1.javascript对象object:封装了属性、方法(每个对象有自己的各不干扰)

var fun=function(){}

fun.name="fun";

var fun2=new fun();

fun2.name是undefined

2.与c++ javas对象类似，但c++ java对象以类为模板生成对象本身。

javascript没有类，以构造函数（constructor）作为对象的模板。

3. c++: （class关键字）定义了类型-->以class Type为模板-->对象称为实例

javascript:没有关键字，语言内置类型就是object(对象类型) Function(函数类型)-->以prototype.constructor为模板-->生成变量实例

4.javascript的\_\_proto\_\_(指向原型对象)和prototype(以此对象为原型对象)

对象（函数）fun: function fun(){}

对象A = new fun();

#1

一个对象的\_\_proto\_\_指向其原型对象。

一个对象的prototype指向一个对象，以此对象为原型，new出（复制出）新的对象。

#2

所有对象都有\_\_proto\_\_，即都有原型对象; 函数才有prototype对象，才能作为new 的操作对象。

A只有: A.\_\_proto\_\_ == fun.prototype

fun是对象: fun.\_\_proto\_\_=Function.prototype

fun是函数，且默认 fun.prototype.constructor==fun自身

#4

所有字面对象，如hand={a:"a"}, 的hand.\_\_proto\_\_==Object.prototype

#4

对象的constructor属性，A的原型对象所在的对象

A.constructor == fun

fun.constructor ==Function

#

prototype都是Object的instance

prototype的\_\_proto\_\_都指向Object.prototype。