Documentation

Language: Swift

Address Book / ABMutableMultiValue

Class

ABMutableMultiValue

A mutable representation of a property that might have multiple values.

macOS

```
class ABMutableMultiValue
```

Overview

Each value in a multivalue list must be of the same type, and must have an associated predefined or user-defined label, and unique identifier. The labels, however, need not be unique. For example, you can have multiple Home phone numbers. Each multivalue object may have a primary identifier —used as a default value when a label is not provided. For example, a person record may have multiple addresses with the labels Home and Work, where Work is designated as the primary value. Instances of `ABMutableMultiValue` are mutable, see `ABMultiValue` for additional methods that access the content of a multivalue list.

The `ABMutableMultiValue` class is “toll-free bridged” with its procedural C opaque-type counterpart. This means that the `ABMutableMultiValue` type is interchangeable in function or method calls with instances of the `ABMutableMultiValue` class.

Topics

Adding a value

```
func add(Any!, withLabel: String!) -> String!
```

Adds a value and its label to a multivalue list.

```
func insert(Any!, withLabel: String!, at: Int) -> String!
```

Inserts a value and its label at the given index in a multivalue list.

Replacing values and labels

```
func replaceLabel(at: Int, withLabel: String!) -> Bool
```

Replaces the label at the given index.

```
func replace(at: Int, withValue: Any!) -> Bool
```

Replaces the value at the given index.

Removing values

```
func removeAndLabel(at: Int) -> Bool
```

Removes the value and label at the given index.

Setting the Primary identifier

```
func setPrimaryIdentifier(String!) -> Bool
```

Sets the primary value to be the value for the given identifier.

Relationships

Inherits From

`ABMultiValue`

Conforms To

```
CVarArg
CustomDebugStringConvertible
CustomStringConvertible
Equatable
Hashable
NSCopying
NSFastEnumeration
NSMutableCopying
```

`NSObjectProtocol`

See Also

Data Types

`class ABPerson`

An object that encapsulates all information about a person in the Address Book database.

`class ABGroup`

An object that represents a group of records in the Address Book database.

`class ABMultiValue`

An immutable representation of a property that might have multiple values.

`protocol ABImageClient`

Methods for responding to a request to load images associated with a contact.

`class ABRecord`

An abstract class that defines the common properties for all Address Book records.