

Erledigt

## Jemand mit XCode/Swiftkenntnissen hier?

**Beitrag von „Sascha\_77“ vom 18. Mai 2018, 16:46**

Hi Leute.

Ich versuche mich gerade etwas an Swift. Ich habe mir schon ein kleines Beispiel zusammengebaut. Bin normal eher in der Bash-Welt angesiedelt und kriege eine Sache jetzt nicht hin. In dem angehangenen Beispiel möchte ich, dass der Output Text von "ls -la" in NSTextView dargestellt wird. Habe mir schon einen heissen gegoogled aber ich komme auf keinen grünen Zweig. Vllt. kann mal ein Swiftkenner ein Auge drauf werfen und mir sagen was ich tun muss? Ist bestimmt nur eine Kleinigkeit die da noch fehlt denke ich.

---

**Beitrag von „Brumbaer“ vom 18. Mai 2018, 16:54**

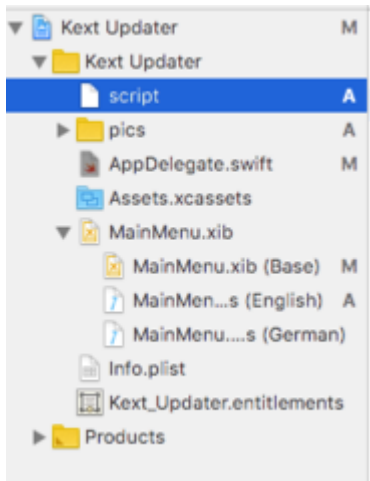
Die Verbindung von Logger zum TextView ist nicht korrekt.

Im InterfaceBuilder - AppDelegate die Outlet Verbindung von logger löschen und neu setzen. Und zwar vom dann freien Kreis eine Linie in den textView ziehen und erst dann loslassen, wenn TextView angezeigt wird.

---

**Beitrag von „Sascha\_77“ vom 18. Mai 2018, 20:31**

Danke klappt soweit. Hab jetzt noch nen anderes Problem. Habe ein Bashscript im Projekt mit drin. Wie teil ich ihm den Pfad mit wo das script sich relativ befindet?



Sowas wie \$SRCROOT funktioniert nicht.

---

### **Beitrag von „Ezak“ vom 18. Mai 2018, 21:03**

.path(forResource:) findet resourcen eigentlich automatisch

```
let path = Bundle.main.url(forResource: "script")
```

xcode kopiert hinzugefügte resourcen nach /Contents/Resources soweit ich weis

Vg

---

### **Beitrag von „Sascha\_77“ vom 19. Mai 2018, 12:52**

Haut irgendwie auch nicht hin. Das let "nested" muss ich wie einbinden?

Code

```
1. let nested = Bundle.main.url(forAuxiliaryExecutable: "script.sh")
2. let path = "/bin/bash"
3. let arguments = nested
4. sender.isEnabled = false
5. let task = Process()
6. task.launchPath = path
7. task.arguments = arguments
8. let outputPipe = Pipe()
9. task.standardOutput = outputPipe
10. task.launch()
11. task.waitUntilExit()
12. let data = (task.standardOutput as! Pipe).fileHandleForReading.readDataToEndOfFile()
13. let output = String(data:data, encoding: .utf8)!
14. sender.isEnabled = true
```

Alles anzeigen

---

### **Beitrag von „Brumbaer“ vom 19. Mai 2018, 13:07**

Da steht doch alles.

Du übergibst ein Variable vom Typ URL? als Argument das ein Array von String sein muss.  
Geht nicht. Du musst die URL entsprechend umwandeln.

---

### **Beitrag von „Ezak“ vom 19. Mai 2018, 14:31**

angenommen das scriptfile heisst test.sh

Code

```
1. //
2. // ViewController.swift
3. // SampleScriptRunner
4. //
```

```
5. // Created by Marc Szymkowiak on 19.05.18.
6. // Copyright © 2018 Marc Szymkowiak. All rights reserved.
7. //
8.
9.
10. import Cocoa
11.
12.
13. class ViewController: NSViewController {
14.
15.
16. @IBOutlet weak var outputBox: NSTextField!
17. override func viewDidLoad() {
18. super.viewDidLoad()
19.
20.
21. // Do any additional setup after loading the view.
22. }
23.
24.
25. override var representedObject: Any? {
26. didSet {
27. // Update the view, if already loaded.
28. }
29. }
30.
31.
32. @IBAction func RunScript(_ sender: Any) {
33. let scriptPath = Bundle.main.path(forResource: "test", ofType: "sh")
34. let task = Process()
35. task.launchPath = "/bin/bash"
36. task.arguments = ["-c", scriptPath] as! [String]
37. let pipe = Pipe()
38. task.standardOutput = pipe
39. task.launch()
40.
41. let data = pipe.fileHandleForReading.readDataToEndOfFile()
42. let output = String(data:data, encoding: .utf8)!
43.
44. outputBox.objectValue = output
45. }
46.
47. }
```

Alles anzeigen

Hats funktioniert? Wenn nicht können wir morgen ja mal danach schauen