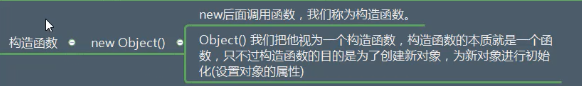
## 构造函数产生对象：

### 介绍：



### 例:

<script>

//构造函数

function person(){

//基类

var obj=new Object();

//定义属性

obj.name=null;

obj.age=null;

obj.sex=null;

//定义方法

obj.stydy=function(){

console.log(this.name+'在学习');

};

obj.sleep=function(){

console.log(this.name+'在睡觉');

};

//这里必须return才可以

return obj;

}

//实例化对象,每次实例化的对象是不一样的

var p1=person();

p1.name="张三";

p1.age=18;

p1.sex="男";

p1.stydy();

p1.sleep();

console.log(p1);

var p2=person();

p2.name="李四";

p2.age=20;

p2.sex="女";

p2.stydy();

p2.sleep();

console.log(p2);

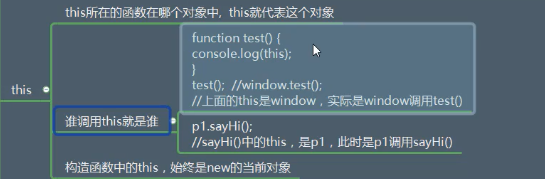
</script>

### this：重要

this就是实例化对象，比如上面例子中的

var p1=person();

所以这个时候this是p1



## 改造后的例子：

<script>

//构造函数

function Person(name,age,sex){

//基类

// var obj=new Object();

//定义属性

this.name=name;

this.age=age;

this.sex=sex;

//定义方法

this.eat=function(someThing){

console.log(this.name+'在吃'+someThing);

};

this.run=function(someWhere){

console.log(this.name+'跑'+someWhere);

};

//这里必须return才可以

// return obj;

}

//实例化对象,这里是new person

var p1=new Person("张三",18,"男");

p1.eat('奶');

p1.run('公园');

console.log(p1);

</script>