Javascript Quiz 3

1. What is the output?

```
function f(){ return 20; }
new f() instanceof f;
```

a) true b) false

2. What is the output?

```
var foo = {
    bar: function(){ return this.baz; },
    baz: 1
}
typeof (f = foo.bar)();
typeof foo.bar();
```

a) “number”, “number” c) “undefined”, “number”
b) “function”, “number” d) “undefined”, “undefined”

3. What is the output of following code when execute in console ?

```
function foo() {
  return this === window;
}();
```

a) true b) false c) Error

4. Choose correct option ?

```
function foo(name) {
   this.name = "sai";
   this.name = name;
   return { name: "vasu" }
}
var bar = new foo;
console.log(new foo().name);
console.log(new foo('naveen').name);
console.log(bar.name);
```

a) vasu, naveen, sai c) naveen, naveen, undefined
b) vasu, vasu, error d) vasu, vasu, vasu

5. What is the output?

```
var result = function foo() {
  return this === window;
}();
console.log(result);
```

a) true b) false c) Error

6. What is the output ?

```
function Actor(name) {
   this.name = name;
   this.industry = 'Telugu';
   this.getName = function() {
       console.log("I am " + this.name + " from " + this.industry + " Industry.";
   }
}
var a = Actor("mahesh");
typeof a;
```

a) “string” b) “object” c) “function” d) “undefined”

7. What is the return value?

```
function f1(){var a = 1; f2();}
function f2(){return a;}
f1();
```

a) undefined b) 1 c) ReferenceError

8. Can we change **window** and **document** objects like this ?

```
window = {};
document = 20;
```

a) Yes b) No

9. What are the alerted values ?

```
var x = 3;
var foo = {
    x: 2,
    baz: {
        x: 1,
        bar: function() {
            return this.x;
        }
    }
}
var go = foo.baz.bar;
alert(go());
alert(foo.baz.bar());
```

a) 1, 2 b) 1, 3 c) 2, 1 d) 2, 3 e) 3, 1

10. What is the output ?

```
var x = 3;
var elem = document.getElementById('gbw');     // Assume #gbw is available in DOM
function foo() {
       var x = 2;
       this.val = 10;
       function print(e){
              console.log(this.x, this.val, x);
       }
       elem.addEventListener('click', print);
}
foo();
elem.click();
```

a) 3, 10, 2 b) 2, 10, 3 c) undefined, undefined, 2 d) 3, undefined, 2

11. What is the output ?

```
var x = 3;
var elem = document.getElementById('gbw');     // Assume #gbw is available in DOM
function foo() {
       var x = 2;
       this.val = 10;
       function print(e){
              console.log(this.x, this.val, x);
       }
       elem.addEventListener('click', print.bind(this))
}
foo();
elem.click();
```

a) 3, 10, 2 b) 2, 10, 3 c) 4, 10, 2 d) undefined, undefined, 2

12. What is the final link in prototype chain.

a) null b) Object.prototype c) Function.prototype d) No final link

13. Is prototype property of any function enumerable? What is the output ?

```
function person() {
   this.location = 'hyd';
}
person.name = 'Sudeep';
person.prototype.designation = "Developer"
person.age = 25;
for(var k in person) {
    console.log(person[k]);
}
```

a) hyd, Sudeep, 25 b) hyd c) hyd, age d) Sudeep, Developer, 25 e) 25

14. Fill in the following blanks.

```
function Person(name){
       this.name = name;
       this.location = "hyd";
       return {
       };
}
var p1 = new Person('Naresh');
p1.name             = _____________
p1.constructor      = _____________
p1.__proto__  = _____________
```

15. What is the output ?

```
typeof a1.__proto__
```

a) "function" b) "object" c) "string" d) "undefined"

16. What are the values printed in console ?

```
function person(name) {
       this.name = name;
       this.location = 'Kerala';
}
person.prototype.officeLocation = 'hyd';
var p1 = new person('Jinu');
for(var k in p1) {
       console.log(p1[k]);
}
```

a) Jinu, Kerala, hyd b) Jinu, Kerala c) Error

17. What the value of this ?

```
function abc() {
       console.log(this);
}
abc();
```

a) undefined b) document object c) window d) null

18. Name the invocation pattern ?

```
function xyz() {
       console.log(this);  // Name the invocation pattern
}
function foo(a) {
       this.name = a;
       xyz();
}
var a = new foo(10);
```

a) Constructor b) Method c) Function d) Apply

19. What is the output ?

```
var name = "Google";
var obj = {
	name: 'divami',
	getName: function() {
		return this.name;
	}
}
var c = { name: 'Microsoft'};
c.d = obj.getName;
c.d();
```

a) divami b) Google c) Microsoft

20. Name the invocation pattern(s) in the below code ?

```
var name = "Google";
var obj = {
	name: 'divami',
	getName: function() {
		return this.name;
	}
}
obj.getName();		// Invocation pattern 1
var x = obj.getName;		// Invocation pattern 2
x();
```

a) Method, Function
b) Function, Function
c) Method, Method
d) Function, Method

21. What is the console output ?

```
var value = 100;
var obj = {
	value: 10,
	getValue: function() {
		return this.value
	},
	setValue: function(v) {
		this.value = v;
	},
	double: function() {
		var helper = function() {
			this.value = this.value * 2;
		}
		helper();
	}
}
obj.setValue(30);
obj.double();
var go = obj.double;
go();
console.log(obj.getValue());
console.log(value);
```

a) 20, 200 b) 60, 200 c) 30, 400 d) 120, 100

22. Fill in the following blank.

```
function Foo(who){
	this.name = who;
}
var a1 = new Foo();
a1._______  == Foo.prototype
```

a) [[Prototype]] b)__proto__ c) prototype d) Both a & b

23. What is the output ?

```
var name = "google";
var myObj = {
       name: 'divami',
       loc: 'hyd'
}
function getName() {
       console.log(this.name);
}
var obj = {
       name: "Informatica",
       getName: function() {
              return this.name;
       }
}

var myGetName1 = getName.bind(myObj);
var myGetName2 = obj.getName.bind(myObj);
getName();          _________________
obj.getName();      _________________
myGetName1();       _________________
myGetName2();       _________________
```

24. What is the value of arguments array ?

```
function sum(a, b) {
       console.log(arguments);
}
go = sum.bind(null, 3, 4, 5);
go(1,2);
```

a) [1, 2, 3, 4, 5] b) [3, 4, 5] c) [3, 4, 5, 1, 2] d) [1, 2]

25. What is the output ?

```
function employee(name, profession) {
       console.log(this);
}
employee.call(null, "Sumathi", "Architect");
```

a) null b) undefined c) window d) Error

26. What is the output ?

```
var value = 100;
function calc() {
    this.value = 10,
    this.getValue = function() {
       setTimeout(function() {
          console.log(this.value);
       }, 2000);
    }
}
var r = new calc();
r.getValue();
```

a) 100 b) 10 c) undefined d) Error

27. If the 26th program output is not **"10"** then how can I get **"10"** ? Change the program accordingly.

Ans:

28. Modify the function such that it should reverse the parameters. Use Array prototype reverse function.

```
function change() {
      // Code here
}
change(1,2,3,4);          // Should return [4,3,2,1]
change(9,8,7);            // Should return [7,8,9]
```

Ans:

29. What is the output ?

```
function abc() {
      console.log('divami');
}
var a1 = new abc();

a1 instanceof abc         ??
a1 instanceof Object            ??
```

a) false, true b) true, false c) true, true d) false, false

30. What are the 4 steps when we create object with **'new'** keyword?

31. What is "dunder proto" ?

Ans:

32. What is the output ?

```
<div class="name"> Name1 </div>
<div class="name"> Name2 </div>
var name = 'Divami';
function Class(){
      var name = "Software";
      function type(e){
            $(this.currentTarget).html(name);
      }
      return type;
}
function name(e){
      $(this.currentTarget).html(name);
}
$('.name')[0].on('click', name);
$('.name')[1].on('click', Class());

$('.name')[0].click(); _________________
$('.name')[1].click(); _________________
```

33. What is the output ?

```
function countdown (num) {
    for (var i = 0; i <= num; i += 1) {
        setTimeout(function () {
            console.log(num - i);
        }, i * 2000);
    }
}
countdown(5); ____________________
```

34. What is the problem in above question? Please give us the solution.

Ans:

35. What is the return value ?

```
(function f(f){
        return typeof f();
})(function(){ return 1; });
```

a) “number” b) “undefined” c) “function” d) Error

36. What is the return value ?

```
var foo = {
        bar: function() { return this.baz; },
        baz: 1
};
(function(){
        return typeof arguments[0]();
})(foo.bar);
```

a) “undefined” b) “object” c) “number” d) “function”

37. What is the output ?

```
var f = (function f(){ return "1"; }, function g(){ return 2; })();
typeof f;
```

a) “undefined” b) “number” c) “string” d) “function”

38. What is the output ?

```
var foo = (function() {
        var x = {
                bar: "bar"
        };
        return {obj: x};
})();
typeof foo;
foo.obj.bar;
```

a) “function”, undefined b) “object”, bar c) “object”, undefined

39. What is the output & which function is closure ?

```
var bar = “bar100”;
function foo() {
        var bar = "bar";
        return function baz() {
                console.log(bar);
        }
}
function bam() {
        foo()();
}
bam();
```

a) bar100, baz b) bar, baz c) bar, bam d) bar, foo

40. What is clouser? Write different ways to create closures ?

41. How to reverse a string without **for loop** and **split()** function ?

Ans:

42. How to check the variable is array or not ? Write all possible solutions.

Ans:

31. Draw prototype chain diagram for the following code.

```
/***** Start *****/
function Shape(id, x, y) {
    this.id = id;
    this.move(x, y);
};
Shape.prototype.move = function (x, y) {
    this.x = x;
    this.y = y;
};
function Rectangle(id, x, y, width, height) {
    Shape.call(this, id, x, y);
    this.width  = width;
    this.height = height;
};
Rectangle.prototype = Object.create(Shape.prototype);
/***** End *****/

var r1 = new Rectangle(1, 10, 20, 200, 100);
r1.constructor = _____ ??
```