计算机生成了可选文字:
/ * ， Created by wwtliu on 14 一 8 一 4 · 
3var person—I 
age:3ø, 
'function()( 
e a t 
alert("RPE") 
alert(person.name); 

也可以后续person.height=100;实现增加属性。

也可以通过函数构造器来创建对象：

计算机生成了可选文字:
Person 
Person. prototype = { 
age:3ø, 
'functxon()( 
e a t ． 
Var 
alert( ' 我 在 吃 鬥 
new Person( ); 

p.name="A";

p.eat();

2.对象的继承：

计算机生成了可选文字:
function People （ 
People. prototype. 
Say 
ale rt 
{ ' he110 ， ); 
function Student()( 
function 0 { 
Student · prototype 
new People(); 
new Student 0 ： 
S.Say 

javascript中的每个对象都有prototype属性，Javascript中对象的prototype属性的解释是：返回对象类型原型的引用。

A.prototype = new B();

理解prototype不应把它和继承混淆。A的prototype为B的一个实例，可以理解A将B中的方法和属性全部克隆了一遍。A能使用B的方法和属性。这里强调的是克隆而不是继承。可以出现这种情况：A的prototype是B的实例，同时B的prototype也是A的实例。

计算机生成了可选文字:
function People()l 
People. prototype. 
Say 
alert( ' he110 ， ， ); 
function Student()( 
function 0 { 
Student · prototype 
new People(); 
function( ， { 
Student ． prototype ． 
、 a Y 
alert( 'stu—heUo' ); 
“ new Studentl( ， 
s 0 

调用父类的方法：

计算机生成了可选文字:
function Peopte()l 
People. prototype. 
Say 
ale rt 
{ ' he110 ， ); 
function Student()( 
function 0 { 
Student · prototype 
new Peopte(); 
var u ， ， 义 = Student ． prototype ． 
Say 
Student ． prototype ． say 
function() 
call(this) ： 
superSsay ， 
ale rt "stu—heUo") ； 
new Student 0 ； 
sa y 0 

传递参数：

计算机生成了可选文字:
function People ， 
this ． name 一 
function 0 { 
People. prototype. 
Say 
alert( 'peo—heUo"+this ． name); 
function Student( ， { 
this ， name n ： 
Student · prototype 
new People(); 
var = Student ． prototype ． 
Say. 
Student ． prototype ． say 
function() { 
call(this) ： 
superSsay. 
alert( "stu—heUo"+thxs ． name) 
new Student (' 矿 ); 
sa 以 ， 

例子：

E:\开发\个人\HtmlStudy\HTMLStudy\src\Object：





