

# es6 相关

1. 下面这个例子最外面会输出10 如何解决

```
for(var i = 0; i < 10; i++) {  
  console.log(1);  
}  
console.log(i);
```

2. 使用结构赋值，实现两个变量的值的交换
3. 使用结构赋值，完成函数的参数默认值
4. 利用数组推导，计算出数组 [1,2,3,4] 每一个元素的平方并组成新的数组。
5. 使用模板字符串改写下面的代码。（ES5 to ES6改写题）

```
let name = "前端研究所";  
let obj = {  
  "name": name,  
  "say": function(){  
    alert('hello world');  
  }  
};
```

6. 用箭头函数的形式改写下面的代码。（ES5 to ES6改写题）

```
arr.forEach(function(v,i) {  
  console.log(i);  
  console.log(v);  
});
```

7. 设计一个对象，键名的类型至少包含一个symbol类型，并且实现遍历所有key。
8. 有一本书的属性为：{"name": "《ES6基础系列》", "price": 56}; 要求使用Proxy对象对其进行拦截处理，name属性对外为"《ES6入门到懵逼》", price属性为只读。（练习题）
9. 阅读下面的代码，并用for...of改成它。（ES5 to ES6改写题）
10. 关于Set结构，阅读下面的代码，回答问题。（代码阅读题）。

```
let s = new Set();  
s.add([1]);  
s.add([1]);  
console.log(s.size);
```

问：打印出来的size的值是多少？为什么

答：

11. 关于Map结构，阅读下面的代码，回答问题。（代码阅读题）

```
let map = new Map();
map.set([1], "ES6系列");
let con = map.get([1]);
console.log(con);
```

问：打印出来的变量con的值是多少，为什么？

答：

12. 11题的如何得到的变量con的值就是：“ES6系列”。
13. 通过class 定义一个类Animal，通过传参初始化它的类型，如：“猫科类”。它有一个实例方法：run，run函数体内容可自行定义。
14. 基于上一题的Animal类，定义一个子类Cat并继承Animal类。初始化Cat类的昵称name和年龄age。并拥有实例方法eat，eat函数体内容可自行定义。
15. 利用module模块，实现两个模块A和B，A模块导出变量name，age和say方法。B模块只导入变量name和say方法，并且重命名name为nickname。
16. 使用字符串模板 把下面代码补充完整

```
const greetings = (name) => {
  return `hello    `; //要补充的部分
}
```

17. 使用es6语法 实现下图代码功能

```
// **ES5**
var Person = function(first, last) {
  if (!(this instanceof Person)) {
    throw new Error("Person is a constructor function, use new with it");
  }
  this.first = first;
  this.last = last;
};

Person.prototype.personMethod = function() {
  return "Result from personMethod: this.first = " + this.first + ", this.last = " + this.last;
};

var Employee = function(first, last, position) {
  if (!(this instanceof Employee)) {
    throw new Error("Employee is a constructor function, use new with it");
  }
  Person.call(this, first, last);
  this.position = position;
};
Employee.prototype = Object.create(Person.prototype);
Employee.prototype.constructor = Employee;
Employee.prototype.personMethod = function() {
  var result = Person.prototype.personMethod.call(this);
  return result + ", this.position = " + this.position;
};
Employee.prototype.employeeMethod = function() {
  // ...
};
```

18. 冻结上面17题中定义的对象 使其无法被修改 ( 温馨提示可以用 `Object.freeze()` ) 简单了解 `Object.freeze()`的用法

参考 [https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Global\\_Objects/Object/freeze](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Global_Objects/Object/freeze)

19. 解释一下 `Object.freeze()` 和 `const` 的区别

20. 说明下列方法为何不能用作 IIFE , 要使其成为 IIFE , 需要进行哪些更改 ?

```
function foo(){ }();
```