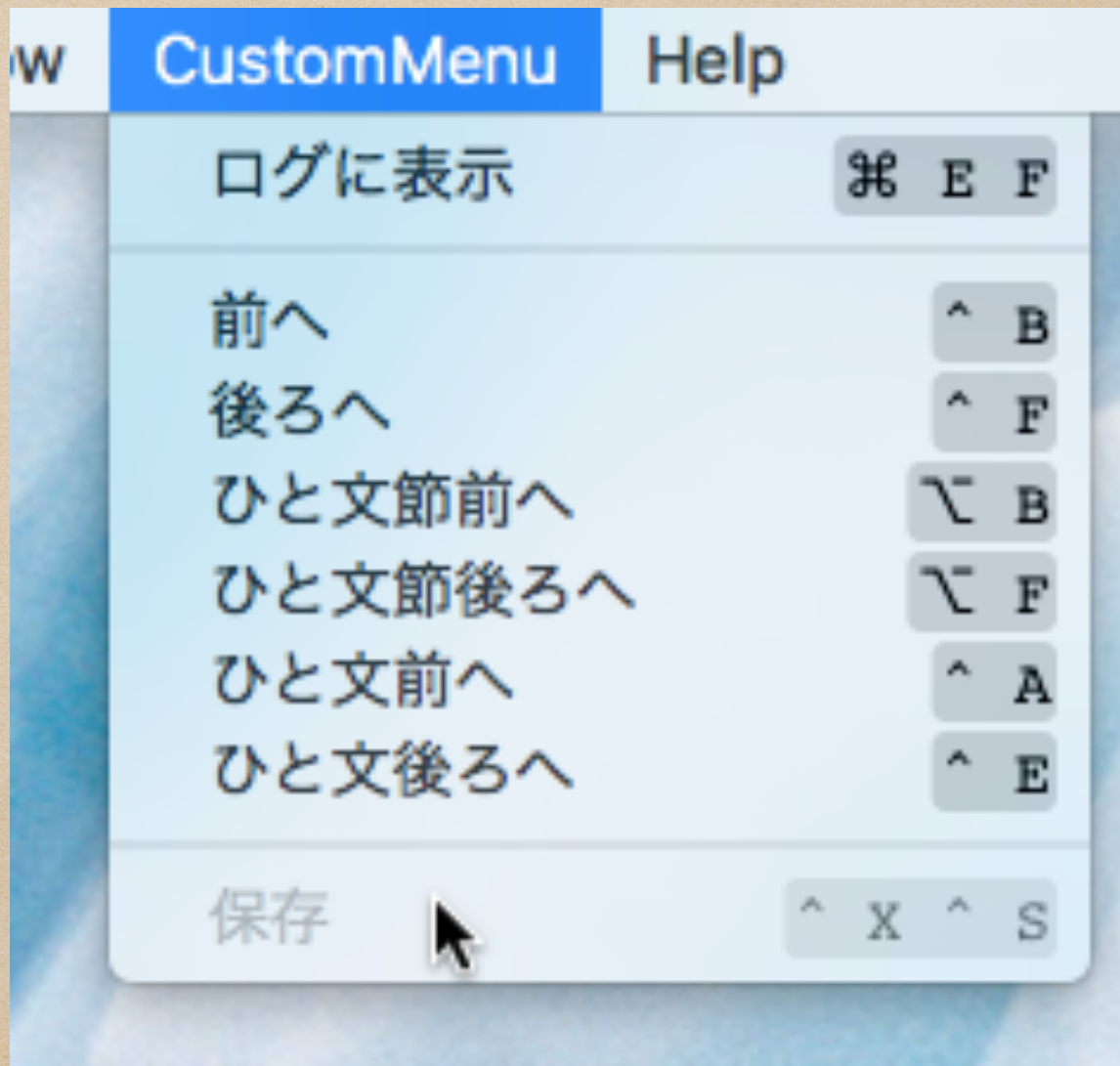


メニューのショートカット 表記の変更

mindtools@mac.com

こんなメニューを作る方法の解説



使い方

追加するファイル

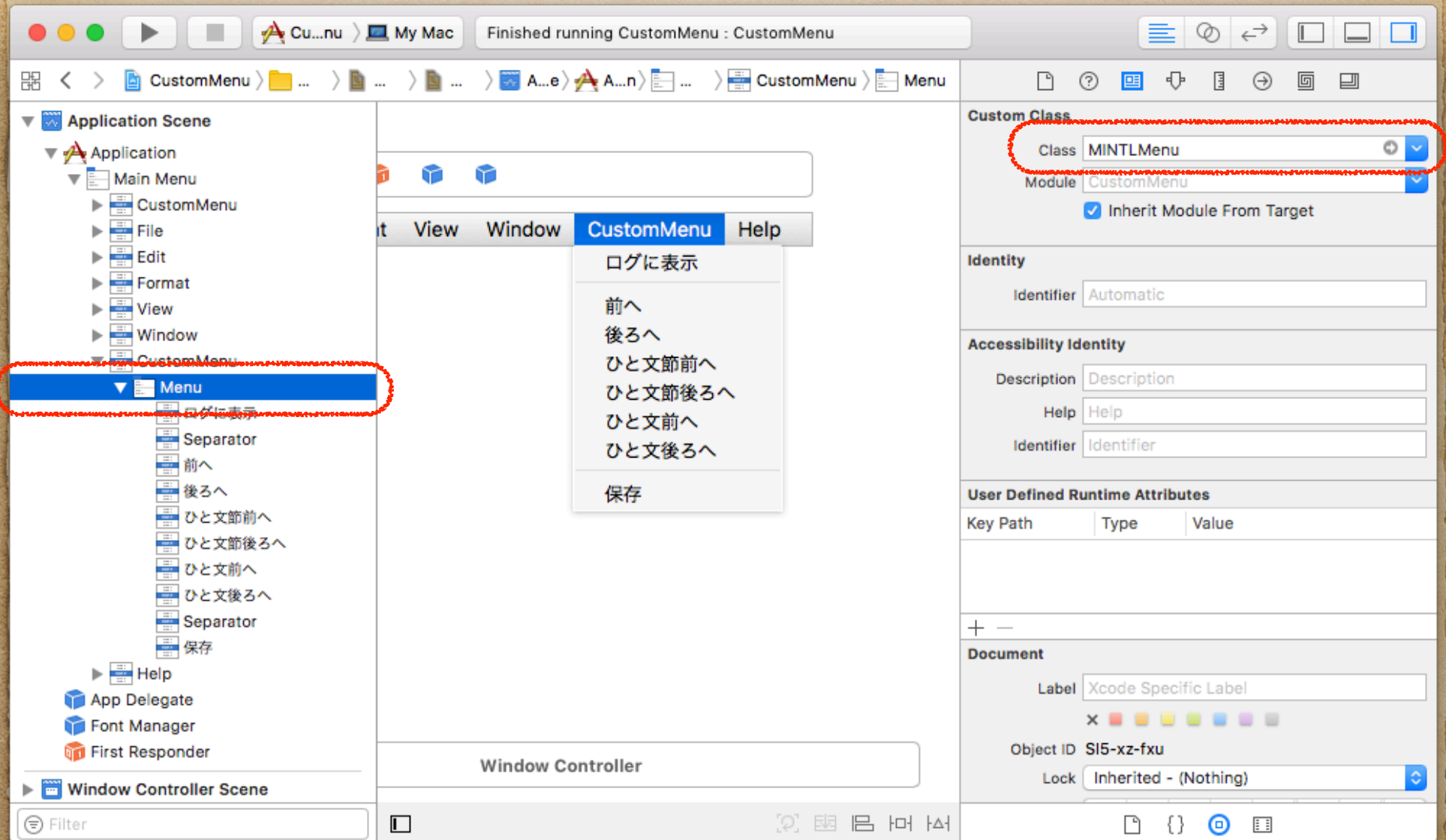
MINTLDynamicLink	swiftからヘッダファイルのないC言語関数を呼び出すための、低レベルなユーティリティ。
MINTLMenu	NSMenuのサブクラス。今回の表記の表示のOn/Offや、ドキュメント化されていない関数の呼び出しなどの雑
MINTLMenuItem	NSMenuItemのサブクラス。表示する文字列を保持し描画する。

追加するコード

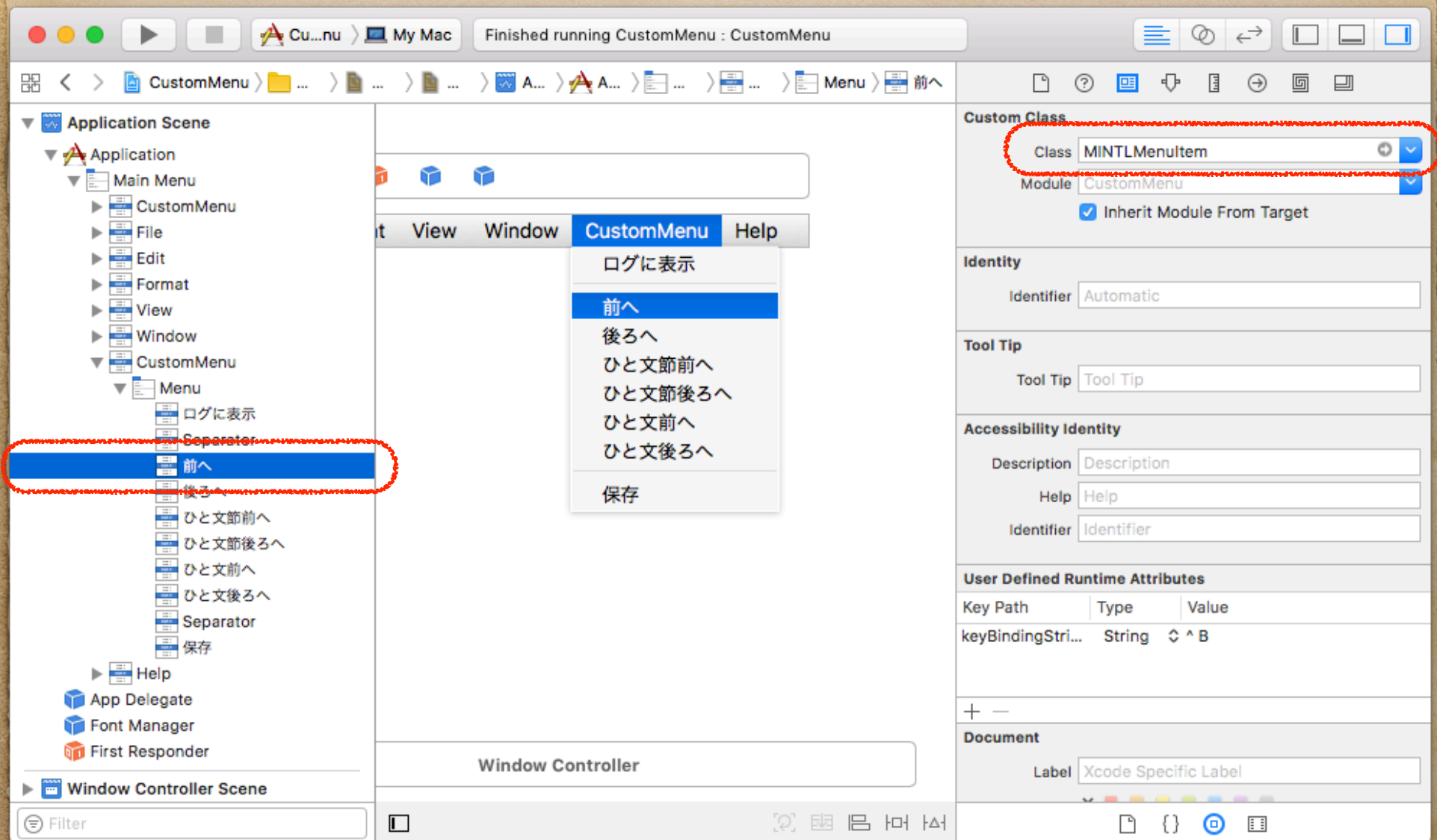
```
class AppDelegate: NSObject, UIApplicationDelegate
{
    .
    .
    .

    func applicationDidFinishLaunching(_ aNotification: Notification)
    {
        MINTLMenu.enableCustomDrawOnMainMenu()
    }
    .
    .
    .
}
```

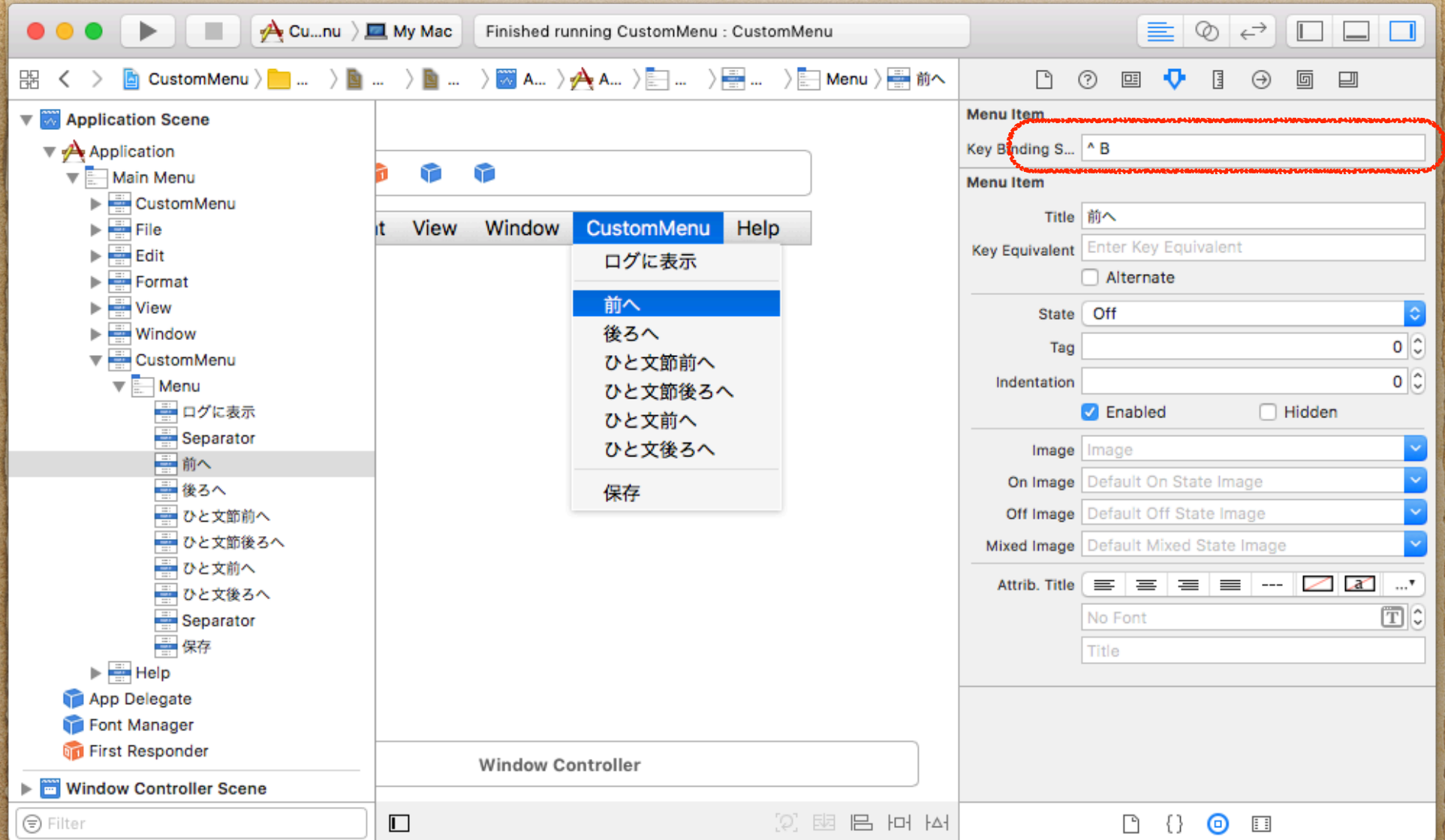
追加するストーリーボード



追加するストーリーボード

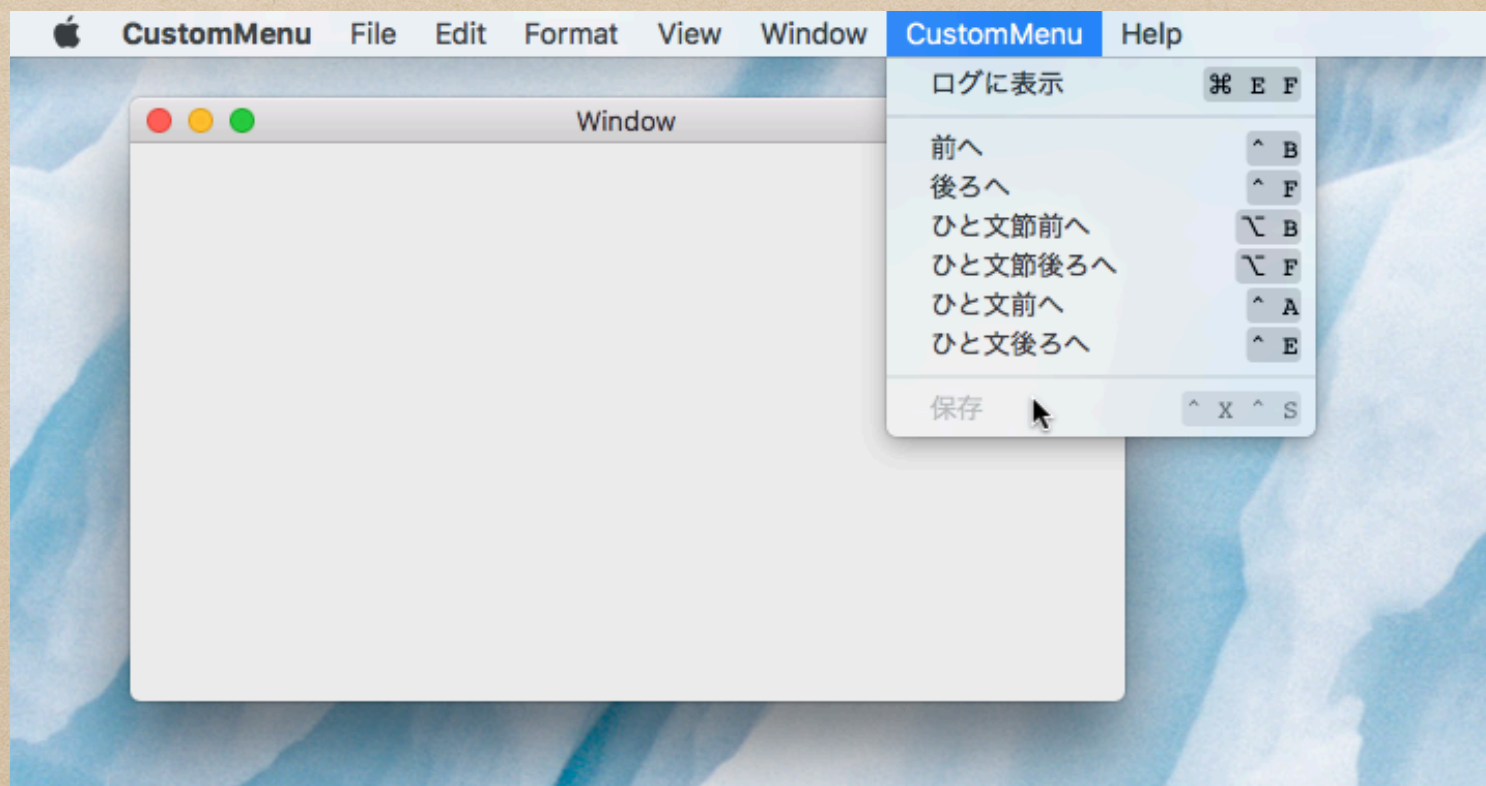


追加するストーリーボード



サンプルコード

CustomMenu



コードの解説

非公開CarbonAPIの呼び出し方

コードの解説

非公開CarbonAPIの呼び出し方

- ~ 関数名を探し出す
- ~ 関数名の文字列から関数ポインターを取得する
- ~ 関数ポインターをSwiftの関数に変換する
- ~ swiftの関数を呼び出す

関数名を探し出す

- ~ コードの話ではない
- ~ ググったり、昔の文献を漁ると出てくる
- ~ 非公開APIだと諦めない

関数ポインターを取得

dlopen、dlcloseは非公開関数ではない

dlsymだけが必要

dlsymにはdlopenで取得したライブラリへのハンドルが必要

ハンドルの値が-3の時はOpen済みのライブラリから検索する

関数ポインターを取得

```
let theSymbolAddress = dlsym( -3,  
    “関数名”.cString(  
        using: String.Encoding.utf8))
```

Swiftの関数に変換

```
unsafeBitCast(theSymbolAddress,  
              to: 関数の型)
```

swiftの関数を呼び出し

~ ここまでくれば普通に呼び出しで終わりです

コード例

コードの解説

```
extern OSStatus      ChangeMenuItemAttributes(  
    MenuRef menu,  
    MenuItemIndex item,  
    MenuItemAttributes setTheseAttributes,  
    MenuItemAttributes clearTheseAttributes  
);// HIToolbox/Menus.h 32bit Only
```

コードの解説

```
typedef ChangeMenuItemAttributesType =  
    @convention(c) (Menu,  
                    MenuItemIndex,  
                    MenuItemAttributes,  
                    MenuItemAttributes) -> OSStatus
```

```
static var changeMenuItemAttributes: ChangeMenuItemAttributesType?  
    = nil
```

C言語の関数ポインタはNullable

コードの解説

```
func changeMenuItemAttributes(  
    _ menu: Menu,  
    _ item: MenuItemIndex,  
    _ setTheseAttributes: MenuItemAttributes,  
    _ clearTheseAttributes: MenuItemAttributes)  
    -> OSStatus  
{  
    if ( MINTLMenu.changeMenuItemAttributes == nil )  
    {  
        MINTLMenu.changeMenuItemAttributes =  
            MINTLDynamicLink.symbolPtrFromSelf(  
                symbol: "ChangeMenuItemAttributes",  
                type: ChangeMenuItemAttributesType.self)  
    }  
  
    return MINTLMenu.changeMenuItemAttributes!(  
        menu,  
        item,  
        setTheseAttributes,  
        clearTheseAttributes)  
}
```