

# Swift-indent

## Introduction

Note: This tool is still a work in progress.

swift-indent is a tool for automatically indenting your Swift files according to a set of rules. It is implemented as another driver kind, like swiftc, the batch compiler, so swift-indent is actually a symbolic link to swift. This tool uses libIDE to indent code, so it can be leveraged from multiple systems and editors.

## Usage

To print all the available options:

```
swift-indent -help
```

By default, swift-indent will output the indented file to the standard output:

```
swift-indent sample.swift
```

You can either output the result to a separate file:

```
swift-indent sample.swift -o result.swift
```

Or you can indent in-place (the original file will be overwritten):

```
swift-indent -in-place sample.swift
```

If you want to indent using tabs instead of spaces, use the `-use-tabs` option:

```
swift-indent -use-tabs sample.swift
```

You can set the number of tabs or spaces using the `-tab-width` and `-indent-width` options, respectively.

If you want to indent cases in switch statements, use the `“-indent-switch-case”` option. The result would be something like this:

```
switch aSwitch {  
    case .some(let s):  
        print(s)
```

swift-indent supports indenting a range of lines from a file:

```
swift-indent -line-range 2:45 sample.swift
```

This will indent the file from lines 2 to 45, inclusive.

You can indent several files, but the `-line-range` option is not supported in that case.

You can also provide several line ranges by using multiple `-line-range` options:

```
swift-indent -line-range 2:45 -line-range 100:120 sample.swift
```