

# GitHub Repository Finder

iOS Developer 조지은 (Jenni Jo)

사용언어 : Swift 4.0

사용라이브러리: Alamofire

Kingfisher

RxSwift

RxCocoa

OAuthSwift



# CONTENTS

A close-up photograph of a person's hands writing on a clipboard with a yellow pen. The person is wearing a white shirt and a black wristband. The clipboard has some handwritten notes on it.

1

---

개발 요건



2

---

개발 일정



3

---

추가 개선 사항



4

---

코딩 규약서

1

## 개발 요건



Step 1 OAuth 를 이용하여 Login 하기

Step 2 사용자 프로필 화면 만들기

Step 3 Repository 검색 및 결과 리스트 화면 만들기

Step 4 결과 리스트 선택 후 상세 화면 이동 및 해당 Respository에 내 Star 추가하기

## 2

## 개발 일정

	기능	8월 31일(금)	9월 1일(토)	9월 2일(일)
1	Github OAuth Token 생성 및 로그인 화면 만들기			
2	User Model, Repo Model 만들기			
3	ProfileViewController, SearchViewController 만들기			
4	DetailViewController, StarsViewController 만들기			
5	Pull To Refresh, Load more 기능 구현			
6	PPT 문서작업			
7	앱 테스트			

## 3

## 추가 개선 방안

- 검색 시 하이라이팅 및 자동완성 기능을 추가한다.  
→ 키워드가 포함된 문자열은 하이라이팅 시켜주는 기능을 추가하여 UX를 개선한다.  
자동완성 기능을 위해 검색어 히스토리를 관리하는 DB생성이 요구된다.

## 네이밍

- 클래스 - 린트 기본 규칙이므로 제외하지 않으면 적용됩니다 (identifier : type\_name)

클래스 이름에는 **UpperCamelCase**를 사용합니다.

클래스 이름에는 접두사 **Prefix**를 붙이지 않습니다.

- 함수 - 린트 기본 규칙이므로 제외하지 않으면 적용됩니다 (identifier : identifier\_name)

함수 이름에는 **lowerCamelCase**를 사용합니다.

함수 이름 앞에는 되도록이면 **get**을 붙이지 않습니다.

## code

```
good:  
func name(for user: User) -> String?  
  
bad:  
func getName(for user: User) -> String?
```

- 변수 - 린트 기본 규칙이므로 제외하지 않으면 적용됩니다 (identifier : identifier\_name)

변수 이름에는 lowerCamelCase를 사용합니다.

## 상수

- 상수 이름에는 lowerCamelCase를 사용합니다. - 린트 기본 규칙이므로 제외하지 않으면 적용됩니다 (identifier : identifier\_name)

### code

```
good:  
let maximumNumberOfLines = 3  
  
bad:  
let kMaximumNumberOfLines = 3  
let MAX_LINES = 3
```

줄바꿈

함수 정의시 parameter인자가 2개 이상일때는 줄바꿈을 하도록 합니다.

2개 이하는 줄바꿈 하지 않고 유자

Page guide at column을 110자로 유지함.

110이 넘어 갈 경우에는 함수의 파라미터를 개행하여 적용함.

개행 규칙은 아래 코드 블록을 참고.

## code

```
func collectionView(_ collectionView: UICollectionView, cellForItemAt indexPath: IndexPath) -> UICollectionViewCell {
    // doSomething()
}

func animationController(forPresented presented: UIViewController,
                        presenting: UIViewController,
                        source: UIViewController) ->
UIViewControllerAnimatedTransitioning? {
    // doSomething()
    return nil
}
```

if let 및 guard let이 구문이 길 경우에는 줄바꿈을 합니다.

code

```
if let user = self.veryLongFunctionNameWhichReturnsOptionalUser(),
    let name = user.veryLongFunctionNameWhichReturnsOptionalName(),
    user.gender == .female {
    // ...
}
```

Enumerations

enum의 case스타일은 Lowercase camel style을 사용하도록 한다.

## code

```
good:  
enum Color {  
    case red  
    case green  
    case blue  
}
```

```
bad:  
enum Color {  
    case Red  
    case Green  
    case Blue  
}
```



## Brackets

함수 및 클래스명등의 시작 bracket은 시작 라인 뒤에서 여는 brackets를 지정하도록 한다.

## code

```
good:  
func sayHello() {  
    print("Hello")  
}
```

```
bad:  
func sayHi()  
{  
    print("Hi")  
}
```



### 기본적으로 클래스는 `final`로 작성 - 권장사항

클래스는 `final`로 시작해야 합니다. 상속이 필요하다면 클래스는 서브클래스를 만들도록 할 수 있게 변경합니다. 심지어, 클래스 내 가능한 많은 정의는 `final`이 되어야 하고, 같은 규칙을 따릅니다.

이유: 일반적으로 구성은 상속을 선호하고, 아마도 상속을 선택하면 결정하는데 좀 더 많은 생각을 해야 합니다.



## 공백 - 적용중

- 공백이 아닌 탭 사용.
- 파일 끝에 줄 바꿈.
- 코드를 논리 단위로 나누도록 빈 줄을 아낌없이 사용. (빈 줄 단위는 1줄을 넘지 않도록 한다.) (identifier: vertical\_whitespace)
- 줄 끝에 공백을 남기지 않기.
- 빈 줄에는 공백이 포함되지 않도록 합니다.
- 모든 파일은 빈 줄로 끝나도록 합니다.
- MARK 구문 위와 아래에는 공백이 필요합니다.

## code

```
// MARK: Layout

override func layoutSubviews() {
    // doSomething()
}

// MARK: Actions
override func menuButtonDidTap() {
    // doSomething()
}
```

**var** 바인딩보다 **let** 바인딩을 선호 - 스위프트 기본 규칙

가능하면 `var foo = expression`보다 `let foo = expression`을 사용합니다.

값이 변할 것을 알고 있고, `weak` 변수에만 `var`를 사용하도록 합니다.

이유: 두 키워드의 목적과 의미는 명확하지만, 기본적으로 `let`이 더 안전하고 명확한 코드입니다.

프로그래머에게 `let`-바인딩은 값이 예상되고, 바뀌지 않을 것임을 보장하고 명확하게 전달합니다. 후속 코드는 변수의 용도를 더 쉽게 추측할 수 있습니다.

코드를 더 쉽게 이해하게 됩니다.

 옵셔널의 강제 언래핑 사용 피하기 - 린트 기본 규칙이므로 제외하지 않으면 적용됩니다 (identifier : force \_cast)

타입 `FooType?` 또는 `FooType!`의 `foo` 식별자가 있다면, 강제 언래핑으로 값(`foo!`)를 값을 얻으려고 하지 않습니다.

대신에, 아래 방식을 추천 :

감사합니다