



ripple motion

Créateur d'applications mobiles



TECH TALK

GRATUITS & OUVERTS AU PUBLIC

Tech talk sous forme de courtes formations
le **mercredi** de **17h** à **18h** !

N'hésitez pas à nous **contacter** afin de participer
aux **prochaines sessions** au sein de nos locaux !

TECH TALK

ANIMÉ PAR



OLIVIER TABONE

CO FONDATEUR / DIRECTEUR TECHNIQUE

olivier.tabone@ripplemotion.fr



SWIFT STYLE



OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

**CODE THAT'S READABLE, CONSISTENT, AND
COMPREHENSIBLE LOWERS COSTS IN TERMS
OF LIFE SPAN, MAINTAINABILITY, AND ERROR
PREVENTION.**

Erica Sadun



OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

STRUCTURE YOUR CODE FOR READABILITY

Spacing + Braces

```
func f1(x:Int=0, y:Int=0)->Int{ // no
    return x+y //no
}

func f2(x: Int = 0, y: Int = 0) -> Int { // yes
    return x + y //yes
}

func f3(x: Int = 0, y: Int = 0) -> Int
{ // no
    return x + y
}

func f4(x: Int = 0, y: Int = 0) -> Int { return x + y } //yes
```



OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE



STRUCTURE YOUR CODE FOR READABILITY

Parameter Alignment

```
func f5(someCell: UITableViewCell, //no
        anOtherCell: UITableViewCell,
        cellNumberThree: UITableViewCell,
        finalCell: UITableViewCell) -> Void {
}
```

```
func f6(someCell: UITableViewCell, //yes
        anOtherCell: UITableViewCell,
        cellNumberThree: UITableViewCell,
        finalCell: UITableViewCell) -> Void {
}
```




OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

STRUCTURE YOUR CODE FOR READABILITY

Closures

```
[1, 2, 3].map({ 2 * $0 }) // short style

[1, 2, 3].map { value -> Int in //ok long style
  // something not trivial here
  return 0
}

[1, 2, 3].map({ value -> Int in //no
  // something not trivial here
  return 0
})
```




OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

STRUCTURE YOUR CODE FOR READABILITY

Promises

```
doSomething()  
  .then { value -> Void in  
        print("got result \ \(value)")  
      }  
  .catch { error -> Void in  
         self.present(error: error)  
       }  
  .always {  
    activityIndicator.stopAnimating()  
  }
```




OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

STRUCTURE YOUR CODE FOR READABILITY

Line width

119



OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

STRUCTURE YOUR CODE FOR READABILITY

Line width

~~119~~

120



OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

STRUCTURE YOUR CODE FOR READABILITY

Colon style

```
let data1 = ["a":1, "b":2, "c":3] //no  
let data2 = ["a": 1, "b": 2, "c": 3] //yes
```




OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

STRUCTURE YOUR CODE FOR READABILITY

Number literals

```
let bigNumber = 1000000 //no  
let oneMillion = 1_000_000 // yes, easier to read
```




OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

PREFERRED PRACTICES

Optionals: same name shadowing

```
if let unwrappedX = x { // no
}

if let x = x { //yes
}

guard let x = x else { return } // yes
```




OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

PREFERRED PRACTICES

Optionals: nil coalescing operator `??`

```
x ?? sideEffect()  
  
if x != nil { //yes  
    sideEffect()  
}
```




OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

PREFERRED PRACTICES

Optionals: no force unwraps / force casts

```
let plusOne = x! + 1 //no

let addOne = (x ?? 0) + 1 //yes provide default

if let x = x { //yes, conditional evaluation
    let oneMore = x + 1
}

guard let x = x else { fatalError("sensible message error") }
let zeroPlusOne = x + 1
```




OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

PREFERRED PRACTICES

Optionals: iterating a sequence of optionals

```
do {  
  let seq = [1, nil, 3]  
  
  for x in seq {  
    if let x = x { // no, cumbersome  
      print(x)  
    }  
  }  
  
  for case let x? in seq { //yes, pattern matching  
    print(x)  
  }  
}
```




OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

PREFERRED PRACTICES

Optionals: iterating an optional sequence

```
let seq: [Int]? = [1, 2, 3]

if let seq = seq {           // no, cumbersome
    for x in seq {
        print(x)
    }
}

for x in seq ?? [] {        // yes
    print(x)
}

seq?.forEach { item in     // preferred
    print(x)
}
```




OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

PREFERRED PRACTICES

Lazy sequence

```
let seq = [1, 2, 3, 4, 5]

seq.map({ $0 * 2 }).first      // no, calls map on every entry

seq.lazy.map({ $0 * 2 }).first // yes
```




OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

PREFERRED PRACTICES

Switch

```
// no
if condition1 && condition2 {
    // code1
} else if condition1 && !condition2 {
    // code2
} else if !condition1 && condition2 {
    // code3
} else { // !condition1 && !condition2
    // code4
}

// yes
switch (condition1, condition2) {
    case (true, true): break // code 1
    case (true, false): break // code 2
    case (false, true): break // code 3
    case (false, false): break // code 4
    // no default case, compiler backs us
}
```




OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

SWIFTLINT TO THE RESCUE



OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

SWIFTLINT

Installation

- **add in Podfile**

```
4 ▼ target 'SwiftStyle' do~  
5     # Comment the next line if you're not using Swift and don't want to use  
6     # dynamic frameworks~  
6     use_frameworks!~  
7     ~  
8     # Pods for SwiftStyle ~  
9     pod 'SwiftLint'~  
10    ~  
11 ▲ end~
```




OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE



SWIFTLINT

Target Configuration

The screenshot shows the Xcode target configuration interface for a target named 'swiftlint'. The 'Shell' is set to '/bin/sh'. A script is defined with the following command:

```
1 $PODS_ROOT/SwiftLint/swiftlint lint --path $TARGETNAME
```

Below the script, there are two unchecked checkboxes:

- Show environment variables in build log
- Run script only when installing

There are also sections for 'Input Files' and 'Output Files', each with a placeholder 'Add input files here' and 'Add output files here' respectively. At the bottom, there is a section for 'Compile Sources (3 items)'.

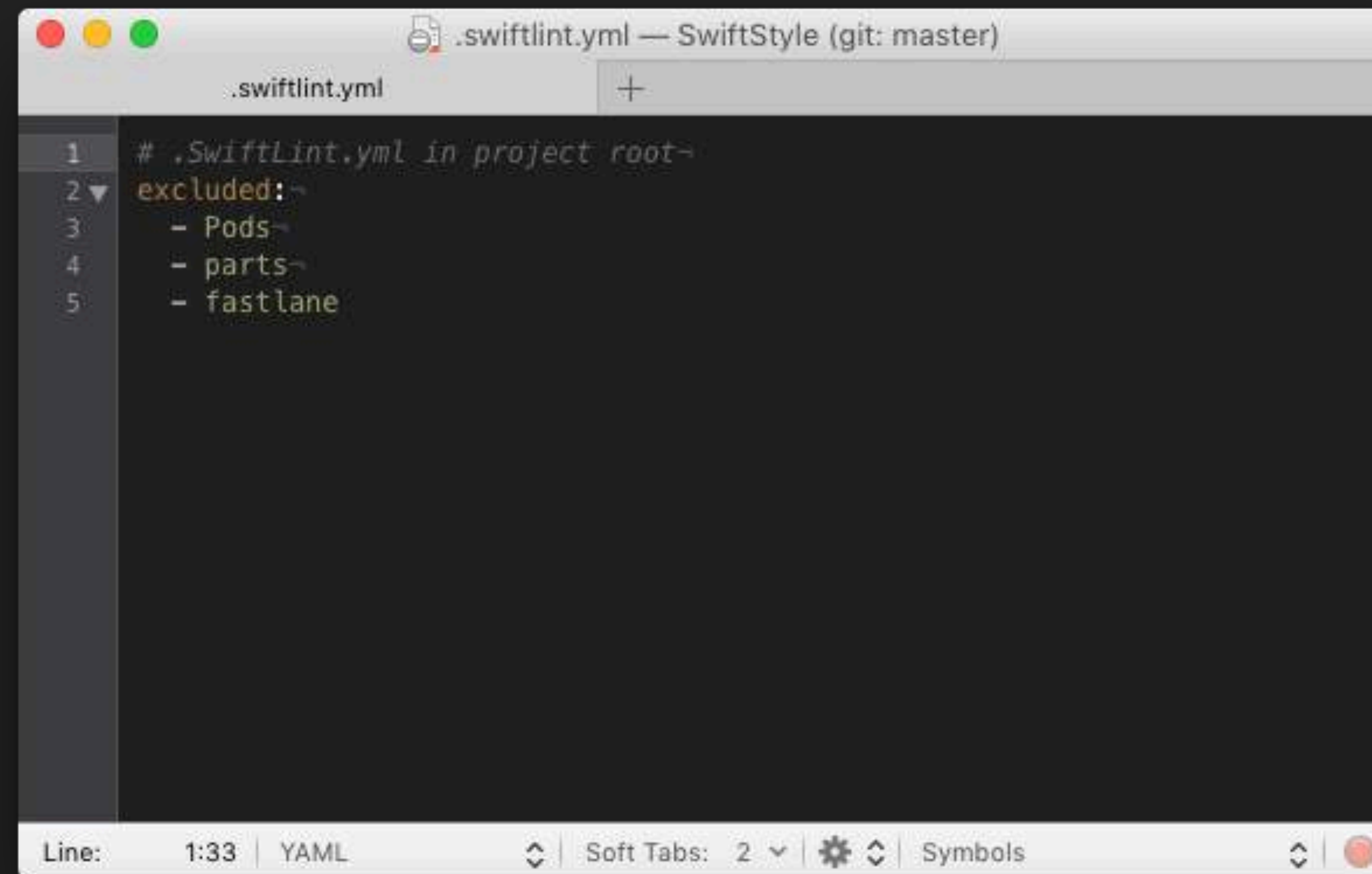


OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

SWIFTLINT

.SwiftLint.yml at project level



The screenshot shows a code editor window titled ".swiftlint.yml — SwiftStyle (git: master)". The editor displays the following YAML content:

```
1 # .SwiftLint.yml in project root~
2 excluded:~
3   - Pods~
4   - parts~
5   - fastlane
```

The editor interface includes a line number column on the left (1-5), a status bar at the bottom showing "Line: 1:33 | YAML", and various editor settings like "Soft Tabs: 2" and "Symbols".



OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE



SWIFTLINT

Run early in the CI build

```
Jenkinsfile — dev (git: xcode9)
.swiftlint.yml
Jenkinsfile
+
1 node ('xcode9') {~
2   stage ('checkout') {~
3     checkout scm~
4   }~
5   ~
6   stage ('buildout') {~
7     wrap([$class: 'AnsiColorBuildWrapper', 'colorMapName': 'XTerm']) {~
8       sh './buildout.sh'~
9     }~
10  }~
11  ~
12  stage ('swiftlint') {~
13    sh './Pods/SwiftLint/swiftlint'~
14  }~
15  ~
16  stage ('prepare simulator') {~
17    sh 'xcrun simctl shutdown booted || true'~
18    sh 'xcrun simctl erase all || true'~
19  }~
20  ~
21  stage ('unit-tests') {~
22    timeout(90) {~
23      wrap([$class: 'AnsiColorBuildWrapper', 'colorMapName': 'XTerm']) {~
24        sh './bin/fastlane unit_tests || true'~
25      }~
26    }~
27  }~
28  ~
29  stage ('ui-tests') {~
30    wrap([$class: 'AnsiColorBuildWrapper', 'colorMapName': 'XTerm']) {~
31      sh './bin/fastlane ui_tests || true'~
32    }~
33  }~
34  ~
35  stage ('publish test results') {~
36    step([$class: 'JUnitResultArchiver', 'testResults': 'build/*tests/report.junit'])~
37  }~
38 }
```

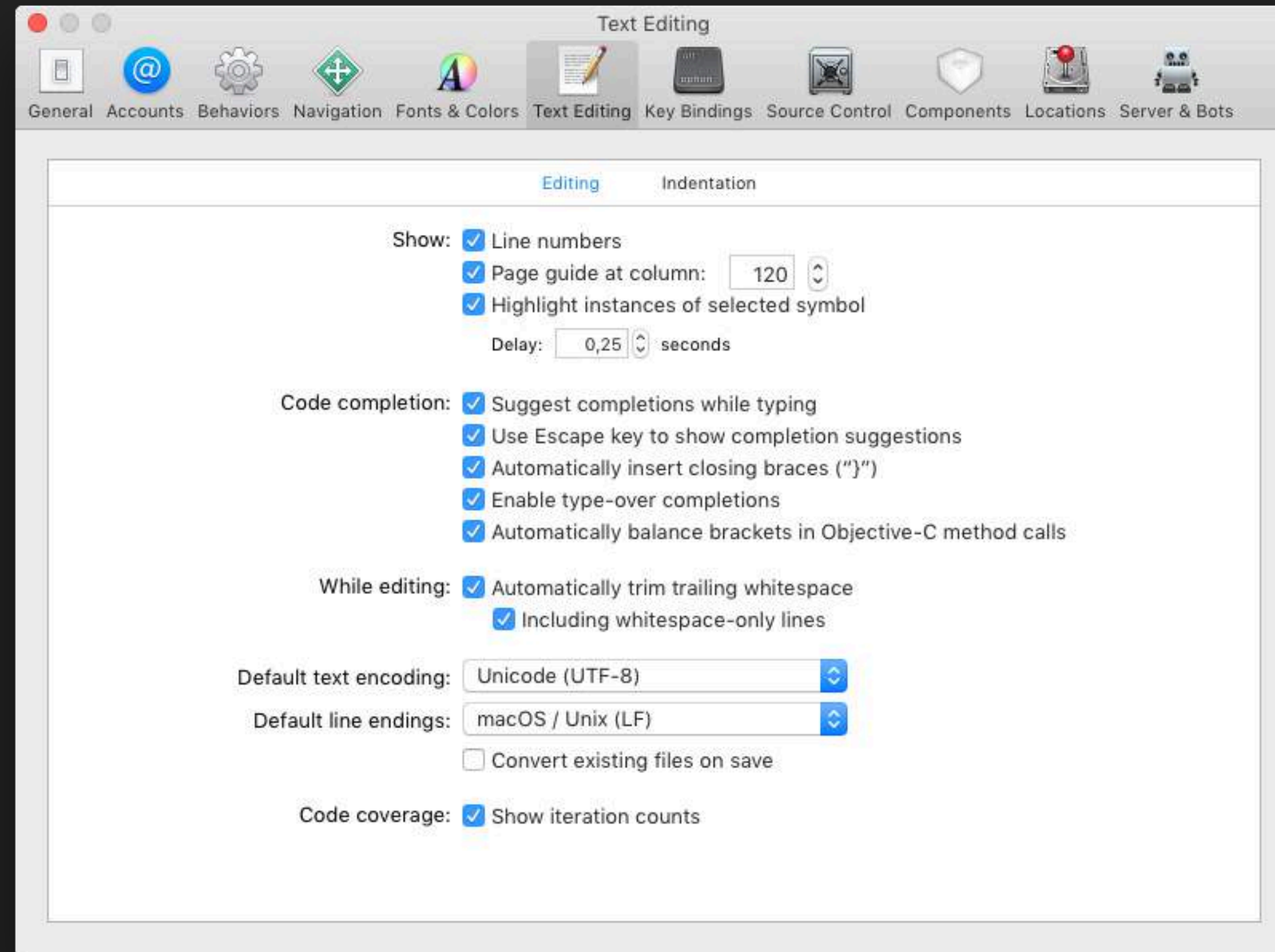


OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

SWIFTLINT

Configure Xcode





OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

SWIFTLINT

Per File/Function/Statement tuning

- Sometime you need exceptions to the linter
 - `//swiftlint:disable:next force_try`
 - `//swiftlint:disable file_length`
 - `//swiftlint:disable:this force_cast`



OLIVIER TABONE

CO FONDATEUR
DIRECTEUR TECHNIQUE

SWIFTLINT

Want More ?

- Sample Project

<https://bitbucket.org/ripplemotion/swift-style-ios>

- Erica Sadun's Swift Style

<https://pragprog.com/book/esswift/swift-style>

(available in my library)

MERCI DE VOTRE PARTICIPATION !



ripple motion

Créateur d'applications mobiles

