following code?

let x = 100;  
if (true) {  
 let y = 200;  
 console.log(y);  
}

* 1. Block scope
  2. Global scope
  3. Function scope
  4. Undefined

### 25. Which of the following will result in an error?

let a = 5;  
const a = 10;

* 1. No error
  2. Syntax error
  3. Runtime error
  4. Reference error

### 26. What will be the output of this code?

function foo() {  
 let x = 20;  
 return function() {  
 console.log(x);  
 };  
}  
const bar = foo();  
bar();

* 1. undefined
  2. 20
  3. Error
  4. ReferenceError

### 27. What will be the output of the following code?

{  
 let a = 10;  
 const b = 20;  
 console.log(a, b);  
}  
console.log(a, b);

* 1. 10, 20, 10, 20
  2. 10, 20, undefined, undefined
  3. 10, 20, Error
  4. 10, 20, undefined, ReferenceError

### 28. What is the result of this code?

let a = 10;  
function test() {  
 let a = 20;  
 {  
 let a = 30;  
 console.log(a);  
 }  
 console.log(a);  
}  
test();  
console.log(a);

* 1. 30, 20, 10
  2. 30, 30, 10
  3. 30, 20, 20
  4. 30, 20, Error

### 29. In which scope is the variable x available in the following code?

function outer() {  
 let x = 10;  
 {  
 let x = 20;  
 {  
 let x = 30;  
 console.log(x);  
 }  
 }  
}  
outer();

* 1. Global scope
  2. Function scope
  3. Block scope
  4. Undefined

### 30. What is the output of this code?

let x = 50;  
function test() {  
 let x = 100;  
 {  
 const x = 200;  
 console.log(x);  
 }  
 console.log(x);  
}  
test();  
console.log(x);

* 1. 200, 100, 50
  2. 200, 100, Error
  3. 200, 200, 50
  4. 200, 100, 100