**Q1. What will happen when this code is executed?**

var x = 5;

{

var x = 10;

}

console.log(x);

* A) 5
* B) 10
* C) undefined
* D) ReferenceError

**Q2. What is the output of the following code?**

let a = 20;

{

let a = 30;

}

console.log(a);

* A) 20
* B) 30
* C) undefined
* D) ReferenceError

**Q3. What will the following code print?**

const y = 50;

{

const y = 100;

}

console.log(y);

* A) 50
* B) 100
* C) undefined
* D) Error

**Q4. What is the value of result?**

let a = 10;

{

let b = 20;

var result = a + b;

}

console.log(result);

* A) 30
* B) undefined
* C) Error
* D) NaN

**Q5. What is the output of the following?**

{

let x = 10;

}

console.log(x);

* A) 10
* B) undefined
* C) Error
* D) null

**Q6. Which variable declaration allows redeclaration in the same scope?**

var a = 5;

var a = 10;

* A) var
* B) let
* C) const
* D) None of the above

**Q7. What does this code log?**

let a = 10;

a = 20;

const b = 30;

b = 40;

console.log(a, b);

* A) 20 40
* B) 20 30
* C) Error
* D) undefined

**Q8. What will happen here?**

console.log(a);

let a = 5;

* A) 5
* B) undefined
* C) Error
* D) null

**Q9. What is the output?**

{

const x = 5;

{

const x = 10;

console.log(x);

}

console.log(x);

}

* A) 10, 5
* B) 5, 10
* C) Error
* D) 10, 10

**Q11. What will the following code output?**

{

var a = 10;

let a = 20;

console.log(a);

}

* A) 10
* B) 20
* C) undefined
* D) SyntaxError

**Q12. What does this log to the console?**

let x = 10;

function test() {

let x = 20;

{

let x = 30;

console.log(x);

}

console.log(x);

}

test();

* A) 30 20
* B) 10 30
* C) 20 30
* D) 30 10

**Q13. What is the result of this code?**

console.log(typeof undeclaredVar);

* A) "undefined"
* B) "object"
* C) ReferenceError
* D) null

**Q14. What is the output?**

let a = 5;

{

var a = 10;

}

console.log(a);

* A) 10
* B) 5
* C) undefined
* D) SyntaxError

**Q15. What is logged to the console?**

function outer() {

var x = 5;

function inner() {

console.log(x);

}

inner();

}

outer();

* A) 5
* B) undefined
* C) null
* D) Error

**Q16. What happens when this code runs?**

console.log(b);

var b = 15;

* A) 15
* B) undefined
* C) ReferenceError
* D) NaN

**Q17. What is the behavior of this code?**

{

const a = 5;

a = 10;

}

* A) 10
* B) 5
* C) Error
* D) undefined

**Q18. Which keyword respects block scope fully?**

* A) var
* B) let
* C) const
* D) Both B and C

**Q19. What will this code output?**

let a = 5;

{

let a = a + 1;

console.log(a);

}

* A) 6
* B) undefined
* C) ReferenceError
* D) NaN

**Q20. What does this log?**

var x = 1;

{

var x = 2;

console.log(x);

}

console.log(x);

* A) 2, 1
* B) 1, 2
* C) 2, 2
* D) undefined, 2