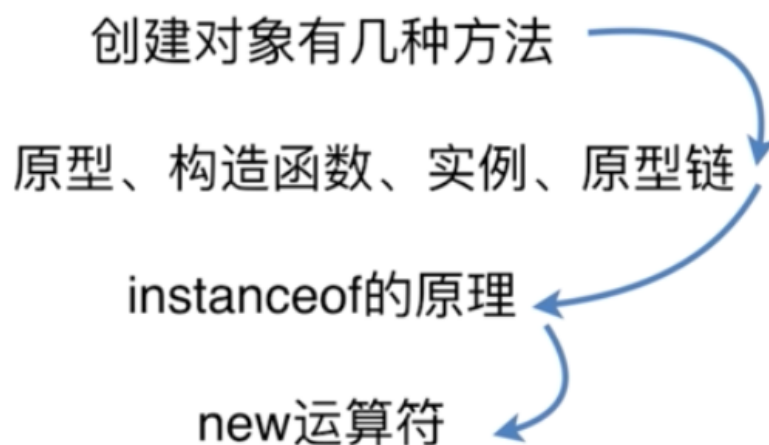


原型链类



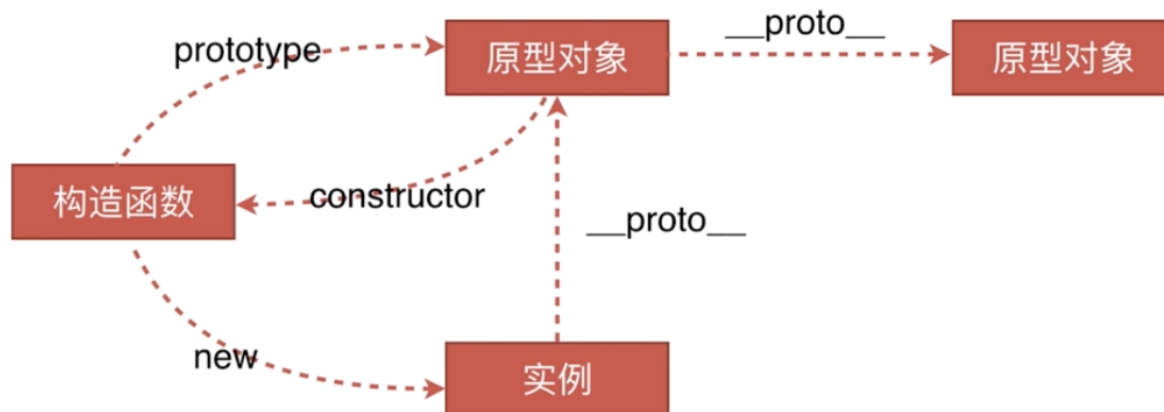
原型链类 创建对象有几种方法

```
var o1={name:'o1'};  
var o11=new Object({name:'o11'});
```

```
var M=function(){this.name='o2'}  
var o2=new M();
```

```
var P={name:'o3'};  
var o3=Object.create(P)
```

原型链类 原型、构造函数、实例、原型链



构造函数 `a()` 创建一个实例，实例的 `__proto__` 属性就是构造函数 `a()` 的原型对象 (prototype)，原型对象的 `constructor` 对象就是这个构造函数 `a()`

```
a().prototype === new a().__proto__;
```

```
a().prototype.constructor === a();
```

原型链类 instanceof



原型链类 new运算符

一个新对象被创建。它继承自foo.prototype



构造函数 foo 被执行。执行的时候，相应的传参会被传入，同时上下文(this)会被指定为这个新实例。new foo 等同于 new foo(), 只能用在传递任何参数的情况



如果构造函数返回了一个“对象”，那么这个对象会取代整个new出来的结果。如果构造函数没有返回对象，那么new出来的结果为步骤1创建的对象