

那么问题来了，id指针指向的objc_object结构体是？

```
struct objc_object {  
    Class isa OBJC_ISA_AVAILABILITY;  
};
```

它里面包含了 Class 类型的isa指针，那么 Class 类型又是什么？这个isa又是啥玩意，别急，我们继续一步一步往下说。

- Class

```
/// An opaque type that represents an Objective-C class.  
typedef struct objc_class *Class;
```

显然，Class是一个指向了objc_class类型结构体指针。在这个结构体中包含了很多有关类的信息，对于去理解一个类的结构、信息非常重要！

objc_class 结构体中值得注意的有:

- super_class: 指向当前类的父类
- ivar: 指向objc_ivar_list结构体的指针, 可以理解为ivar的数组, 里面包含了当前类中的所有成员变量的信息, 而每一个成员变量都是objc_ivar类型的结构体, 它存储了单个成员变量的信息。
- methodLists: 指向objc_method_list结构体的二级指针, 可以理解为method的数组, 里面包含了当前类中的所有方法信息, 而每一个成员变量都是objc_method类型的结构体, 它存储了单个方法的信息。

至于objc_class结构体所包含的其他东西呢? 我们。。。就不在继续深入了, 大家有兴趣可以在如下路径中查看。

usr/include > objc > runtime.h