

# KVC - By Dorayo

## 什么是KVC --->What

KVC指的就是NSKeyValueCoding非正式协议。

KVC是一种间接地访问对象的属性的机制。

这种间接表现在通过字符串来标识属性，而不是通过调用存取方法或直接地访问实例变量的方式。

KVC机制不仅支持对象，还支持标量和结构体类型，这些非对象的类型会被自动的装箱和开箱。

### Key & Key Path

键(Key)是一个字符串用来标识对象里面的一个指定的属性。一般一个键对应对象的存取方法或实例变量。键必须是ASCII码，一般以小写字母开始，不能包含空格。

A key path is a string of dot separated keys that is used to specify a sequence of object properties to traverse. The property of the first key in the sequence is relative to the receiver, and each subsequent key is evaluated relative to the value of the previous property.

键路径（Key Path）是一个由点进行分割的一系列键组成的字符串

## KVC的使用

```
- (void)setValue:(id)value forKey:(NSString *)key
- (id)valueForKey:(NSString *)key
- (void)setValue:(id)value forKeyPath:(NSString *)keyPath
- (id)valueForKeyPath:(NSString *)keyPath
- (id)valueForKey:(NSString *)key
- (void)setNilValueForKey:(NSString *)key
```

### DEMO

注意：

- KVC与点语法是正交的，也就是一点关系都没有
- The accessor patterns are written in the form -set<Key>: or -<key>

## KVC在集合中的使用

@avg

```
NSNumber *transactionAverage = [transactions valueForKeyPath:@"@avg.amount"];
```

@count

```
NSNumber *numberOfTransactions = [transactions valueForKeyPath:@"@count"];
```

@max

```
NSDate *latestDate = [transactions valueForKeyPath:@"@max.date"];
```

@min

```
NSDate *earliestDate = [transactions valueForKeyPath:@"@min.date"];
```

@sum

```
NSNumber *amountSum = [transactions valueForKeyPath:@"@sum.amount"];
```

---

## 什么是KVO

键值观察(Key-value observing)是一种机制。这种机制允许当一个对象的某些属性的值被修改后会通知另外观察该变化的对象。常用于MVC中Model与Controller之间的通信。

---

## 如何使用KVO

1. 注册，指定被观察者的属性
2. 实现回调方法
3. 移除观察

---

## 常用方法及数据类型说明

```
// 添加监听对象，并指定监听的keyPath，以及监听哪些值的变化
- (void)addObserver:(NSObject *)anObserver forKeyPath:(NSString *)keyPath
options:(NSKeyValueObservingOptions)options context:(void *)context
options参数：
    NSKeyValueObservingOptions 说明你要监听什么样的变化：
    NSKeyValueObservingOptionNew 属性变化的时候， 只会通知新的值
    NSKeyValueObservingOptionOld 属性变化的时候，只会通知旧的数据值
    NSKeyValueObservingOptionInitial 当调用添加监听的方法时候，回调方法立即调用，
通知初始值
    NSKeyValueObservingOptionPrior 取代单次通知，在值变化前和变化后都会调用，比如
设置old后，不单单只通知一次， 变化前后会通知两次。

// 移除指定键路径的观察者对象
- (void)removeObserver:(NSObject *)anObserver forKeyPath:(NSString *)keyP
ath

// 当监听的属性发生变化后会调用此方法，根据监听option的设置，change的值会存储对应的值
- (void)observeValueForKeyPath:(NSString *)keyPath ofObject:(id)object ch
ange:(NSDictionary *)change context:(void *)context
```

*注意：必须用set方法，或者KVC的方法修改属性的值，才会引起KVO通知*

For detail, see [Introduction to Key-Value Observing Programming Guide](#)

Demo