1.

```
function greet() {
    console.log("Hello, World!");
}
greet();
```

What will be the output?

A) Hello, World!

B) Undefined

C) Error

D) Nothing

2.

```
function sayHi(name) {
    console.log("Hi, " + name);
}
sayHi("John");
```

What will be the output?

A) Hi, John

B) Hi, undefined

C) Error

D) Nothing

3.

```
function multiply(a, b) {
    console.log(a * b);
}
multiply(2, 3);
```

What will be the output?

A) 5

B) 6

C) Undefined

D) Error

4.

```
function subtract(a, b) {
    console.log(a - b);
}
subtract(5);
```

What will be the output?

A) NaN

B) 5

C) Undefined

D) Error

5.

```
function add(a, b = 2) {
    console.log(a + b);
}
add(3);
```

What will be the output?

A) 5

B) 3

C) Undefined

D) NaN

6.

```
function showMessage() {
    console.log("Hello!");
```

```
}
showMessage();
```

What will be the output?

A) Hello!

B) Undefined

C) Error

D) NaN

7.

```
function printValue(a) {
    console.log(a);
}
printValue();
```

What will be the output?

A) Undefined

B) Error

C) Null

D) NaN

8.

```
function concatStrings(a, b) {
    console.log(a + b);
}
concatStrings("Hello", "World");
```

What will be the output?

A) HelloWorld

B) Hello World

C) Undefined

D) NaN

9.

```
function greet() {
    console.log("Hi!");
}
function callGreet() {
    greet();
}
callGreet();
```

What will be the output?

A) Hi!

B) Undefined

C) Error

D) Nothing

10.

```
function double(a) {
    console.log(a * 2);
}
double(4);
double();
```

What will be the output?

A) 8 and NaN

B) 8 and Undefined

C) Error

D) NaN

11.

```
function outer() {
    console.log("Outer function");
    function inner() {
        console.log("Inner function");
    }
```

```
    inner();
}
outer();
```

What will be the output?

A) Outer function
Inner function

B) Outer function

C) Inner function

D) Error

12.

```
function calculate(a, b, c) {
    console.log(a + b * c);
}
calculate(2, 3, 4);
```

What will be the output?

A) 14

B) 20

C) 10

D) NaN

13.

```
function logValue(x) {
    console.log(x);
}
logValue();
logValue(5);
```

What will be the output?

A) Undefined, 5

B) NaN, 5

C) 5, Undefined

D) Error

14.

```
function funOne() {
    console.log("funOne started");
    funTwo();
}
function funTwo() {
    console.log("funTwo executed");
}
funOne();
```

What will be the output?

A) funOne started
funTwo executed

B) funOne started

C) funTwo executed

D) Error

15.

```
function test(a, b) {
    console.log(a + b);
}
test(4);
```

What will be the output?

A) NaN

B) 4

C) Undefined

D) Error

16.

```
function callInner() {
    function inner() {
        console.log("Inner function called");
    }
    inner();
}
callInner();
```

What will be the output?

A) Inner function called

B) Undefined

C) Error

D) Nothing

17.

```
function outer(a) {
    function inner(b) {
        console.log(a + b);
    }
    inner(3);
}
outer(7);
```

What will be the output?

A) 10

B) NaN

C) Error

D) Undefined

18.

```
function calculate(x) {
    console.log(x * x);
}
calculate(0);
```

What will be the output?

A) 0

B) 1

C) Undefined

D) Error

19.

```
function funOne(a, b) {
    console.log(a + b);
    funTwo();
}
function funTwo() {
    console.log("Function Two executed");
}
funOne(2, 3);
```

What will be the output?

A) 5
Function Two executed

B) 5

C) Function Two executed

D) Error

20.

```
function display(a) {
    console.log("Value is: " + a);
}
display();
```

What will be the output?

A) Value is: undefined

B) Error

C) Nothing

D) NaN

21.

```
function funOne(a, b, c) {
    console.log("funOne started");
    console.log(a + b + c);
    funTwo(a, c);
}
function funTwo(a, b) {
    console.log(a * b);
    funThree(a - b);
}
function funThree(x) {
    console.log("funThree executed");
    console.log(x + 5);
}
funOne(2, 3, 4);
```

What will be the output?

A) funOne started, 9, 8, funThree executed, 3

B) funOne started, 8, 6, funThree executed, 2

C) funOne started, 9, 8, funThree executed, 7

D) Error

22.

```
function calculate(a, b) {
    console.log("Calculation begins");
    sum(a, b);
}
function sum(x, y) {
    console.log("Sum is:", x + y);
    difference(x, y);
}
function difference(x, y) {
    console.log("Difference is:", x - y);
}
calculate(5, 3);
```

What will be the output?

A) Calculation begins, Sum is: 8, Difference is: 2

B) Calculation begins, Sum is: 2, Difference is: 8

C) Calculation begins, Difference is: 2, Sum is: 8

D) Error

23.

```
function outer(a) {
    console.log("Outer function: " + a);
    function inner(b){
        console.log("Inner function: " + b);
        nested(b * 2); 8*2 16
    }
    function nested(c) {
        console.log("Nested function: " + c);
    }
    inner(a + 3);
}
outer(5);
```

What will be the output?

A) Outer function: 5, Inner function: 8, Nested function: 16

B) Outer function: 5, Inner function: 8, Nested function: 11

C) Outer function: 5, Inner function: 3, Nested function: 6

D) Error

24.

```
function main(a) {
    console.log("Main started");
    secondary(a * 2);
}
function secondary(x) {
    console.log("Secondary value:", x);
    helper(x + 3);
}
function helper(y) {
    console.log("Helper value:", y);
}
main(4);
```

What will be the output?

A) Main started, Secondary value: 8, Helper value: 11

B) Main started, Secondary value: 4, Helper value: 7

C) Main started, Helper value: 11, Secondary value: 8

D) Error

25.

```
function taskOne(a, b) {
    console.log("Task one started");
    taskTwo(a - b);
}
function taskTwo(x) {
    console.log("Task two value:", x);
    taskThree(x * 3);
}
function taskThree(y) {
    console.log("Task three result:", y);
}
taskOne(10, 2);
```

What will be the output?

A) Task one started, Task two value: 8, Task three result: 24

B) Task one started, Task two value: 12, Task three result: 36

C) Task one started, Task three result: 24, Task two value: 8

D) Error

26.

```
function compute(a) {
    console.log("Compute started");
    square(a);
}
function square(x) {
    console.log("Square:", x * x);
    cube(x);
}
function cube(y) {
    console.log("Cube:", y * y * y);
}
compute(3);
```

What will be the output?

A) Compute started, Square: 9, Cube: 27

B) Compute started, Square: 27, Cube: 9

C) Compute started, Cube: 9, Square: 27

D) Error

27.

```
function first(a, b) {
    console.log("First function");
    second(a + b);
}
function second(c) {
    console.log("Second function with value:", c);
    third(c * 2);
}
function third(d) {
    console.log("Third function result:", d);
}
first(2, 3);
```

What will be the output?

A) First function, Second function with value: 5, Third function result: 10

B) First function, Second function with value: 6, Third function result: 12

C) First function, Second function with value: 5, Third function result: 8

D) Error

28.

```
function funMain(a) {
    console.log("Main function value:", a);
    function funSub1(b) {
        console.log("Sub-function 1 value:", b);
        funSub2(b + 2);
    }
    function funSub2(c) {
        console.log("Sub-function 2 value:", c);
    }
    funSub1(a * 2);
```

```
}
funMain(3);
```

What will be the output?

A) Main function value: 3, Sub-function 1 value: 6, Sub-function 2 value: 8

B) Main function value: 3, Sub-function 1 value: 5, Sub-function 2 value: 7

C) Main function value: 3, Sub-function 1 value: 6, Sub-function 2 value: 7

D) Error

29.

```
function outer(a, b) {
    console.log("Outer function started");
    function middle(x) {
        console.log("Middle function:", x);
        inner(x - 1);
    }
    function inner(y) {
        console.log("Inner function:", y);
    }
    middle(a * b);
}
outer(3, 4);
```

What will be the output?

A) Outer function started, Middle function: 12, Inner function: 11

B) Outer function started, Middle function: 7, Inner function: 6

C) Outer function started, Middle function: 12, Inner function: 6

D) Error

30.

```
function start(a) {
    console.log("Start function with value:", a);
    function process(b) {
        console.log("Process function with value:", b);
        finalize(b + 5);
    }
    function finalize(c) {
```

```
        console.log("Finalize function result:", c);
    }
    process(a * 3);
}
start(2);
```

What will be the output?

A) Start function with value: 2, Process function with value: 6, Finalize function result: 11

B) Start function with valuhe: 2, Process function with value: 5, Finalize function result: 10

C) Start function with value: 2, Process function with value: 6, Finalize function result: 10

D) Error
