继承、ajax

2018年7月5日 星期一 14:29

1、昨日内容回顾

1、原型对象

构造函数有一个属性 prototype ，指向的是构造函数的原型

实例化对象有一个属性 __proto__，指向的是构造函数的原型

构造函数的原型有一个属性 constructor，指向的是构造函数

2、原型链

当对象访问属性或方法时，先从自身查找，找不到就去构造函数的原型对象上找，如果找不到，就去原型对象的原型对象上找，直到 Object.prototype.__proto__ == null. 如果没找到，返回 undefined。

2、继承

javascript 继承基本实现是以原型链实现的。

对象 A 继承对象 B

1、原型链继承

1、把对象 A 的原型指向对象 B 的原型

A.prototype = B.prototype;

缺点 ： 1、不能继承构造函数上的属性和方法

2、不能向构造函数内传参

2、把对 A 的原型指向 实例化对象 b

A.prototype = new B();

缺点 ： 不能像构造函数内传参

把原型对象的 constructor 属性，指回A 的构造函数

A.prototype.constructor = A;

2、修改构造函数继承

```
function A(){
    B.apply(this,arguments);
}
function B(name,age){
    this.name = name;
    this.age = age;
}
```

相当于构造函数A在构造函数 B 里面初始化所有的属性。

优点 ： 可以向构造函数内传参

缺点 ： 不能继承原型上的属性和方法

3、原型链 + 构造函数

```
function A(name,age,like){
    this.like = like;
    B.call(this, name,age);
}

function B(name,age){
    this.name = name;
    this.age = age;
}

B.prototype.eat = function(){
    console.log(1);
}
```

```
};

A.prototype = B.prototype;
A.prototype.constructor = A;

var a = new A("lisi",10,"code");
console.log(a);
a.eat();
```

作业：

1、使用原型链继承 ， 对象 Dog 继承对象 Cat

2、使用构造函数继承， 对象 Study 继承 对象 Teacher

3、使用原型链+构造函数继承 ，对象 son 继承对象 father.

4、对象 son 继承 对象father，对象 father 继承 对象 grandfather。

5、把今天日考技能做完。