1)

var funcs=[];

for(let i=0;i<5;i++){

console.log(i);

funcs.push(function(){

console.log(i);

});

}

funcs[0]();

funcs[1]();

funcs[2]();

2)

function foo(item){

var myPrivateItem = item;

this.getPrivateData = function(){

return myPrivateItem;

};

}

var Tom = new foo('Some private data.');

console.log(Tom.myPrivateItem); //undefined

console.log(Tom.getPrivateData()); //Some private data.



3)

function foo1()

{

return {

bar: "hello"

};

}

function foo2()

{

return

{

bar: "hello"

};

}

console.log("foo1 returns:");

console.log(foo1());

console.log("foo2 returns:");

console.log(foo2());

op is undefined in one case why ?



4)  
function buildFunctions2() {

var arr = [];

for (var i = 0; i < 3; i++) {

var j = i;

arr.push(

(function(j) {

return function() {console.log(j);}

})(i)

);

}

return arr;

}

var fs2 = buildFunctions2();

fs2[0]();

fs2[7]();

fs2[8]();



5)

function foo1()

{

  return

  {

      bar: "hello"

  };

}

function foo2()

{

  return {

      bar: "hello"

  };

}

console.log("foo1 returns:");

console.log(foo1());

console.log("foo2 returns:");

console.log(foo2());



6)

function foo(outer\_arg){

  function inner(inner\_arg){

    return outer\_arg+inner\_arg;

  }

  return inner;

}

var get\_function = foo(3);

console.log(get\_function(4));



7)

function outer()

{

    function create\_Closure(val)

    {

        return function()

        {

            return val;

        }

    }

    var arr = [];

    var i;

    for (i = 0; i < 4; i++)

    {

        arr[i] = create\_Closure(i);

    }

    return arr;

}

var get\_arr = outer();

console.log(get\_arr[0]());

console.log(get\_arr[1]());

console.log(get\_arr[2]());

console.log(get\_arr[3]());



8)

var a={},

    b={key:'b'},

    c={key:'c'};

a[b]=123;

a[c]=456;

console.log(a[b]);



9)

var length = 10;

function fn() {

                console.log(this.length);

}

var obj = {

  length: 5,

  method: function(fn) {

    fn();

    arguments[0]();

  }

};

obj.method(fn, 1);



10)

function person () {

  var name = 'Paul';  // Local variable

  var actions = {

    speak: function () {

         console.log('My name is ', name) ;

    }

  };

  return actions;

}

person().speak() ;



11)

const arr = [10, 12, 15, 21];

for (var i = 0; i < arr.length; i++) {

  console.log(i);

  setTimeout(function() {

    console.log('Index: ' + i + ', element: ' + arr[i]);

  }, 3000);

}



12)

(function(){

  var a = b = 3;

})();

console.log(typeof a);

console.log(typeof b);

console.log("a defined? " + (typeof a !== 'undefined'));

console.log("b defined? " + (typeof b !== 'undefined'));

console.log(1 +  "2" + "2");

console.log(1 +  +"2" + "2");



13)

(function() {

    console.log(1);

    setTimeout(function(){console.log(2)}, 1000);

    setTimeout(function(){console.log(3)}, 0);

    console.log(4);

})();