FRONT END – PART 2
STARTING WITH JAVASCRIPT

(What can Javascript do?)

1. What can we achieve using Javascript?

a) WEBSITES **b) PRESENTATIONS**

c) GAMES **d) MOBILE APPS**

(Initial Name for JS)

2. What was the initial name for Javascript?

a) Script b) Livescript

c) Mocha d) None of the above

SOLUTION DESCRIPTION:

In September 1995, a Netscape programmer named Brandan Eich developed a new scripting language in just 10 days. It was originally named Mocha, but quickly became known as LiveScript and, later, JavaScript.

(ECMA Official Release)

3. When was ECMAscript officially released?

a) 1997 b) 1995

c) 1999 d) 2001

(Invalid Variable Name)

4. Which of the following variable name is not valid in JavaScript?

a) var myname = "John"; **b) var my name = "John";**

c) var myName = "John"; d) var my_name = "John";

SOLUTION DESCRIPTION:

You can't have a space between the words in your variable names.

(Basic Data Types)

5. Which of the following are basic data types in Javascript?

a) integer **b) number** **c) null** d) float

e) string f) character **g) undefined** **h) object**

(Declare an Integer)

6. How can you declare an integer?

a) int a = 10; b) number a = 10;

c) var a = 10; d) integer a = 10;

(Error on Execution)

7. Find out that if any of the lines below will produce an error on execution?

```
A) var a = 10;
B) console.log(a)
C) a = " 'Coding Ninjas' ";
D) console.log(a);
```

a) A

b) B

c) C

d) D

e) No error

(Print Local Var)

8. What will be printed on the console from the following code -

```
var a;
console.log(a);
```

a) 0

b) undefined

c) null

d) Show error - 'a is not defined'

(Integer Minus Char)

10. What will be printed on the console from the following code -

```
var a = "A";
console.log(65 – a);
```

a) 0

b) undefined

c) null

d) NaN

SOLUTION DESCRIPTION:

When we try to operate on a number with a non-numeric value, it will give the result as NAN.

(Type of Negative)

11. What will the below statement print on the console?

```
Typeof("-1")
```

a) string

b) number

c) 'string'

d) "number"

(Type of Nothing)

12. What is the below line print on the console?

```
var a = null
typeof(a)
```

a) "null"

b) "string"

c) "undefined"

d) "object"

(Find Output)

13. What will be the output of the following code?

```
var x = 4;
var y = x++;
y += 1;
```

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

a) 6 **b) 5** c) 7 d) 8

SOLUTION DESCRIPTION:

In this line: var y = x++ the value of x is assigned to y before x is incremented,

so y equals 4 on line 2, while x equals 5. Therefore on line 3, y now equals 5 instead of 6.

(Evaluate Arithmetic Expression)

14. What will be output of the statement given below?

```
console.log(20 + 12 * 2 - 10 / 2)
```

a) 39

b) 27

c) -16

d) None of the above

SOLUTION DESCRIPTION:

The order of execution of operators i.e. operator precedence is 'multiplication', 'division', 'addition' and then 'subtraction'.

(Integer Plus Char)

15. What will the below line print on the console?

```
console.log(1 + '1')
```

a) 98

b) 11

c) 2

d) Show error - 'TypeMismatch'

(Var Plus Char)

16. What will the below line print on the console?

```
var a;
console.log(a + "b");
```

a) undefined

b) ab

c) b

d) undefinedb

(Number Minus Char)

17. What will the below line print on the console?

```
console.log(1 – '1')
```

a) 98

b) 1

c) 0

d) Show error - 'NaN'

(String Equals Integer)

18. What will be the output for the below comparison in Javascript?

```
"1" == 1
```

a) 1

b) 0

c) true

d) false

(Null and Undefined)

19. What will be the output for the below comparison in Javascript?

```
null == undefined
```

a) true

b) false

c) null

d) undefined

SOLUTION DESCRIPTION:

"null == undefined" means null is equal to undefined but not identical. In JavaScript, when you declare a variable without assigning any value. (that value is by default "undefined"), but When you assign a variable to a null value, then we are trying to convey that the variable is empty.

So "null" and "undefined" both mean "nothing", so having them be equal makes intuitive sense.

(If Negative)

20. What will the below code produce on the console?

```
if( -1 ) {
    console.log("true")
}
else {
    console.log("false")
}
```

a) true

b) false

c) error is shown

d) none of the above

(If Empty String)

21. What will the below code produce on the console?

```
if( "" ) {
    console.log("true")
}
else {
    console.log("false")
}
```

a) true

b) false

c) "true"

d) "false"

(For Loop)

22. Which of the following 'for' loops would not produce any error?

```
A) for(var i=0; i<5; ++i) {
      console.log("Hello") ;
   }

B) var i , j ;
   for(i=0, j = 10; i < 10, j < 100) {
   console.log("Hello");
   }

C) var i ;
   for(i=0; i<5) {
     console.log("Hello");
   }

D) var i , j ;
   for(i=0, j=10; i<5; ++i) {
      console.log("Hello");
   }
```

a) A b) B c) C d) D

(Print at Least Once)

23. Which of the following statements will print 'hii' at least once?

```
A) var i=10 ;
   do {
       console.log("hii");
     } while(i<5);

B) var i=0 ;
   while(i<5); {
      console.log("hii");
      ++i ;
   }

C) for(i=0, j=10; i<5; ++i) {
       if(i<5)
         break;
       console.log("hii");
   }

D) var i=0 ;
   while(i<5) {
      console.log("hii");
      ++i;
    }
```

a) A b) B c) C **d) D**

(Find the Output)

24. What will be the output of the following code:

```
var a = 2;
var b = 0;
while(a <= 4){
    a++;
    b += a * 2;
    console.log(b);
}
```

a) 4 10 18

b) 6 14 24

c) 2 3 4

d) None of the above.

SOLUTION DESCRIPTION:

The loop will run 3 times, before meeting the exit condition.

First value of b will be 3 * 2 = 6, followed by 6 + 4 * 2 = 14, and then value of 14 + 5 * 2 = 24.

ASSIGNMENT

(Javascript Possibilities)

25. What all is possible using Javascript?

a) Add styles to the web page

b) Add structure to the web page

c) Add pop ups to the web page

d) Add animation to web page

e) Add HTML code to web page

f) Add click events to web page

(Javascript Possibilities)

26. Which of the following will display an alert box on web page?

a) window.alert();

b) alert();

c) both of the above

d) none of the above

(Ten by Zero)

27. What will be printed on the console from the following code -

```
var a = 10/0;
console.log(a);
```

a) 0

b) undefined

c) Infinity

d) null

(Null Compare Undefined)

28. What will be the output of the following statements on console?

```
console.log(null == undefined);
console.log(typeof(null) == typeof(undefined));
```

a) true true

b) true false

c) false true

d) false false

SOLUTION DESCRIPTION:

console.log(typeof(null) == typeof(undefined)) output will be false because In javaScript the type of “null” is object while type of “undefined” is undefined. So both are not equal.

(Integer Plus String)

29. What will be printed by the JS code below?

```
console.log(65 + "H")
```

a) 65H

b) AH

c) 137

d) Show error – 'TypeMismatch'

SOLUTION DESCRIPTION:

JavaScript converts the number 65 into a string. During the addition of a numeric type (65) and a string type ('H'), the number is treated as a string.

We can concatenate strings like "Hello" + "World", so what's happening here is "65" + "H" which returns "65H".

(Null Equal Undefined)

30. What will the output shown on console from the following code?

```
null === undefined
```

a) true

b) false

c) null

d) undefined

(What is the Type)

31. What will the below statement print on the console?

```
typeof( typeof( typeof( 100 ) ) )
```

a) "number"

b) "integer"

c) "string"

d) none of the above

SOLUTION DESCRIPTION:

typeof(100) will produce "number" as output in string form. The other two will produce "string" as output.

(Conditional)

32. What will the below code produce on the console?

```
var a;
if( typeof(a) ) {
    console.log("true")
}
else {
    console.log("false")
}
```

a) true

b) false

c) error is shown

d) none of the above

SOLUTION DESCRIPTION:

typeof(a) return undefined as a String and a non-empty string returns true

(Ternary Operator)

33. Which of the following syntax is correct for Javascript Ternary?

a) condition? exprIfTrue ; exprIfFalse

b) condition ? exprIfTrue : exprIfFalse

c) condition : exprIfTrue ? ExprIfFalse

d) condition ? exprIfFalse : exprIfTrue

(Switch Syntax)

34. In Switch statement syntax, the expression is compared with the case labels using the following operator -

```
switch(expression)
{
    statements
}
```

a) ==

b) equals

c) ===

d) equal

****