1、7种基本数据类型：Number ,String,Boolean,Object,Null,Undefined,Sysmbol

2、装箱转换与拆箱转换：

装箱转换：将基本类型转换为对应的对象，如new String(‘hello’)

使用Object.prototype.toString可以准确识别对象对应的基本类型；

Call 本身会发生装箱操作；

拆箱操作：将对象转换为对应的基本类型，先拆箱，再转换

3、对象具有两类属性：

数据属性：

Value 属性的值

Writable 决定属性能否被fu值

Enumerable 决定for in能否枚举该属性

Configurable 决定属性能否被删除或修改

访问器属性：

Getter/setter

Enumerable/configurable

4、原型链

在一个对象中寻找某个属性，找不到会继续在这个对象的原型上找，原型也是一个对象，可以继续往原型对象的原型上寻找，直到找到object的原型，也就是undefined为止，这个寻找路径，被称为原型链

Javascript提供了一系列的内置函数，以便更直接的访问操纵原型：

Object.create 根据指定的原型创建新对象，原型可以是null

Object.getPrototypeOf 获取一个对象的原型

Object.setPrototypeOf 设置一个对象的原型

New 运算接受一个构造器和一组调用参数，做了下面几件事

以构造器的prototype属性（区分于[[prototype]]）为原型，创建新对象  
 将this和调用参数传给构造器，执行；

如果构造器返回的是对象，则返回，否则，返回第一步创建的对象；

5、javascript中的对象分类：

宿主对象：由javascript宿主环境提供的对象，他们的行为完全由宿主环境决定；

内置对象：由javascript语言提供的对象：固有对象-由标准规定，随着javascript运行时自动创建的对象实例/原生对象-可以由用户通过Array\RegExp等内置构造器或者特殊语法创建的对象/普通对象-由{}语法，Object构造器或者class关键字定义类创建的对象，它能够被原型继承。

Error:[[ErrorData]]

Boolean:[[BooleanData]]

Number:[[NumberData]]

Date:[[DateValue]]

RegExp:[[RegExpMatcher]]

Sysmbol:[[SysmbolData]]

Map:[[MapData]]