

- SEL:它是selector在Objc中的表示类型。selector是方法选择器，SEL的数据结构如下：

```
/// An opaque type that represents a method selector.  
typedef struct objc_selector *SEL;
```

其实SEL就是个映射到方法的字符串，通常我们在添加方法事件时，使用的是Objc编译器命令@selector(),除此之外，还可以直接通过runtime提供的 sel_registerName 函数获取对应方法映射地址SEL。

- IMP:

```
typedef id (*IMP)(id, SEL, ...);
```

它是一个函数指针，指向参数为(id, SEL,.....)返回id类型的函数。这是由编译器生成的。当你发送一个objc消息后，真正在运行时所执行的函数，正是由IMP指针所指向的。如此一来，在某些情况下，我们可以直接通过IMP指针去执行某一个函数。

- isa:

指的是一个指向objc_class的指针，表明当前class的类型。objc_class结构体中仅仅包含了Class类型的isa指针，所以形象的说isa表示一个Class对象的Class,又被称为Meta Class。当然了，元类也是对象，而且也是某个类的实例，这个类称之为 Root metaClass。剩下关于isa所指向的 Meta Class 具体内容，就不在展开讲解，大家感兴趣的话可以参考如下地址：

<http://blog.leichunfeng.com/blog/2015/04/25/objective-c-object-model/>

- Method:

```
typedef struct objc_method *Method;
```

它表示一种代表类中的某个方法的类型,而objc_method则有对应另外的结构体，它里面包含了方法名、方法类型、方法实现。如下图所示