

题目

说说你对JS中的原型链的理解

回答

- 思路

- 原型 (prototype)

- 是function对象的一个属性，它定义了构造函数制造出的对象的公共祖先，通过该构造函数产生的对象，可以继承该原型的属性和方法，原型也是对象

```
function Person() {}  
Person.prototype.name = '大钊';  
Person.prototype.a = function () {  
    console.log(11);  
};  
  
var person1 = new Person();  
  
console.log(person1.name);  
person1.a();
```

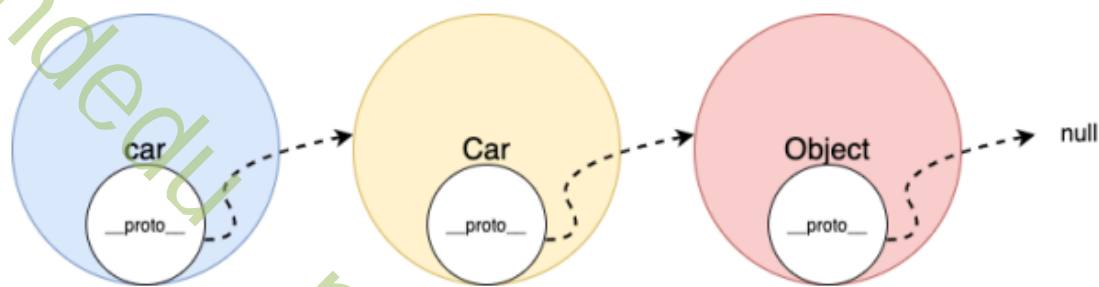
- 作用

- 构造函数实例化出来的对象可以使用公共的属性或者方法

- 函数对象才有 prototype 属性

- 原型链

- js里万物皆对象，所以一直访问 __proto__ 属性就会产生一条链条
 - 链条的尽头是null (Object.prototype.__proto__)
 - 当js引擎查找对象的属性时，会先判断对象本身是否存在该属性
 - 不存在的属性就会沿着原型链往上找



```
function Car() {}  
Car.prototype.name = '大钊';  
var car = new Car();
```

○ 总结

- 原型主要是解决继承问题
- 每个对象拥有一个原型对象，通过 `__proto__` 指针指向其原型对象，并从中继承方法和属性
- 同时原型对象也可能拥有原型，这样一层一层，最终指向 `null` (`Object.prototype.proto`指向的是`null`)
- 上述的关系被称为原型链，通过原型链一个对象可以拥有定义在其他对象中的属性和方法

