

# 구조체

# 구조체

- ▶ 스위프트의 대부분 타입은 구조체
- ▶ 내부에 프로퍼티, 함수를 선언하고 인스턴스를 만들어서 사용

# 구조체 선언

```
1  struct 구조체이름 {  
2      |    // 코드  
3      }
```

# 구조체 활용법

```
1  import Foundation
2
3  struct Example {
4      // 가변 프로퍼티
5      var mutableProperty: String = ""
6
7      // 불변 프로퍼티
8      let immutableProperty: String = ""
9
10     // 인스턴스 메소드
11     func showPrint() -> Void {
12         print("출력")
13     }
14 }
15
16 // 불변 인스턴스
17 let mutableInstance: Example = Example()
18 // mutableInstance.mutableProperty = "변경" (에러발생)
19 mutableInstance.showPrint()
20
21 // 가변 인스턴스
22 var immutableInstance: Example = Example()
23 immutableInstance.mutableProperty = "변경"
24 immutableInstance.showPrint()
```

클래스

# 클래스

- ▶ 내부에 프로퍼티, 또는 함수를 선언하고 인스턴스를 만들어서 사용
- ▶ 다중 상속 불가능

# 클래스 정의

```
1  class Person {  
2  |    // 코드  
3  }
```

# 클래스 활용법

```
1  import Foundation
2
3  class Example {
4      // 가변 프로퍼티
5      var mutableProperty: String = ""
6
7      // 불변 프로퍼티
8      let immutableProperty: String = ""
9
10     // 인스턴스 메소드
11     func showPrint() -> Void {
12         print("출력")
13     }
14 }
15
16 // 불변 인스턴스
17 let mutableInstance: Example = Example()
18 mutableInstance.mutableProperty = "변경"
19 mutableInstance.showPrint()
20
21 // 가변 인스턴스
22 var immutableInstance: Example = Example()
23 immutableInstance.mutableProperty = "변경"
24 immutableInstance.showPrint()
```



# 클래스 vs 구조체

	구조체	클래스
평가전략	Call by value	Call by reference
메모리 영역	스택	힙
Codable	가능	불가능

열거형

# 열거형

- ▶ 유사한 종류의 여러 값을 한 곳에 모아서 정의한 것(예: 요일, 계절 등)
- ▶ Enum 자체가 하나의 데이터 타입
- ▶ 각각의 case는 그 자체가 고유의 값임
- ▶ 열거형 내 메소드 추가 가능

# 열거형 정의

```
1  enum 이름 {  
2      case 이름1  
3      case 이름2, 이름3, ...  
4  }
```

# 열거형 활용법

```
1  enum Day {
2      case sun, mon, tue, wed, thu, fri, sat
3
4      func showToday() -> Void {
5          switch self {
6              case .sun:
7                  print("일")
8              case .mon:
9                  print("월")
10             case .tue:
11                 print("화")
12             case .wed:
13                 print("수")
14             case .thu:
15                 print("목")
16             case .fri:
17                 print("금")
18             case .sat:
19                 print("토")
20         }
21     }
22 }
23
24 let today:Day = Day.mon
25
26 switch today {
27     case .sun, .sat:
28         print("주말")
29     case .mon, .tue, .wed, .thu, .fri:
30         print("평일")
31 }
```

클로저

# 클로저

- ▶ 한번만 사용하는 일회용 함수
- ▶ 실행가능한 코드 블록
- ▶ 일급 객체로써 전달인자, 변수, 상수 등에 저장 및 전달 가능
- ▶ 매개변수 전달과 반환 값이 존재한다는 면에서는 함수와 동일하나 클로저는 이름정의가 필요하지 않음

# 클로저 정의

```
1  ⇨ { (매개 변수들) -> 반환 타입 in  
2    |   실행 코드  
3    }
```



# 클로저 활용법

```
1  let checkAdult = { (age:Int) -> String in
2    |   return age < 20 ? "청소년" : "성인"
3    }
4
5    print("이 고객은 \ (checkAdult(10))입니다.")
```

The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern, layered effect. The word "guard" is centered in a green, lowercase, sans-serif font.

guard

# Guard

- ▶ If문을 대체하기 위해 사용
- ▶ 조건이 참일 시 guard문은 통과하며, 거짓일 시 else구문을 실행한 후 종료
- ▶ else 내부에는 return이나 break가 반드시 존재하여야 함

# Guard 사용법

```
1 func showAdultWithGuard(age: Int?) -> Void {  
2     guard let age: Int = age, age < 130, age >= 20 else {  
3         print("나이값을 잘못되었거나 성인이 아닙니다.")  
4         return  
5     }  
6  
7     print("당신은 성인입니다.")  
8 }  
9  
10 showAdultWithGuard(age: 20)
```

# Guard vs If

- ▶ If문을 사용하여 옵셔널 바인딩을 할 경우 외부에서는 옵셔널 바인딩을 한 값을 사용할 수 없음

상속

# 상속

- ▶ 다중상속을 지원하지 않음
- ▶ 클래스, 프로토콜에서 가능하며, 열거형, 구조체는 상속 불가

# 상속 사용법

```
1  class Car {  
2      var type:String = "승용차"  
3  
4      func showMyCar() {}  
5  }  
6  
7  // Car를 상속받음  
8  class Brand: Car {  
9      var brand:String = "메르세데츠 벤츠"  
10  
11     override func showMyCar() -> Void {  
12         print("나의 차는 \(type), 브랜드는 \(brand)입니다.")  
13     }  
14 }  
15  
16 let brand:Brand = Brand()  
17  
18 brand.showMyCar()
```



# 프로토콜

# 프로토콜

- ▶ 특정 작업 또는 기능에 적합한 메소드, 속성 및 기타 요구 사항의 청사진
- ▶ 자바의 Interface와 유사함
- ▶ 객체 혹은 인터페이스 대신 프로토콜을 중심으로 설계

# 프로토콜 정의

- ▶ Protocol 키워드를 사용하여 정의

```
1 protocol 프로토콜 명 {  
2     // 내용  
3 }
```

# 프로토콜 구현

```
1  import Foundation
2
3  // 프로토콜 정의
4  @objc protocol SchoolProtocol {
5
6      var name:String { get set }
7
8      // 필수 정의 메소드
9      func showSchoolName() -> Void
10
11      // 선택 정의 메소드
12      @objc optional func setSchoolName() -> Void
13  }
14
15  // 프로토콜 상속
16  class School: SchoolProtocol {
17      func showSchoolName() -> Void {
18          print("동서대학교")
19      }
20
21      // optional로 정의할 경우 기재하지 않아도 가능
22  }
23
24  let school:School = School()
25
26  school.showSchoolName()
```

# 프로토콜 vs 인터페이스

- ▶ 프로토콜은 옵셔널 키워드를 통해 선택적으로 메소드를 설계할 수 있음
- ▶ 파라미터에 기본 값을 지정할 수 없음
- ▶ Static 정적 멤버 정의

익스텐션

# 익스텐션

- ▶ 구조체, 클래스, 열거형, 프로토콜 타입에 새로운 추가할 수 있는 기능

# 익스텐션 정의

```
1  extension 확장이름 {  
2  |    // 코드  
3  |}  
4
```



# 익스텐션 구현

```
1  import UIKit
2
3  extension UIView {
4      func addSubview/views: [UIView]) {
5          for view in views {
6              self.addSubview(view: view)
7          }
8      }
9  }
```

# 익스텐션 vs 상속

	상속	익스텐션
확장	수직확장	수평확장
사용	클래스 타입	모든 타입
재정의	가능	불가능

# 상수와 변수 / 명명법

# 상수와 변수

- ▶ `let` : 상수 선언 키워드
- ▶ `var` : 변수 선언 키워드
- ▶ 상수는 값을 추후에 수정할 수 없으며, 변수는 수정이 가능하다
- ▶ 값의 타입이 명확할 경우 타입 생략이 가능함

# 상수와 변수

```
1 // 상수, 변수 변수명 타입 = 값
2 let constant:String = "변경 가능"
3 var variable:String = "변경 불가능"
4
5 variable = "변수변경"
6 constant = "상수변경" // 오류발생
```

# 명명법

카멜표기법	파스칼표기법
함수, 변수, 상수	클래스, 구조체, 열거형, 확장 (extension)

# 문자열 보간법 및 출력

# 문자열 보간법

- ▶ 프로그램 실행 중 문자열 내에 변수 또는 상수의 값을 표현하기 위해 사용
- ▶ `\()` 을 사용하며 `\()` 안에 값을 대입
- ▶ 변수에 대입 가능



# 출력

- ▶ print  
단순 문자열 출력
- ▶ dump  
인스턴스의 자세한 설명까지 출력

# 실습

```
1 // 상수, 변수 변수명 타입 = 값
2 let constant:String = "변경 가능"
3 var variable:String = "변경 불가능"
4
5 variable = "변수변경"
6 constant = "상수변경" // 오류발생
7
8 let schoolName = "동서대학교"
9
10 print("재학중인 학교는 \(schoolName)입니다")
11
12 struct School {
13 |   let name: String = "동서대학교"
14 }
15
16 let school = School()
17
18 dump(school)
```

# 반복문

# For .. in문

- ▶ 일정 횟수만큼 특정 구문을 반복하기 위해 사용
- ▶ 기존 언어의 for - each와 유사

```
1 let number = [1,2,3,4,5]
2
3 for index in number {
4     print(index)
5 }
```

Array

```
1 let personInfo = ["code" : 90000, "age" : 30]
2
3 for (key, value) in personInfo {
4     print("\(key) \(value)")
5 }
```

Dictionary

# While문

- ▶ 기존 언어의 while문과 유사

# Repeat-while문

- ▶ 기존 언어의 do-while문과 유사

```
1  let numbers: Array<Int> = [1,2,3,4,5]
2  var count = 0
3
4  repeat {
5      print(numbers[count])
6      count += 1
7  } while count < numbers.count
```

# 반복문 실습

# 조건문



# 조건문

- ▶ If - else
- ▶ Switch

# 조건문 실습

옵셔널

# 옵셔널 타입

- ▶ 값이 있을수도 없을 수도 있음을 표현할 때 사용
- ▶ 옵셔널 타입은 대입하지 않는 이상 기본값은 nil

# 옵셔널 추출

- ▶ 옵셔널 바인딩
- ▶ 강제추출

# 옵셔널 바인딩

- ▶ Nil 체크를 하고 안전하게 값을 추출하는 방식
- ▶ 옵셔널 안에 값이 들어있는지 확인한 후 값이 있으면 값을 추출

# 강제 추출

- ▶ Nil 체크 없이 강제로 값을 추출하는 방식
- ▶ 값이 없을 경우 런타임 오류 발생

실습



데이터 타입

# 기본 데이터 타입

- ▶ 참/거짓 : Bool
- ▶ 정수 : Int / UInt
- ▶ 실수 : Float/Double
- ▶ 문자/문자형 : Character / String

# 컬렉션

# 컬렉션 타입

- ▶ Array
- ▶ Dictionary
- ▶ Set

# Array

- ▶ 순서가 있는 리스트 컬렉션

# Dictionary

- ▶ 키와 값의 쌍으로 이루어진 컬렉션

```
1  var dictionary: Dictionary<String, String> = [String:String]()
2
3  // 키에 해당하는 값을 할당하거나 변경
4  dictionary["school"] = "동서대"
5
6  print(dictionary)
7
8  // 키에 해당하는 값 제거
9  dictionary.removeValue(forKey: "school")
10
11 print(dictionary)
```

# Set

- ▶ 순서가 없으며, 값이 유일한 컬렉션
- ▶ 집합으로 이루어져있음
- ▶ Set이라고 명시하지 않으면 배열로 인식

```
1  var intSet: Set<Int> = [1,2,3,4,5]
2
3  print(intSet)
4
5  // 새로운 값 입력
6  intSet.insert(55)
7
8  print(intSet)
9
10 // 값 존재 여부 확인
11 print(intSet.contains(55))
12
13 // 값 삭제
14 intSet.remove(55)
15
16 print(intSet)
```

# 컬렉션 및 데이터타입 실습



# 함수

# 함수

- ▶ Swift는 func 로 시작함

```
1 func 함수명(매개변수1: 이름1, ...) -> 반환형 {  
2     ... // 코드 구현  
3     return 반환값  
4 }
```

# 매개변수 기본값

- ▶ 매개변수에 기본 값을 미리 지정해둘 수 있음
- ▶ 기본값을 가지는 매개변수는 뒤에 두는 것이 가독성에 좋으며, 호출시 생략가능

```
1 func showAction(place: String, action: String = "공부") -> Void {  
2     |    print("나는 \ (place)에서 \ (action) 중입니다.")  
3     |  
4     |  
5     |  
6     |  
7     |  
8     |  
9     |  
10    |  
11    |  
12    |  
13    |  
14    |  
15    |  
16    |  
17    |  
18    |  
19    |  
20    |  
21    |  
22    |  
23    |  
24    |  
25    |  
26    |  
27    |  
28    |  
29    |  
30    |  
31    |  
32    |  
33    |  
34    |  
35    |  
36    |  
37    |  
38    |  
39    |  
40    |  
41    |  
42    |  
43    |  
44    |  
45    |  
46    |  
47    |  
48    |  
49    |  
50    |  
51    |  
52    |  
53    |  
54    |  
55    |  
56    |  
57    |  
58    |  
59    |  
60    |  
61    |  
62    |  
63    |  
64    |  
65    |  
66    |  
67    |  
68    |  
69    |  
70    |  
71    |  
72    |  
73    |  
74    |  
75    |  
76    |  
77    |  
78    |  
79    |  
80    |  
81    |  
82    |  
83    |  
84    |  
85    |  
86    |  
87    |  
88    |  
89    |  
90    |  
91    |  
92    |  
93    |  
94    |  
95    |  
96    |  
97    |  
98    |  
99    |  
100   |  
101   |  
102   |  
103   |  
104   |  
105   |  
106   |  
107   |  
108   |  
109   |  
110   |  
111   |  
112   |  
113   |  
114   |  
115   |  
116   |  
117   |  
118   |  
119   |  
120   |  
121   |  
122   |  
123   |  
124   |  
125   |  
126   |  
127   |  
128   |  
129   |  
130   |  
131   |  
132   |  
133   |  
134   |  
135   |  
136   |  
137   |  
138   |  
139   |  
140   |  
141   |  
142   |  
143   |  
144   |  
145   |  
146   |  
147   |  
148   |  
149   |  
150   |  
151   |  
152   |  
153   |  
154   |  
155   |  
156   |  
157   |  
158   |  
159   |  
160   |  
161   |  
162   |  
163   |  
164   |  
165   |  
166   |  
167   |  
168   |  
169   |  
170   |  
171   |  
172   |  
173   |  
174   |  
175   |  
176   |  
177   |  
178   |  
179   |  
180   |  
181   |  
182   |  
183   |  
184   |  
185   |  
186   |  
187   |  
188   |  
189   |  
190   |  
191   |  
192   |  
193   |  
194   |  
195   |  
196   |  
197   |  
198   |  
199   |  
200   |  
201   |  
202   |  
203   |  
204   |  
205   |  
206   |  
207   |  
208   |  
209   |  
210   |  
211   |  
212   |  
213   |  
214   |  
215   |  
216   |  
217   |  
218   |  
219   |  
220   |  
221   |  
222   |  
223   |  
224   |  
225   |  
226   |  
227   |  
228   |  
229   |  
230   |  
231   |  
232   |  
233   |  
234   |  
235   |  
236   |  
237   |  
238   |  
239   |  
240   |  
241   |  
242   |  
243   |  
244   |  
245   |  
246   |  
247   |  
248   |  
249   |  
250   |  
251   |  
252   |  
253   |  
254   |  
255   |  
256   |  
257   |  
258   |  
259   |  
260   |  
261   |  
262   |  
263   |  
264   |  
265   |  
266   |  
267   |  
268   |  
269   |  
270   |  
271   |  
272   |  
273   |  
274   |  
275   |  
276   |  
277   |  
278   |  
279   |  
280   |  
281   |  
282   |  
283   |  
284   |  
285   |  
286   |  
287   |  
288   |  
289   |  
290   |  
291   |  
292   |  
293   |  
294   |  
295   |  
296   |  
297   |  
298   |  
299   |  
300   |  
301   |  
302   |  
303   |  
304   |  
305   |  
306   |  
307   |  
308   |  
309   |  
310   |  
311   |  
312   |  
313   |  
314   |  
315   |  
316   |  
317   |  
318   |  
319   |  
320   |  
321   |  
322   |  
323   |  
324   |  
325   |  
326   |  
327   |  
328   |  
329   |  
330   |  
331   |  
332   |  
333   |  
334   |  
335   |  
336   |  
337   |  
338   |  
339   |  
340   |  
341   |  
342   |  
343   |  
344   |  
345   |  
346   |  
347   |  
348   |  
349   |  
350   |  
351   |  
352   |  
353   |  
354   |  
355   |  
356   |  
357   |  
358   |  
359   |  
360   |  
361   |  
362   |  
363   |  
364   |  
365   |  
366   |  
367   |  
368   |  
369   |  
370   |  
371   |  
372   |  
373   |  
374   |  
375   |  
376   |  
377   |  
378   |  
379   |  
380   |  
381   |  
382   |  
383   |  
384   |  
385   |  
386   |  
387   |  
388   |  
389   |  
390   |  
391   |  
392   |  
393   |  
394   |  
395   |  
396   |  
397   |  
398   |  
399   |  
400   |  
401   |  
402   |  
403   |  
404   |  
405   |  
406   |  
407   |  
408   |  
409   |  
410   |  
411   |  
412   |  
413   |  
414   |  
415   |  
416   |  
417   |  
418   |  
419   |  
420   |  
421   |  
422   |  
423   |  
424   |  
425   |  
426   |  
427   |  
428   |  
429   |  
430   |  
431   |  
432   |  
433   |  
434   |  
435   |  
436   |  
437   |  
438   |  
439   |  
440   |  
441   |  
442   |  
443   |  
444   |  
445   |  
446   |  
447   |  
448   |  
449   |  
450   |  
451   |  
452   |  
453   |  
454   |  
455   |  
456   |  
457   |  
458   |  
459   |  
460   |  
461   |  
462   |  
463   |  
464   |  
465   |  
466   |  
467   |  
468   |  
469   |  
470   |  
471   |  
472   |  
473   |  
474   |  
475   |  
476   |  
477   |  
478   |  
479   |  
480   |  
481   |  
482   |  
483   |  
484   |  
485   |  
486   |  
487   |  
488   |  
489   |  
490   |  
491   |  
492   |  
493   |  
494   |  
495   |  
496   |  
497   |  
498   |  
499   |  
500   |  
501   |  
502   |  
503   |  
504   |  
505   |  
506   |  
507   |  
508   |  
509   |  
510   |  
511   |  
512   |  
513   |  
514   |  
515   |  
516   |  
517   |  
518   |  
519   |  
520   |  
521   |  
522   |  
523   |  
524   |  
525   |  
526   |  
527   |  
528   |  
529   |  
530   |  
531   |  
532   |  
533   |  
534   |  
535   |  
536   |  
537   |  
538   |  
539   |  
540   |  
541   |  
542   |  
543   |  
544   |  
545   |  
546   |  
547   |  
548   |  
549   |  
550   |  
551   |  
552   |  
553   |  
554   |  
555   |  
556   |  
557   |  
558   |  
559   |  
560   |  
561   |  
562   |  
563   |  
564   |  
565   |  
566   |  
567   |  
568   |  
569   |  
570   |  
571   |  
572   |  
573   |  
574   |  
575   |  
576   |  
577   |  
578   |  
579   |  
580   |  
581   |  
582   |  
583   |  
584   |  
585   |  
586   |  
587   |  
588   |  
589   |  
590   |  
591   |  
592   |  
593   |  
594   |  
595   |  
596   |  
597   |  
598   |  
599   |  
600   |  
601   |  
602   |  
603   |  
604   |  
605   |  
606   |  
607   |  
608   |  
609   |  
610   |  
611   |  
612   |  
613   |  
614   |  
615   |  
616   |  
617   |  
618   |  
619   |  
620   |  
621   |  
622   |  
623   |  
624   |  
625   |  
626   |  
627   |  
628   |  
629   |  
630   |  
631   |  
632   |  
633   |  
634   |  
635   |  
636   |  
637   |  
638   |  
639   |  
640   |  
641   |  
642   |  
643   |  
644   |  
645   |  
646   |  
647   |  
648   |  
649   |  
650   |  
651   |  
652   |  
653   |  
654   |  
655   |  
656   |  
657   |  
658   |  
659   |  
660   |  
661   |  
662   |  
663   |  
664   |  
665   |  
666   |  
667   |  
668   |  
669   |  
670   |  
671   |  
672   |  
673   |  
674   |  
675   |  
676   |  
677   |  
678   |  
679   |  
680   |  
681   |  
682   |  
683   |  
684   |  
685   |  
686   |  
687   |  
688   |  
689   |  
690   |  
691   |  
692   |  
693   |  
694   |  
695   |  
696   |  
697   |  
698   |  
699   |  
700   |  
701   |  
702   |  
703   |  
704   |  
705   |  
706   |  
707   |  
708   |  
709   |  
710   |  
711   |  
712   |  
713   |  
714   |  
715   |  
716   |  
717   |  
718   |  
719   |  
720   |  
721   |  
722   |  
723   |  
724   |  
725   |  
726   |  
727   |  
728   |  
729   |  
730   |  
731   |  
732   |  
733   |  
734   |  
735   |  
736   |  
737   |  
738   |  
739   |  
740   |  
741   |  
742   |  
743   |  
744   |  
745   |  
746   |  
747   |  
748   |  
749   |  
750   |  
751   |  
752   |  
753   |  
754   |  
755   |  
756   |  
757   |  
758   |  
759   |  
760   |  
761   |  
762   |  
763   |  
764   |  
765   |  
766   |  
767   |  
768   |  
769   |  
770   |  
771   |  
772   |  
773   |  
774   |  
775   |  
776   |  
777   |  
778   |  
779   |  
780   |  
781   |  
782   |  
783   |  
784   |  
785   |  
786   |  
787   |  
788   |  
789   |  
790   |  
791   |  
792   |  
793   |  
794   |  
795   |  
796   |  
797   |  
798   |  
799   |  
800   |  
801   |  
802   |  
803   |  
804   |  
805   |  
806   |  
807   |  
808   |  
809   |  
810   |  
811   |  
812   |  
813   |  
814   |  
815   |  
816   |  
817   |  
818   |  
819   |  
820   |  
821   |  
822   |  
823   |  
824   |  
825   |  
826   |  
827   |  
828   |  
829   |  
830   |  
831   |  
832   |  
833   |  
834   |  
835   |  
836   |  
837   |  
838   |  
839   |  
840   |  
841   |  
842   |  
843   |  
844   |  
845   |  
846   |  
847   |  
848   |  
849   |  
850   |  
851   |  
852   |  
853   |  
854   |  
855   |  
856   |  
857   |  
858   |  
859   |  
860   |  
861   |  
862   |  
863   |  
864   |  
865   |  
866   |  
867   |  
868   |  
869   |  
870   |  
871   |  
872   |  
873   |  
874   |  
875   |  
876   |  
877   |  
878   |  
879   |  
880   |  
881   |  
882   |  
883   |  
884   |  
885   |  
886   |  
887   |  
888   |  
889   |  
890   |  
891   |  
892   |  
893   |  
894   |  
895   |  
896   |  
897   |  
898   |  
899   |  
900   |  
901   |  
902   |  
903   |  
904   |  
905   |  
906   |  
907   |  
908   |  
909   |  
910   |  
911   |  
912   |  
913   |  
914   |  
915   |  
916   |  
917   |  
918   |  
919   |  
920   |  
921   |  
922   |  
923   |  
924   |  
925   |  
926   |  
927   |  
928   |  
929   |  
930   |  
931   |  
932   |  
933   |  
934   |  
935   |  
936   |  
937   |  
938   |  
939   |  
940   |  
941   |  
942   |  
943   |  
944   |  
945   |  
946   |  
947   |  
948   |  
949   |  
950   |  
951   |  
952   |  
953   |  
954   |  
955   |  
956   |  
957   |  
958   |  
959   |  
960   |  
961   |  
962   |  
963   |  
964   |  
965   |  
966   |  
967   |  
968   |  
969   |  
970   |  
971   |  
972   |  
973   |  
974   |  
975   |  
976   |  
977   |  
978   |  
979   |  
980   |  
981   |  
982   |  
983   |  
984   |  
985   |  
986   |  
987   |  
988   |  
989   |  
990   |  
991   |  
992   |  
993   |  
994   |  
995   |  
996   |  
997   |  
998   |  
999   |  
1000  |  
1001  |  
1002  |  
1003  |  
1004  |  
1005  |  
1006  |  
1007  |  
1008  |  
1009  |  
1010  |  
1011  |  
1012  |  
1013  |  
1014  |  
1015  |  
1016  |  
1017  |  
1018  |  
1019  |  
1020  |  
1021  |  
1022  |  
1023  |  
1024  |  
1025  |  
1026  |  
1027  |  
1028  |  
1029  |  
1030  |  
1031  |  
1032  |  
1033  |  
1034  |  
1035  |  
1036  |  
1037  |  
1038  |  
1039  |  
1040  |  
1041  |  
1042  |  
1043  |  
1044  |  
1045  |  
1046  |  
1047  |  
1048  |  
1049  |  
1050  |  
1051  |  
1052  |  
1053  |  
1054  |  
1055  |  
1056  |  
1057  |  
1058  |  
1059  |  
1060  |  
1061  |  
1062  |  
1063  |  
1064  |  
1065  |  
1066  |  
1067  |  
1068  |  
1069  |  
1070  |  
1071  |  
1072  |  
1073  |  
1074  |  
1075  |  
1076  |  
1077  |  
1078  |  
1079  |  
1080  |  
1081  |  
1082  |  
1083  |  
1084  |  
1085  |  
1086  |  
1087  |  
1088  |  
1089  |  
1090  |  
1091  |  
1092  |  
1093  |  
1094  |  
1095  |  
1096  |  
1097  |  
1098  |  
1099  |  
1100  |  
1101  |  
1102  |  
1103  |  
1104  |  
1105  |  
1106  |  
1107  |  
1108  |  
1109  |  
1110  |  
1111  |  
1112  |  
1113  |  
1114  |  
1115  |  
1116  |  
1117  |  
1118  |  
1119  |  
1120  |  
1121  |  
1122  |  
1123  |  
1124  |  
1125  |  
1126  |  
1127  |  
1128  |  
1129  |  
1130  |  
1131  |  
1132  |  
1133  |  
1134  |  
1135  |  
1136  |  
1137  |  
1138  |  
1139  |  
1140  |  
1141  |  
1142  |  
1143  |  
1144  |  
1145  |  
1146  |  
1147  |  
1148  |  
1149  |  
1150  |  
1151  |  
1152  |  
1153  |  
1154  |  
1155  |  
1156  |  
1157  |  
1158  |  
1159  |  
1160  |  
1161  |  
1162  |  
1163  |  
1164  |  
1165  |  
1166  |  
1167  |  
1168  |  
1169  |  
1170  |  
1171  |  
1172  |  
1173  |  
1174  |  
1175  |  
1176  |  
1177  |  
1178  |  
1179  |  
1180  |  
1181  |  
1182  |  
1183  |  
1184  |  
1185  |  
1186  |  
1187  |  
1188  |  
1189  |  
1190  |  
1191  |  
1192  |  
1193  |  
1194  |  
1195  |  
1196  |  
1197  |  
1198  |  
1199  |  
1200  |  
1201  |  
1202  |  
1203  |  
1204  |  
1205  |  
1206  |  
1207  |  
1208  |  
1209  |  
1210  |  
1211  |  
1212  |  
1213  |  
1214  |  
1215  |  
1216  |  
1217  |  
1218  |  
1219  |  
1220  |  
1221  |  
1222  |  
1223  |  
1224  |  
1225  |  
1226  |  
1227  |  
1228  |  
1229  |  
1230  |  
1231  |  
1232  |  
1233  |  
1234  |  
1235  |  
1236  |  
1237  |  
1238  |  
1239  |  
1240  |  
1241  |  
1242  |  
1243  |  
1244  |  
1245  |  
1246  |  
1247  |  
1248  |  
1249  |  
1250  |  
1251  |  
1252  |  
1253  |  
1254  |  
1255  |  
1256  |  
1257  |  
1258  |  
1259  |  
1260  |  
1261  |  
1262  |  
1263  |  
1264  |  
1265  |  
1266  |  
1267  |  
1268  |  
1269  |  
1270  |  
1271  |  
1272  |  
1273  |  
1274  |  
1275  |  
1276  |  
1277  |  
1278  |  
1279  |  
1280  |  
1281  |  
1282  |  
1283  |  
1284  |  
1285  |  
1286  |  
1287  |  
1288  |  
1289  |  
1290  |  
1291  |  
1292  |  
1293  |  
1294  |  
1295  |  
1296  |  
1297  |  
1298  |  
1299  |  
1300  |  
1301  |  
1302  |  
1303  |  
1304  |  
1305  |  
1306  |  
1307  |  
1308  |  
1309  |  
1310  |  
1311  |  
1312  |  
1313  |  
1314  |  
1315  |  
1316  |  
1317  |  
1318  |  
1319  |  
1320  |  
1321  |  
1322  |  
1323  |  
1324  |  
1325  |  
1326  |  
1327  |  
1328  |  
1329  |  
1330  |  
1331  |  
1332  |  
1333  |  
1334  |  
1335  |  
1336  |  
1337  |  
1338  |  
1339  |  
1340  |  
1341  |  
1342  |  
1343  |  
1344  |  
1345  |  
1346  |  
1347  |  
1348  |  
1349  |  
1350  |  
1351  |  
1352  |  
1353  |  
1354  |  
1355  |  
1356  |  
1357  |  
1358  |  
1359  |  
1360  |  
1361  |  
1362  |  
1363  |  
1364  |  
1365  |  
1366  |  
1367  |  
1368  |  
1369  |  
1370  |  
1371  |  
1372  |  
1373  |  
1374  |  
1375  |  
1376  |  
1377  |  
1378  |  
1379  |  
1380  |  
1381  |  
1382  |  
1383  |  
1384  |  
1385  |  
1386  |  
1387  |  
1388  |  
1389  |  
1390  |  
1391  |  
1392  |  
1393  |  
1394  |  
1395  |  
1396  |  
1397  |  
1398  |  
1399  |  
1400  |  
1401  |  
1402  |  
1403  |  
1404  |  
1405  |  
1406  |  
1407  |  
1408  |  
1409  |  
1410  |  
1411  |  
1412  |  
1413  |  
1414  |  
1415  |  
1416  |  
1417  |  
1418  |  
1419  |  
1420  |  
1421  |  
1422  |  
1423  |  
1424  |  
1425  |  
1426  |  
1427  |  
1428  |  
1429  |  
1430  |  
1431  |  
1432  |  
1433  |  
1434  |  
1435  |  
1436  |  
1437  |  
1438  |  
1439  |  
1440  |  
1441  |  
1442  |  
1443  |  
1444  |  
1445  |  
1446  |  
1447  |  
1448  |  
1449  |  
1450  |  
1451  |  
1452  |  
1453  |  
1454  |  
1455  |  
1456  |  
1457  |  
1458  |  
1459  |  
1460  |  
1461  |  
1462  |  
1463  |  
1464  |  
1465  |  
1466  |  
1467  |  
1468  |  
1469  |  
1470  |  
1471  |  
1472  |  
1473  |  
1474  |  
1475  |  
1476  |  
1477  |  
1478  |  
1479 
```

# 전달인자 레이블

- ▶ 함수를 호출할 때 매개변수의 역할을 명확히 표현할 때 사용
- ▶ 동일한 이름의 함수를 중복으로 생성할 수 있음

```
1 func showAction(place: String, action situation: String = "공부") -> Void {  
2     |    print("나는 \ (place)에서 \ (situation) 중입니다.")  
3     |  
4     |  
5     |  
6     |  
7     |  
8     |  
9     |  
10    |  
11    |  
12    |  
13    |  
14    |  
15    |  
16    |  
17    |  
18    |  
19    |  
20    |  
21    |  
22    |  
23    |  
24    |  
25    |  
26    |  
27    |  
28    |  
29    |  
30    |  
31    |  
32    |  
33    |  
34    |  
35    |  
36    |  
37    |  
38    |  
39    |  
40    |  
41    |  
42    |  
43    |  
44    |  
45    |  
46    |  
47    |  
48    |  
49    |  
50    |  
51    |  
52    |  
53    |  
54    |  
55    |  
56    |  
57    |  
58    |  
59    |  
60    |  
61    |  
62    |  
63    |  
64    |  
65    |  
66    |  
67    |  
68    |  
69    |  
70    |  
71    |  
72    |  
73    |  
74    |  
75    |  
76    |  
77    |  
78    |  
79    |  
80    |  
81    |  
82    |  
83    |  
84    |  
85    |  
86    |  
87    |  
88    |  
89    |  
90    |  
91    |  
92    |  
93    |  
94    |  
95    |  
96    |  
97    |  
98    |  
99    |  
100   |  
101   |  
102   |  
103   |  
104   |  
105   |  
106   |  
107   |  
108   |  
109   |  
110   |  
111   |  
112   |  
113   |  
114   |  
115   |  
116   |  
117   |  
118   |  
119   |  
120   |  
121   |  
122   |  
123   |  
124   |  
125   |  
126   |  
127   |  
128   |  
129   |  
130   |  
131   |  
132   |  
133   |  
134   |  
135   |  
136   |  
137   |  
138   |  
139   |  
140   |  
141   |  
142   |  
143   |  
144   |  
145   |  
146   |  
147   |  
148   |  
149   |  
150   |  
151   |  
152   |  
153   |  
154   |  
155   |  
156   |  
157   |  
158   |  
159   |  
160   |  
161   |  
162   |  
163   |  
164   |  
165   |  
166   |  
167   |  
168   |  
169   |  
170   |  
171   |  
172   |  
173   |  
174   |  
175   |  
176   |  
177   |  
178   |  
179   |  
180   |  
181   |  
182   |  
183   |  
184   |  
185   |  
186   |  
187   |  
188   |  
189   |  
190   |  
191   |  
192   |  
193   |  
194   |  
195   |  
196   |  
197   |  
198   |  
199   |  
200   |  
201   |  
202   |  
203   |  
204   |  
205   |  
206   |  
207   |  
208   |  
209   |  
210   |  
211   |  
212   |  
213   |  
214   |  
215   |  
216   |  
217   |  
218   |  
219   |  
220   |  
221   |  
222   |  
223   |  
224   |  
225   |  
226   |  
227   |  
228   |  
229   |  
230   |  
231   |  
232   |  
233   |  
234   |  
235   |  
236   |  
237   |  
238   |  
239   |  
240   |  
241   |  
242   |  
243   |  
244   |  
245   |  
246   |  
247   |  
248   |  
249   |  
250   |  
251   |  
252   |  
253   |  
254   |  
255   |  
256   |  
257   |  
258   |  
259   |  
260   |  
261   |  
262   |  
263   |  
264   |  
265   |  
266   |  
267   |  
268   |  
269   |  
270   |  
271   |  
272   |  
273   |  
274   |  
275   |  
276   |  
277   |  
278   |  
279   |  
280   |  
281   |  
282   |  
283   |  
284   |  
285   |  
286   |  
287   |  
288   |  
289   |  
290   |  
291   |  
292   |  
293   |  
294   |  
295   |  
296   |  
297   |  
298   |  
299   |  
300   |  
301   |  
302   |  
303   |  
304   |  
305   |  
306   |  
307   |  
308   |  
309   |  
310   |  
311   |  
312   |  
313   |  
314   |  
315   |  
316   |  
317   |  
318   |  
319   |  
320   |  
321   |  
322   |  
323   |  
324   |  
325   |  
326   |  
327   |  
328   |  
329   |  
330   |  
331   |  
332   |  
333   |  
334   |  
335   |  
336   |  
337   |  
338   |  
339   |  
340   |  
341   |  
342   |  
343   |  
344   |  
345   |  
346   |  
347   |  
348   |  
349   |  
350   |  
351   |  
352   |  
353   |  
354   |  
355   |  
356   |  
357   |  
358   |  
359   |  
360   |  
361   |  
362   |  
363   |  
364   |  
365   |  
366   |  
367   |  
368   |  
369   |  
370   |  
371   |  
372   |  
373   |  
374   |  
375   |  
376   |  
377   |  
378   |  
379   |  
380   |  
381   |  
382   |  
383   |  
384   |  
385   |  
386   |  
387   |  
388   |  
389   |  
390   |  
391   |  
392   |  
393   |  
394   |  
395   |  
396   |  
397   |  
398   |  
399   |  
400   |  
401   |  
402   |  
403   |  
404   |  
405   |  
406   |  
407   |  
408   |  
409   |  
410   |  
411   |  
412   |  
413   |  
414   |  
415   |  
416   |  
417   |  
418   |  
419   |  
420   |  
421   |  
422   |  
423   |  
424   |  
425   |  
426   |  
427   |  
428   |  
429   |  
430   |  
431   |  
432   |  
433   |  
434   |  
435   |  
436   |  
437   |  
438   |  
439   |  
440   |  
441   |  
442   |  
443   |  
444   |  
445   |  
446   |  
447   |  
448   |  
449   |  
450   |  
451   |  
452   |  
453   |  
454   |  
455   |  
456   |  
457   |  
458   |  
459   |  
460   |  
461   |  
462   |  
463   |  
464   |  
465   |  
466   |  
467   |  
468   |  
469   |  
470   |  
471   |  
472   |  
473   |  
474   |  
475   |  
476   |  
477   |  
478   |  
479   |  
480   |  
481   |  
482   |  
483   |  
484   |  
485   |  
486   |  
487   |  
488   |  
489   |  
490   |  
491   |  
492   |  
493   |  
494   |  
495   |  
496   |  
497   |  
498   |  
499   |  
500   |  
501   |  
502   |  
503   |  
504   |  
505   |  
506   |  
507   |  
508   |  
509   |  
510   |  
511   |  
512   |  
513   |  
514   |  
515   |  
516   |  
517   |  
518   |  
519   |  
520   |  
521   |  
522   |  
523   |  
524   |  
525   |  
526   |  
527   |  
528   |  
529   |  
530   |  
531   |  
532   |  
533   |  
534   |  
535   |  
536   |  
537   |  
538   |  
539   |  
540   |  
541   |  
542   |  
543   |  
544   |  
545   |  
546   |  
547   |  
548   |  
549   |  
550   |  
551   |  
552   |  
553   |  
554   |  
555   |  
556   |  
557   |  
558   |  
559   |  
560   |  
561   |  
562   |  
563   |  
564   |  
565   |  
566   |  
567   |  
568   |  
569   |  
570   |  
571   |  
572   |  
573   |  
574   |  
575   |  
576   |  
577   |  
578   |  
579   |  
580   |  
581   |  
582   |  
583   |  
584   |  
585   |  
586   |  
587   |  
588   |  
589   |  
590   |  
591   |  
592   |  
593   |  
594   |  
595   |  
596   |  
597   |  
598   |  
599   |  
600   |  
601   |  
602   |  
603   |  
604   |  
605   |  
606   |  
607   |  
608   |  
609   |  
610   |  
611   |  
612   |  
613   |  
614   |  
615   |  
616   |  
617   |  
618   |  
619   |  
620   |  
621   |  
622   |  
623   |  
624   |  
625   |  
626   |  
627   |  
628   |  
629   |  
630   |  
631   |  
632   |  
633   |  
634   |  
635   |  
636   |  
637   |  
638   |  
639   |  
640   |  
641   |  
642   |  
643   |  
644   |  
645   |  
646   |  
647   |  
648   |  
649   |  
650   |  
651   |  
652   |  
653   |  
654   |  
655   |  
656   |  
657   |  
658   |  
659   |  
660   |  
661   |  
662   |  
663   |  
664   |  
665   |  
666   |  
667   |  
668   |  
669   |  
670   |  
671   |  
672   |  
673   |  
674   |  
675   |  
676   |  
677   |  
678   |  
679   |  
680   |  
681   |  
682   |  
683   |  
684   |  
685   |  
686   |  
687   |  
688   |  
689   |  
690   |  
691   |  
692   |  
693   |  
694   |  
695   |  
696   |  
697   |  
698   |  
699   |  
700   |  
701   |  
702   |  
703   |  
704   |  
705   |  
706   |  
707   |  
708   |  
709   |  
710   |  
711   |  
712   |  
713   |  
714   |  
715   |  
716   |  
717   |  
718   |  
719   |  
720   |  
721   |  
722   |  
723   |  
724   |  
725   |  
726   |  
727   |  
728   |  
729   |  
730   |  
731   |  
732   |  
733   |  
734   |  
735   |  
736   |  
737   |  
738   |  
739   |  
740   |  
741   |  
742   |  
743   |  
744   |  
745   |  
746   |  
747   |  
748   |  
749   |  
750   |  
751   |  
752   |  
753   |  
754   |  
755   |  
756   |  
757   |  
758   |  
759   |  
760   |  
761   |  
762   |  
763   |  
764   |  
765   |  
766   |  
767   |  
768   |  
769   |  
770   |  
771   |  
772   |  
773   |  
774   |  
775   |  
776   |  
777   |  
778   |  
779   |  
780   |  
781   |  
782   |  
783   |  
784   |  
785   |  
786   |  
787   |  
788   |  
789   |  
790   |  
791   |  
792   |  
793   |  
794   |  
795   |  
796   |  
797   |  
798   |  
799   |  
800   |  
801   |  
802   |  
803   |  
804   |  
805   |  
806   |  
807   |  
808   |  
809   |  
810   |  
811   |  
812   |  
813   |  
814   |  
815   |  
816   |  
817   |  
818   |  
819   |  
820   |  
821   |  
822   |  
823   |  
824   |  
825   |  
826   |  
827   |  
828   |  
829   |  
830   |  
831   |  
832   |  
833   |  
834   |  
835   |  
836   |  
837   |  
838   |  
839   |  
840   |  
841   |  
842   |  
843   |  
844   |  
845   |  
846   |  
847   |  
848   |  
849   |  
850   |  
851   |  
852   |  
853   |  
854   |  
855   |  
856   |  
857   |  
858   |  
859   |  
860   |  
861   |  
862   |  
863   |  
864   |  
865   |  
866   |  
867   |  
868   |  
869   |  
870   |  
871   |  
872   |  
873   |  
874   |  
875   |  
876   |  
877   |  
878   |  
879   |  
880   |  
881   |  
882   |  
883   |  
884   |  
885   |  
886   |  
887   |  
888   |  
889   |  
890   |  
891   |  
892   |  
893   |  
894   |  
895   |  
896   |  
897   |  
898   |  
899   |  
900   |  
901   |  
902   |  
903   |  
904   |  
905   |  
906   |  
907   |  
908   |  
909   |  
910   |  
911   |  
912   |  
913   |  
914   |  
915   |  
916   |  
917   |  
918   |  
919   |  
920   |  
921   |  
922   |  
923   |  
924   |  
925   |  
926   |  
927   |  
928   |  
929   |  
930   |  
931   |  
932   |  
933   |  
934   |  
935   |  
936   |  
937   |  
938   |  
939   |  
940   |  
941   |  
942   |  
943   |  
944   |  
945   |  
946   |  
947   |  
948   |  
949   |  
950   |  
951   |  
952   |  
953   |  
954   |  
955   |  
956   |  
957   |  
958   |  
959   |  
960   |  
961   |  
962   |  
963   |  
964   |  
965   |  
966   |  
967   |  
968   |  
969   |  
970   |  
971   |  
972   |  
973   |  
974   |  
975   |  
976   |  
977   |  
978   |  
979   |  
980   |  
981   |  
982   |  
983   |  
984   |  
985   |  
986   |  
987   |  
988   |  
989   |  
990   |  
991   |  
992   |  
993   |  
994   |  
995   |  
996   |  
997   |  
998   |  
999   |  
1000  |  
1001  |  
1002  |  
1003  |  
1004  |  
1005  |  
1006  |  
1007  |  
1008  |  
1009  |  
1010  |  
1011  |  
1012  |  
1013  |  
1014  |  
1015  |  
1016  |  
1017  |  
1018  |  
1019  |  
1020  |  
1021  |  
1022  |  
1023  |  
1024  |  
1025  |  
1026  |  
1027  |  
1028  |  
1029  |  
1030  |  
1031  |  
1032  |  
1033  |  
1034  |  
1035  |  
1036  |  
1037  |  
1038  |  
1039  |  
1040  |  
1041  |  
1042  |  
1043  |  
1044  |  
1045  |  
1046  |  
1047  |  
1048  |  
1049  |  
1050  |  
1051  |  
1052  |  
1053  |  
1054  |  
1055  |  
1056  |  
1057  |  
1058  |  
1059  |  
1060  |  
1061  |  
1062  |  
1063  |  
1064  |  
1065  |  
1066  |  
1067  |  
1068  |  
1069  |  
1070  |  
1071  |  
1072  |  
1073  |  
1074  |  
1075  |  
1076  |  
1077  |  
1078  |  
1079  |  
1080  |  
1081  |  
1082  |  
1083  |  
1084  |  
1085  |  
1086  |  
1087  |  
1088  |  
1089  |  
1090  |  
1091  |  
1092  |  
1093  |  
1094  |  
1095  |  
1096  |  
1097  |  
1098  |  
1099  |  
1100  |  
1101  |  
1102  |  
1103  |  
1104  |  
1105  |  
1106  |  
1107  |  
1108  |  
1109  |  
1110  |  
1111  |  
1112  |  
1113  |  
1114  |  
1115  |  
1116  |  
1117  |  
1118  |  
1119  |  
1120  |  
1121  |  
1122  |  
1123  |  
1124  |  
1125  |  
1126  |  
1127  |  
1128  |  
1129  |  
1130  |  
1131  |  
1132  |  
1133  |  
1134  |  
1135  |  
1136  |  
1137  |  
1138  |  
1139  |  
1140  |  
1141  |  
1142  |  
1143  |  
1144  |  
1145  |  
1146  |  
1147  |  
1148  |  
1149  |  
1150  |  
1151  |  
1152  |  
1153  |  
1154  |  
1155  |  
1156  |  
1157  |  
1158  |  
1159  |  
1160  |  
1161  |  
1162  |  
1163  |  
1164  |  
1165  |  
1166  |  
1167  |  
1168  |  
1169  |  
1170  |  
1171  |  
1172  |  
1173  |  
1174  |  
1175  |  
1176  |  
1177  |  
1178  |  
1179  |  
1180  |  
1181  |  
1182  |  
1183  |  
1184  |  
1185  |  
1186  |  
1187  |  
1188  |  
1189  |  
1190  |  
1191  |  
1192  |  
1193  |  
1194  |  
1195  |  
1196  |  
1197  |  
1198  |  
1199  |  
1200  |  
1201  |  
1202  |  
1203  |  
1204  |  
1205  |  
1206  |  
1207  |  
1208  |  
1209  |  
1210  |  
1211  |  
1212  |  
1213  |  
1214  |  
1215  |  
1216  |  
1217  |  
1218  |  
1219  |  
1220  |  
1221  |  
1222  |  
1223  |  
1224  |  
1225  |  
1226  |  
1227  |  
1228  |  
1229  |  
1230  |  
1231  |  
1232  |  
1233  |  
1234  |  
1235  |  
1236  |  
1237  |  
1238  |  
1239  |  
1240  |  
1241  |  
1242  |  
1243  |  
1244  |  
1245  |  
1246  |  
1247  |  
1248  |  
1249  |  
1250  |  
1251  |  
1252  |  
1253  |  
1254  |  
1255  |  
1256  |  
1257  |  
1258  |  
1259  |  
1260  |  
1261  |  
1262  |  
1263  |  
1264  |  
1265  |  
1266  |  
1267  |  
1268  |  
1269  |  
1270  |  
1271  |  
1272  |  
1273  |  
1274  |  
1275  |  
1276  |  
1277  |  
1278  |  
1279  |  
1280  |  
1281  |  
1282  |  
1283  |  
1284  |  
1285  |  
1286  |  
1287  |  
1288  |  
1289  |  
1290  |  
1291  |  
1292  |  
1293  |  
1294  |  
1295  |  
1296  |  
1297  |  
1298  |  
1299  |  
1300  |  
1301  |  
1302  |  
1303  |  
1304  |  
1305  |  
1306  |  
1307  |  
1308  |  
1309  |  
1310  |  
1311  |  
1312  |  
1313  |  
1314  |  
1315  |  
1316  |  
1317  |  
1318  |  
1319  |  
1320  |  
1321  |  
1322  |  
1323  |  
1324  |  
1325  |  
1326  |  
1327  |  
1328  |  
1329  |  
1330  |  
1331  |  
1332  |  
1333  |  
1334  |  
1335  |  
1336  |  
1337  |  
1338  |  
1339  |  
1340  |  
1341  |  
1342  |  
1343  |  
1344  |  
1345  |  
1346  |  
1347  |  
1348  |  
1349  |  
1350  |  
1351  |  
1352  |  
1353  |  
1354  |  
1355  |  
1356  |  
1357  |  
1358  |  
1359  |  
1360  |  
1361  |  
1362  |  
1363  |  
1364  |  
1365  |  
1366  |  
1367  |  
1368  |  
1369  |  
1370  |  
1371  |  
1372  |  
1373  |  
1374  |  
1375  |  
1376  |  
1377  |  
1378  |  
1379  |  
1380  |  
1381  |  
1382  |  
1383  |  
1384  |  
1385  |  
1386  |  
1387  |  
1388  |  
1389  |  
1390  |  
1391  |  
1392  |  
1393  |  
1394  |  
1395  |  
1396  |  
1397  |  
1398  |  
1399  |  
1400  |  
1401  |  
1402  |  
1403  |  
1404  |  
1405  |  
1406  |  
1407  |  
1408  |  
1409  |  
1410  |  
1411  |  
1412  |  
1413  |  
1414  |  
1415  |  
1416  |  
1417  |  
1418  |  
1419  |  
1420  |  
1421  |  
1422  |  
1423  |  
1424  |  
1425  |  
1426  |  
1427  |  
1428  |  
1429  |  
1430  |  
1431  |  
1432  |  
1433  |  
1434  |  
1435  |  
1436  |  
1437  |  
1438  |  
1439  |  
1440  |  
1441  |  
1442  |  
1443  |  
1444  |  
1445  |  
1446  |  
1447  |  
1448  |  
1449  |  
1450  |  
1451  |  
1452  |  
1453  |  
1454  |  
1455  |  
1456  |  
1457  |  
1458  |  
1459  |  
1460  |  
1461  |  
1462  |  
1463  |  
1464  |  
1465  |  
1466  |  
1467  |  
1468  |  
1469  |  
1470  |  
1471  |  
1472  |  
1473  |  
1474  |  
1475  |  
1476  |  
1477  |  
1478  |  
1479  |  
148
```

# 함수 실습

# 스토리보드

# 스토리보드

- ▶ iOS 또는 tvOS 앱을 위한 그래픽 기반의 UI를 구성하고 관리하는 프레임워크

# 스토리보드 생성 실습



The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern, layered effect. The shapes are concentrated on the left and right sides of the frame, leaving a central white area.

# UIKit

# UIKit

- ▶ iOS 또는 tvOS 앱을 위한 그래픽 기반의 UI를 구성하고 관리하는 프레임워크
- ▶ 제스처, 애니메이션, 그림 그리기, 이미지 및 텍스트 처리 등 다양한 사용자 이벤트들을 처리
- ▶ 화면을 구성하기 위해 필수적으로 상속해야함

The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern, layered effect. The word "View" is centered in a green, sans-serif font.

View

# View

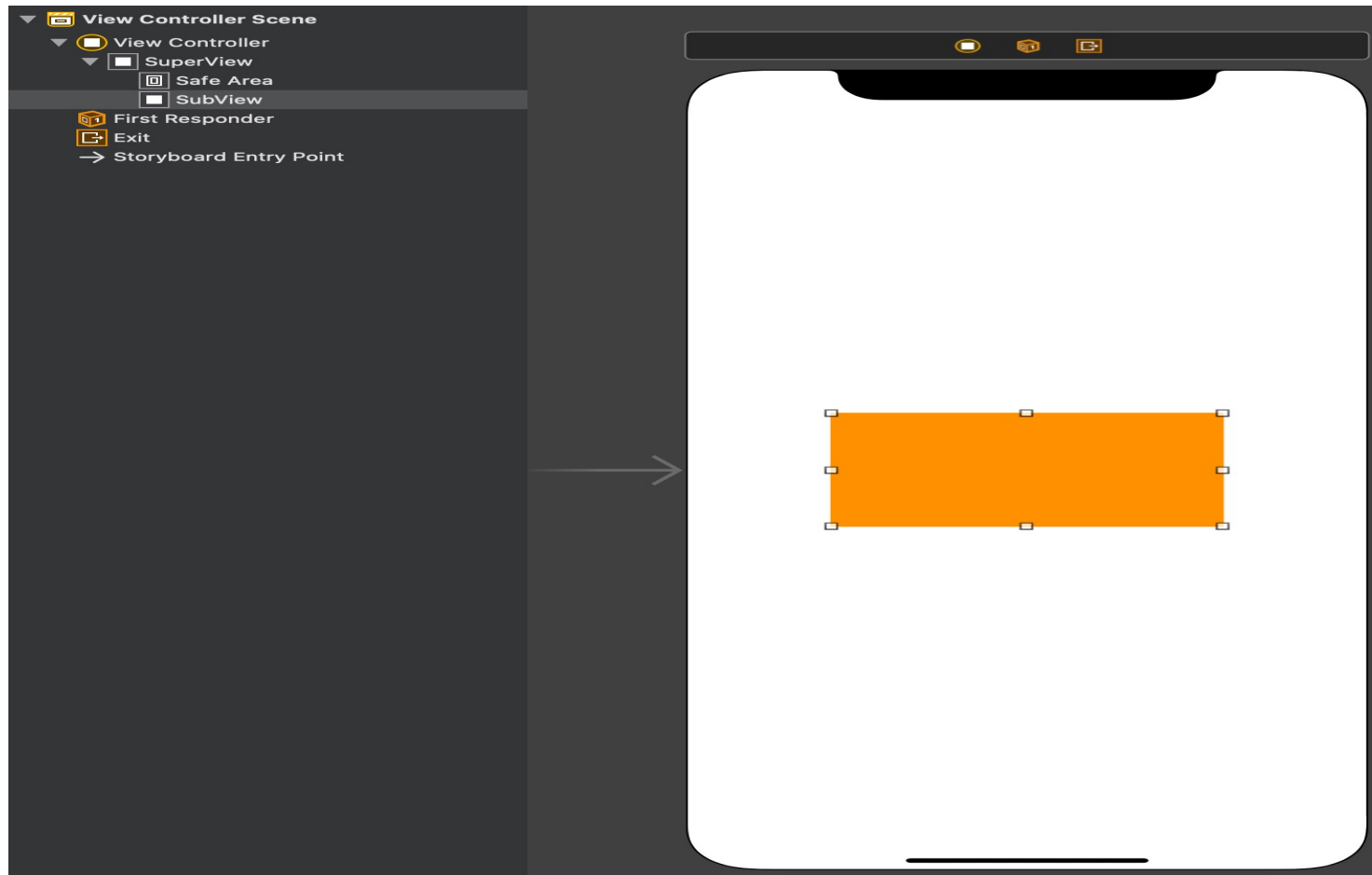
- ▶ iOS 어플리케이션 화면에서 보는 내용은 윈도우와 뷰를 사용하여 나타냄
- ▶ UIView 클래스나 UIView의 하위클래스의 인스턴스로 윈도우의 한 영역에서 콘텐츠 표시
- ▶ 제스처 혹은 터치 이벤트 처리
- ▶ iOS 어플리케이션 화면에서 보는 내용은 윈도우와 뷰를 사용하여 나타냄

# 뷰의 계층구조와 서브뷰

- ▶ 다른 뷰를 위한 컨테이너로서의 역할도 병행
- ▶ 자식 뷰는 서브 뷰, 부모 뷰는 슈퍼 뷰로 불림

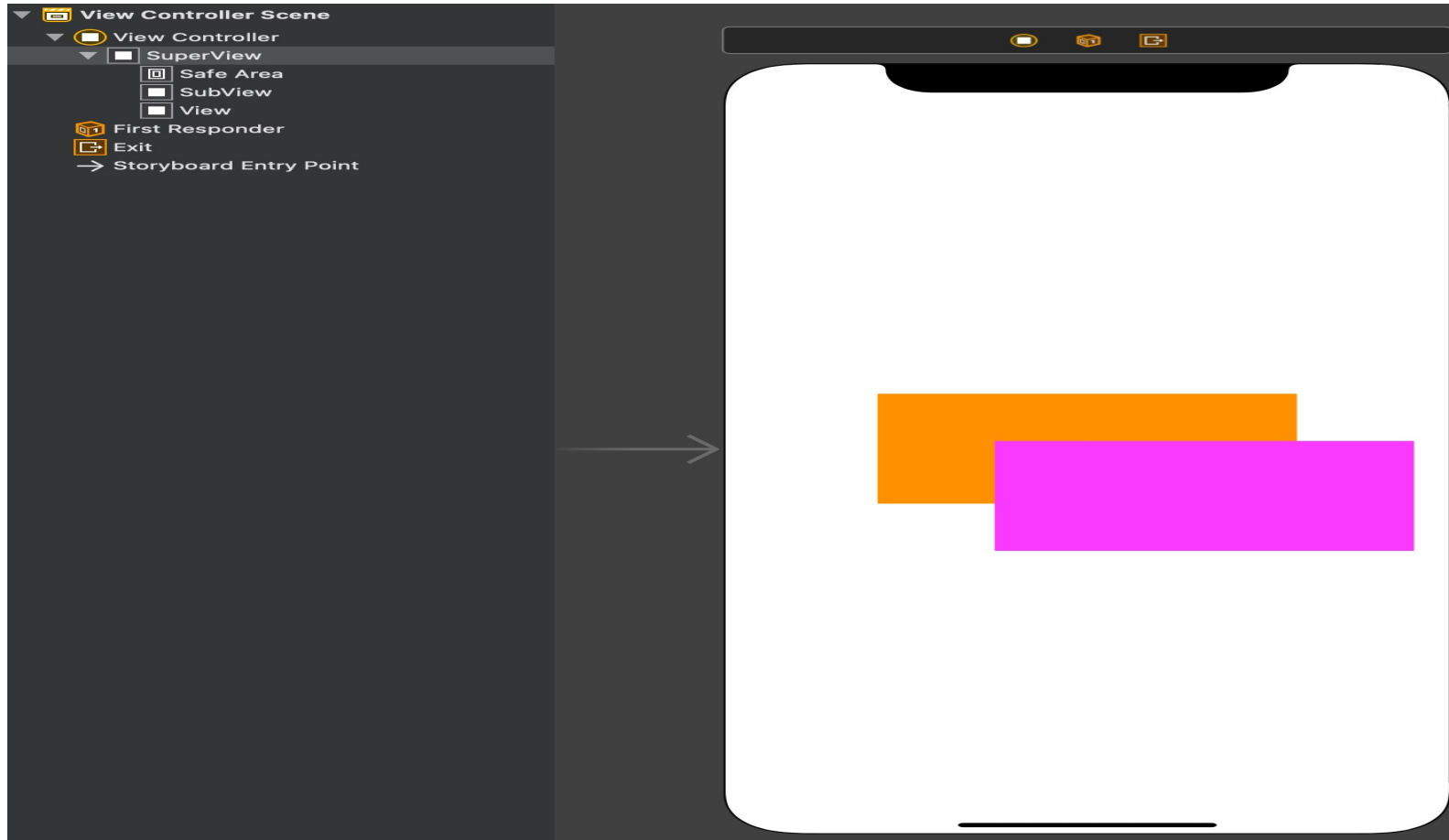
# 뷰의 계층구조와 서브뷰

- ▶ 슈퍼뷰와 서브뷰 관계에 있을 경우 슈퍼뷰가 서브뷰에 가려짐



# 뷰의 계층구조와 서브뷰

- ▶ 하나의 슈퍼뷰에 두개 이상의 서브뷰가 겹치게 된다면 나중에 추가된 서브뷰가 맨 위에 보여짐



# 뷰 컨트롤러



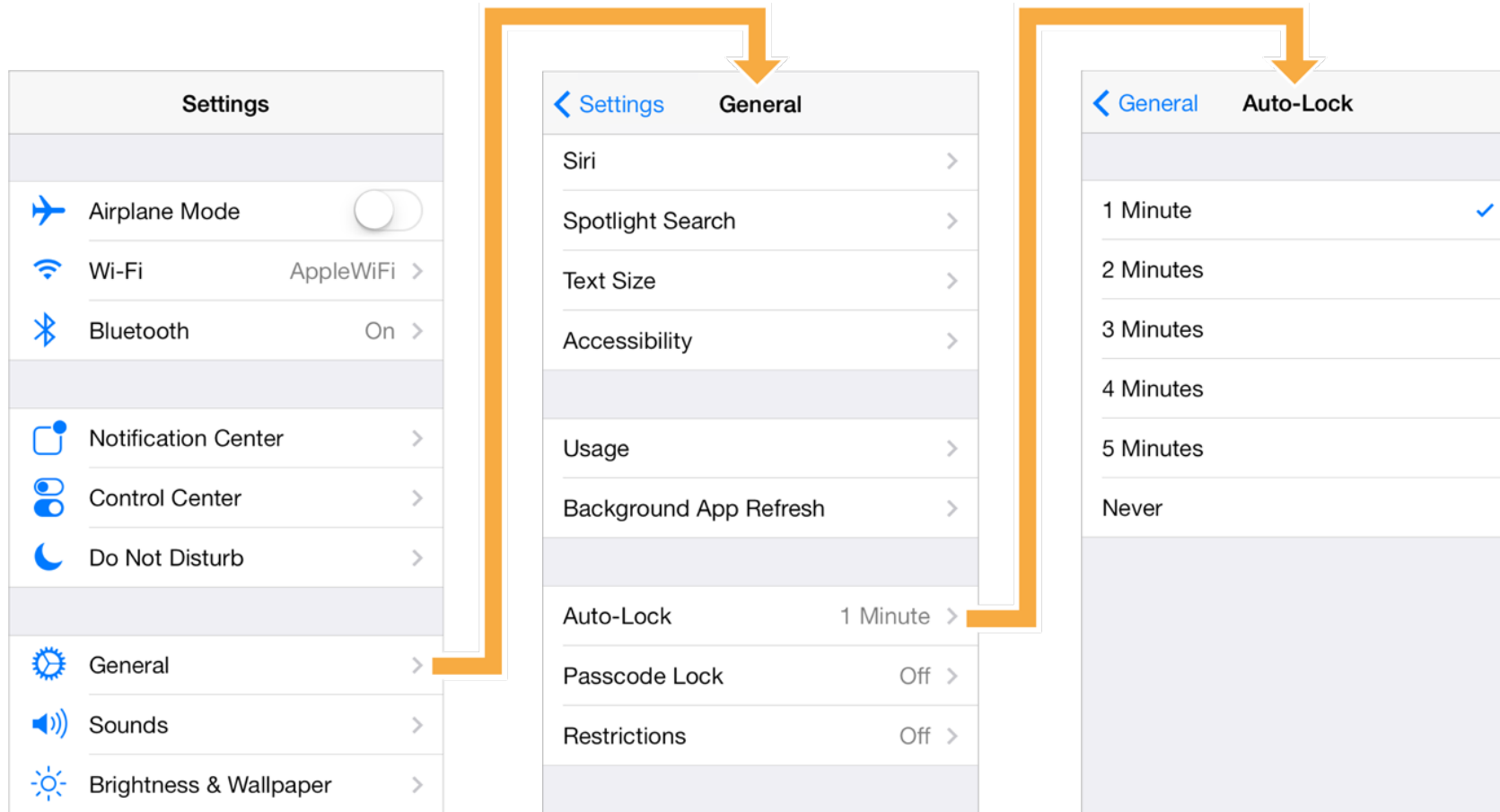
# 뷰 컨트롤러

- ▶ UIKit을 사용하는 앱의 인터페이스를 관리하기 위한 도구
- ▶ 뷰 컨트롤러는 하나의 루트 뷰 만을 관리하고 해당 루트 뷰가 여러 개의 서브 뷰를 가지는 방식으로 구성

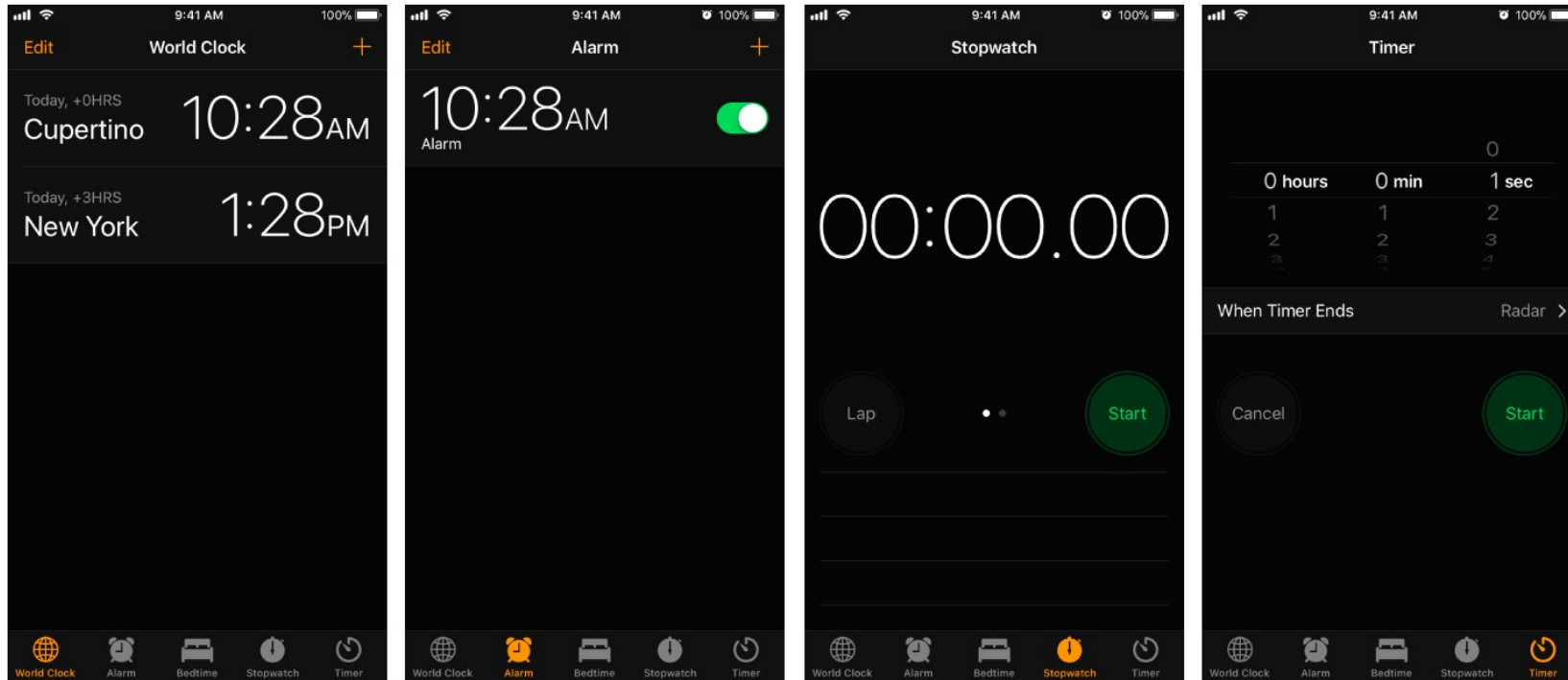
# 뷰 컨트롤러의 종류

- ▶ 네비게이션 컨트롤러
- ▶ 탭 바 컨트롤러
- ▶ 테이블뷰 컨트롤러
- ▶ 그 외 여러 컨트롤러가 존재

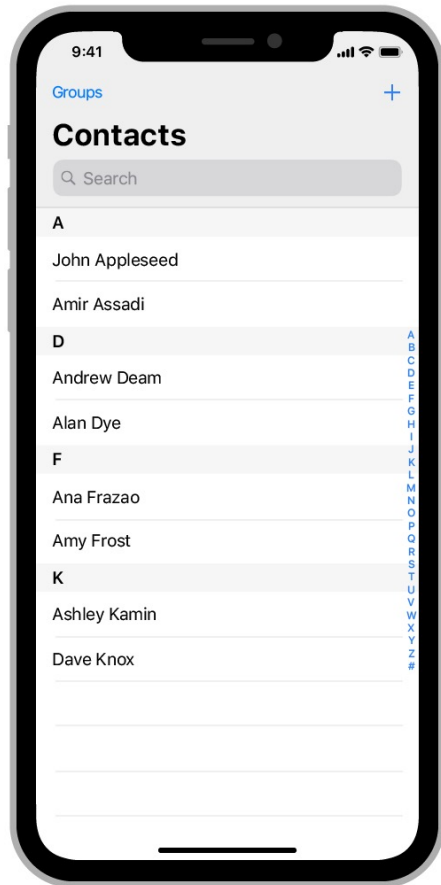
# 네비게이션 컨트롤러



# 탭바 컨트롤러



# 테이블뷰 컨트롤러



# 뷰 컨트롤러 생성 및 연결

- ▶ 뷰 컨트롤러에 사용할 파일을 새로 생성
- ▶ 스토리보드에서 뷰 컨트롤러를 추가한 후 연결

# 뷰 컨트롤러 실습

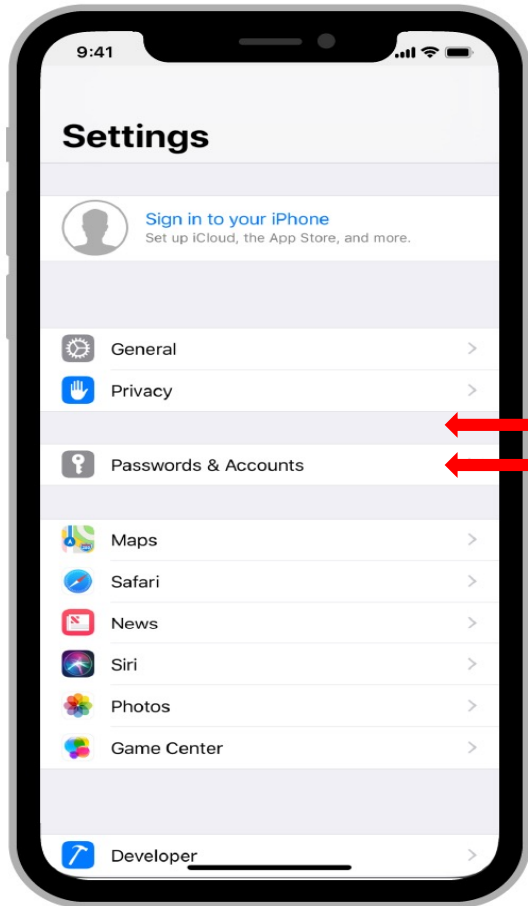
테이블뷰



# UITableView

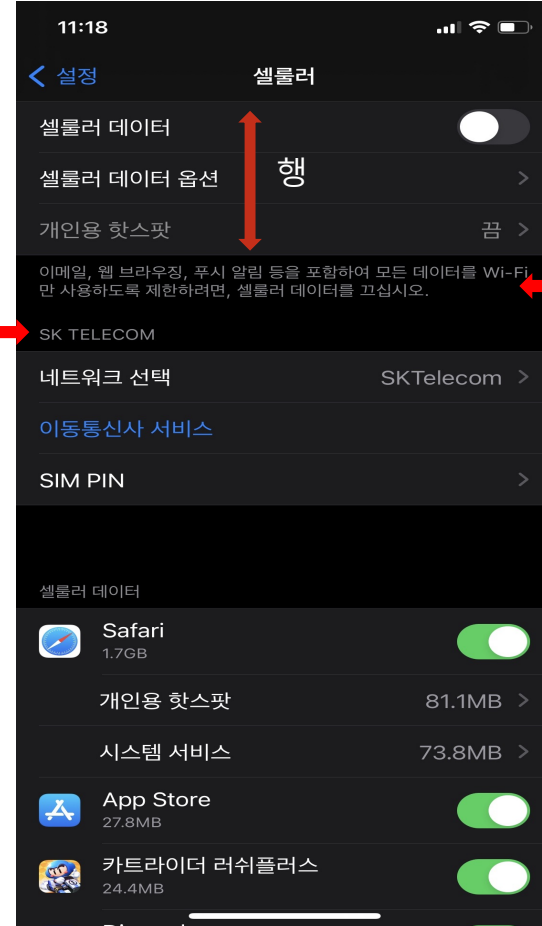
- ▶ 안드로이드의 ListView와 RecyclerView와 동일
- ▶ 정보를 리스트 형태로 보여주기 위해 사용

# 테이블 뷰의 기본 형태



헤더

섹션  
행

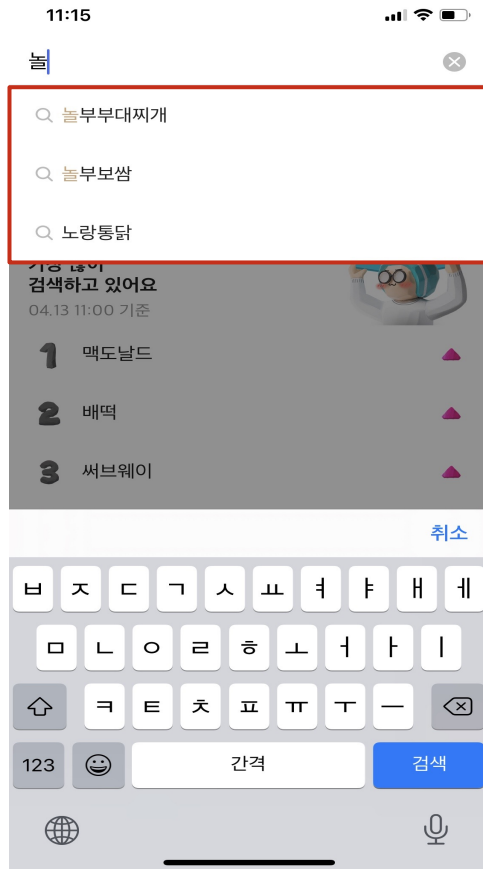


푸터

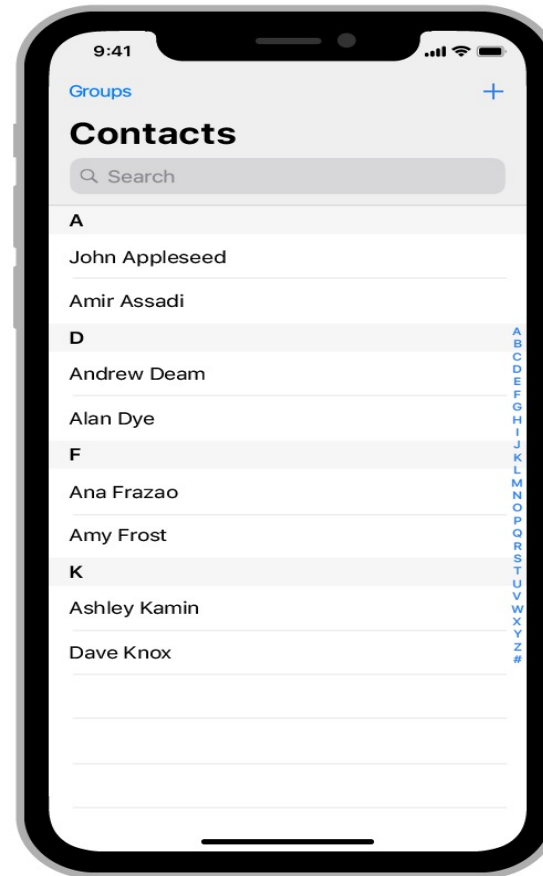
# 테이블 뷰 스타일

	일반	그룹
형태	연속적인 행의 리스트 형태	섹션을 기준으로 그룹화 되어있는 리스트 형태
용도	빠른 탐색을 하거나 옵션을 선택할 때 사용	정보를 특정 기준에 따라 구분할 경우
예시	검색창 <sub>1)</sub>	전화번호부 <sub>2)</sub>

# 테이블 뷰 스타일



1) 일반

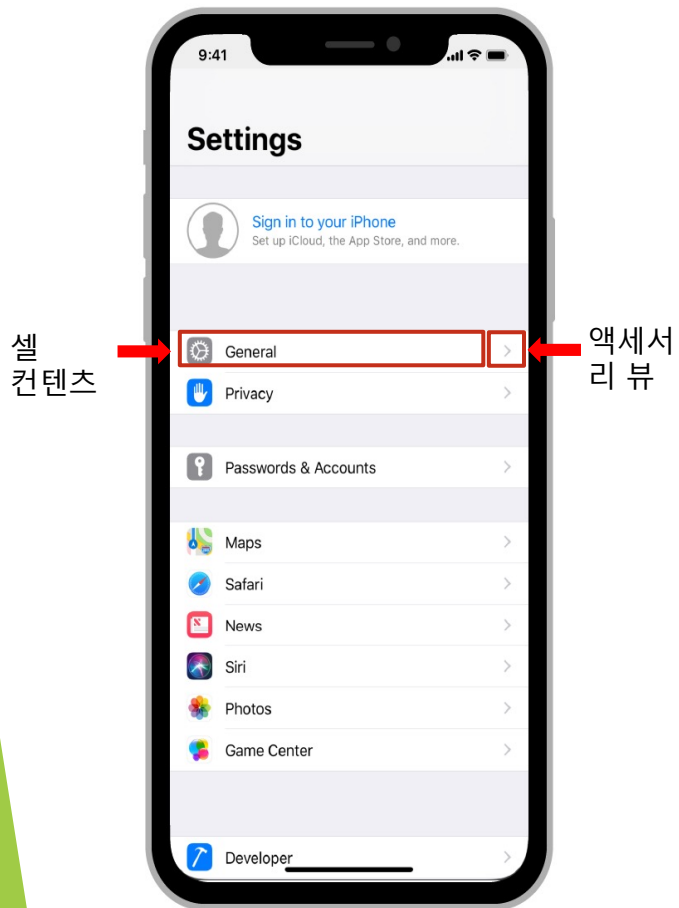


2) 그룹

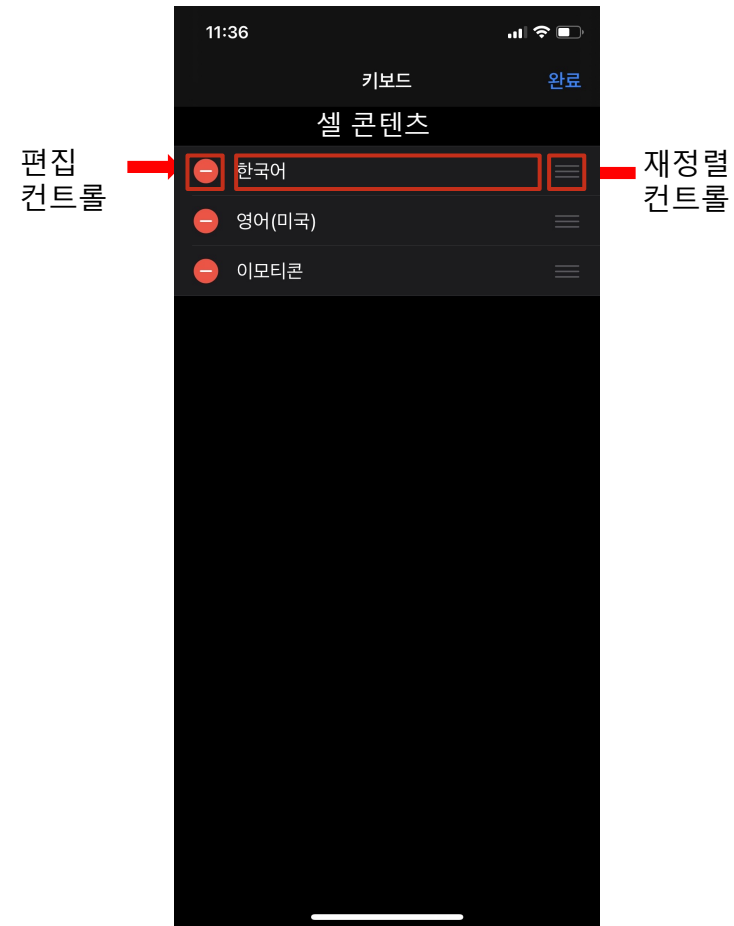
# 테이블 뷰 생성

정적 셀	동적 셀
정해진 형태를 가지고 있는 테이블 뷰에 사용	셀 하나를 디자인하여 템플릿 형식으로 사용할 때

# 테이블 뷰 셀의 구조



기본



편집모드

# UITableViewDataSource와 UITableViewDelegate

# DataSource와 Delegate

- ▶ DataSource

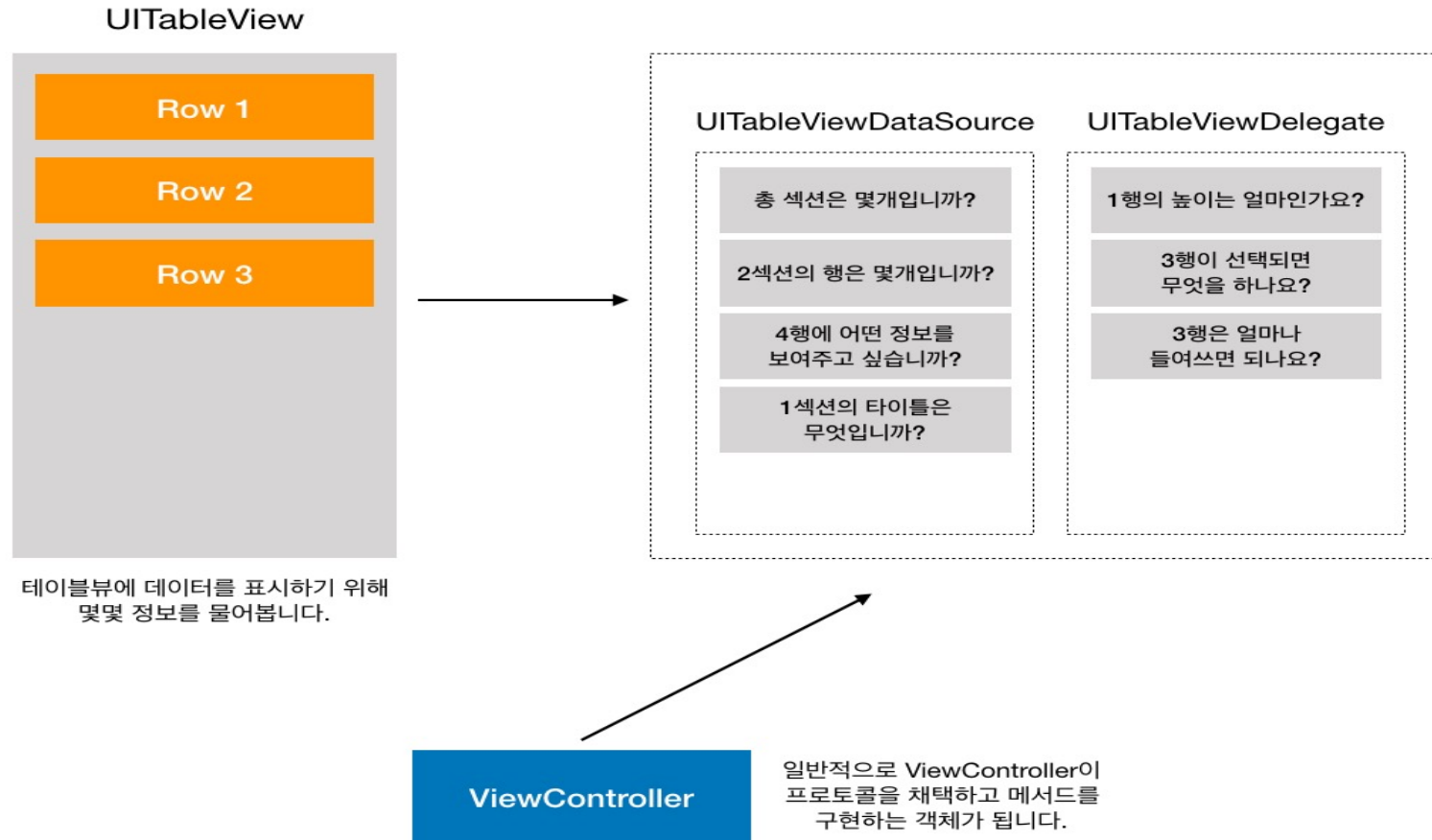
- ▶ 테이블 뷰를 생성하고 수정하는데 필요한 정보를 테이블 뷰에 제공

- ▶ Delegate

- ▶ 테이블 뷰의 시각적인 부분 수정, 행의 선택 관리, 액세서리뷰 지원 그리고 테이블뷰의 개별 행 편집을 도와줌



# DataSource와 Delegate



출처 : 부스트코스

# UIAlertController

# UIAlertController

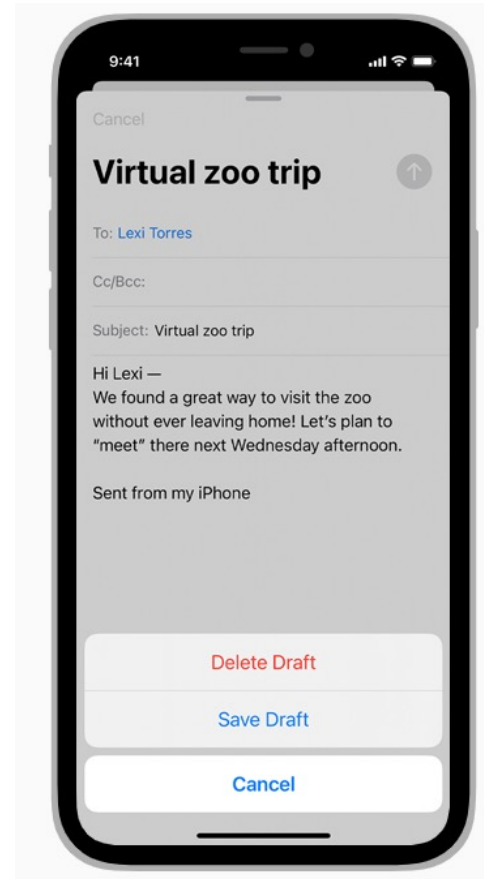
- ▶ 안드로이드의 Alert Dialog와 유사
- ▶ 경고, 확인창을 표시할 때 사용
- ▶ 알림 스타일은 actionSheet, alert 2가지가 있음

# UIAlertController 프로퍼티

title	경고창 제목
message	경고창에 표시할 내용
action	경고 혹은 액션시트에 추가할 액션
preferredStyle	경고창 스타일(alert, actionSheet 중 택 1)

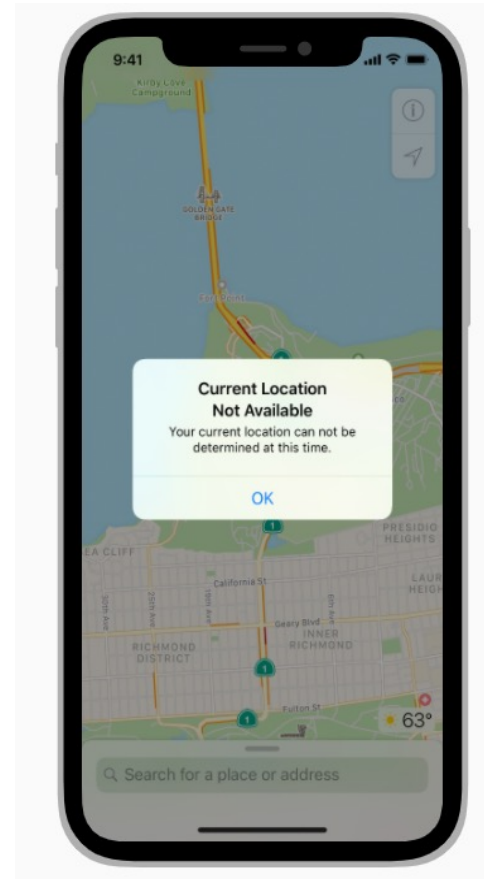
# ActionSheet

- ▶ 두 개 이상의 선택사항을 제공하는 경고 스타일
- ▶ 하단에서 위로 미끄러짐



# Alert

- ▶ 일반적인 경고 스타일
- ▶ 중요한 작업을 하기 전에 표시



# UIAlertAction

- ▶ 사용자가 경고 또는 액션시트에서 사용할 버튼과 버튼을 클릭했을 때 수행할 작업 구성

# UIAlertAction 프로퍼티

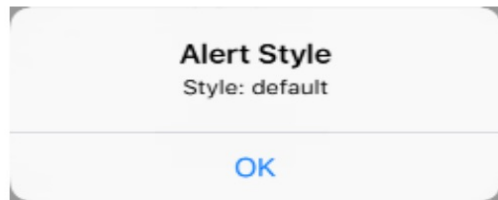
title	액션 제목
style	액션 스타일
isEnabled	액션 사용가능여부



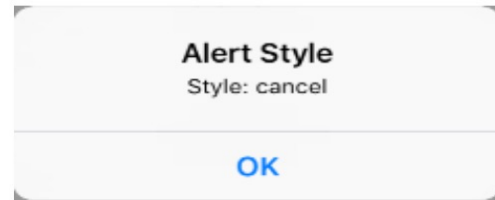
# UIAlertAction 스타일

default	기본 스타일
cancel	작업을 취소하거나 변경사항이 없을 경우
destructive	데이터가 변경되거나 삭제되어 돌이킬 수 없을 경우

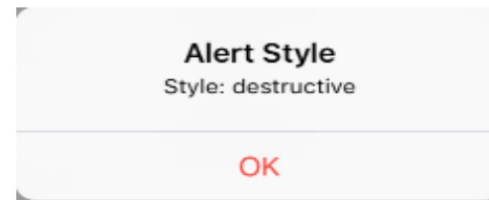
## 얼럿의 액션 스타일 종류



default



cancel



destructive

출처 : 부스트코스

# Alert vs ActionSheet

## Alert

- 중요한 행동 전 경고가 필요할 때
- 행동을 취소할 기회를 제공해야할 때
- 작업을 한번 더 확인하거나 삭제 등의 작업을 수행하거나 문제 사항을 알릴 때
- 결정이 필요한 중요한 정보를 표시할 때

## ActionSheet

- 사용자가 고를 수 있는 액션 목록이 여러 개일 경우
- 되돌리거나 동작이 중요하지 않을 시
- 새 작업 창을 열거나 종료여부 확인 시

The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern, layered effect. The word "Sudo" is centered in a green, sans-serif font.

Sudo

# Sudo

- ▶ 유닉스 혹은 그 계열 운영체제에서 슈퍼 유저로서 프로그램을 구동할 수 있도록 하는 프로그램
- ▶ 윈도우의 cmd를 우측마우스 클릭할 때 나타나는 '관리자 권한으로 실행' 과 같은 기능

The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern and dynamic visual effect. The shapes are layered, with some appearing more prominent than others, and they extend towards the corners of the frame.

# CocoaPod

# CocoaPod

- ▶ 안드로이드의 Gradle, Maven과 유사
- ▶ iOS 라이브러리를 관리하는 툴

# Podfile

- ▶ 해당 앱에 종속되는 라이브러리, 버전 등에 대한 정보를 입력
- ▶ Nodejs의 package.json과 비슷

# CocoaPod 설치 및 프로젝트 연결

- ▶ `sudo gem install cocoapods`
- ▶ `pod init`
- ▶ `pod install`
- ▶ 프로젝트 폴더 내 xcworkspace 확장자를 가진 파일이 추가됨
- ▶ xcworkspace 실행



오토레이아웃

# 오토레이아웃

- ▶ 안드로이드의 Constraint layout과 유사
- ▶ 다양한 디바이스 크기에 대응하기 위해 사용
- ▶ 뷰에 주어진 조건에 따라 동적으로 계산하여 크기 조절하여 내, 외부 변화에 동적으로 반응

# 오토레이아웃의 사용목적

- ▶ 디바이스의 크기가 다양한 경우
- ▶ 디바이스를 회전할 경우
- ▶ 지역화(다국어)를 지원하는 경우
- ▶ 콘텐츠가 동적으로 보여지는 경우
- ▶ 상태표시줄에 녹음 시 나타나는 오디오바와 전화 중에 나타나는 액티브 콜이 보여지거나 사라질 경우

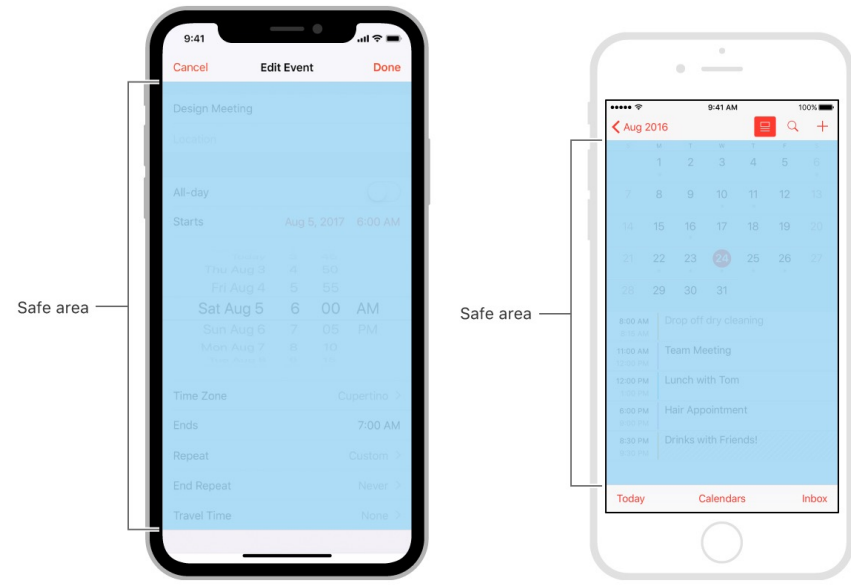
# Constant와 Multiplier

- ▶ Constant : 뷰와 레이아웃의 간격(값)
- ▶ Multiplier : 뷰와 레이아웃간의 비율(%)

# 오토레이아웃 실습

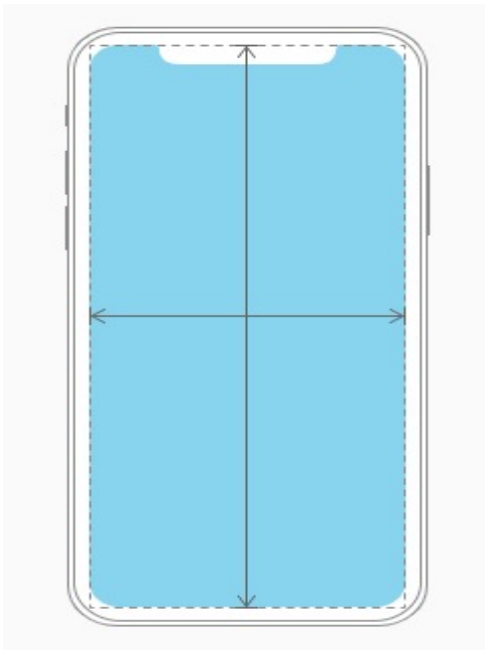
# 안전영역(Safe Area)

- ▶ 새로운 디바이스(노치)가 등장하면서 새로 생긴 개념
- ▶ 콘텐츠가 상태바, 네비게이션바, 툴바, 탭바를 가리는 것을 방지하는 영역



# 기존의 레이아웃 vs 안전영역

- ▶ SafeArea없이 사용할 경우 Superview를 가르키므로 탭바, 네비게이션 바 포함되므로 노치가 적용된 기기에 대응하기 힘들



Superview



SafeArea

# 프로그래밍으로 제어한다면?

- ▶ 스토리보드에서 해당 Constraint 객체를 코드와 연결
- ▶ Constant 혹은 multiplier 값을 변경



# 오토레이아웃 코드 제어 실습

# UIAlertController

# UIAlertController

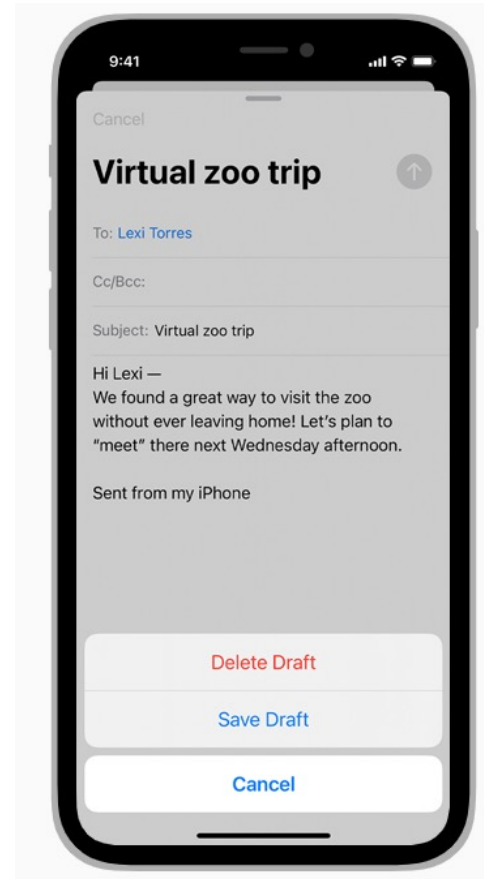
- ▶ 안드로이드의 Alert Dialog와 유사
- ▶ 경고, 확인창을 표시할 때 사용
- ▶ 알림 스타일은 actionSheet, alert 2가지가 있음

# UIAlertController 프로퍼티

title	경고창 제목
message	경고창에 표시할 내용
action	경고 혹은 액션시트에 추가할 액션
preferredStyle	경고창 스타일(alert, actionSheet 중 택 1)

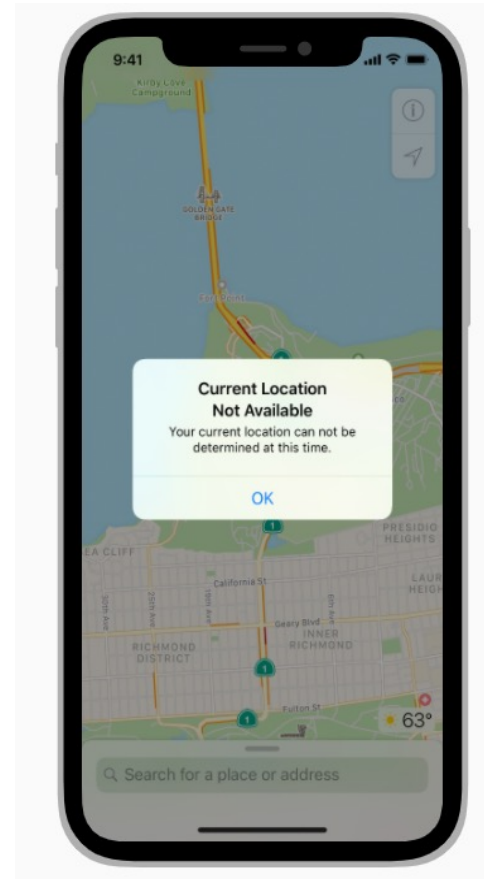
# ActionSheet

- ▶ 두 개 이상의 선택사항을 제공하는 경고 스타일
- ▶ 하단에서 위로 미끄러짐



# Alert

- ▶ 일반적인 경고 스타일
- ▶ 중요한 작업을 하기 전에 표시



# UIAlertAction

- ▶ 사용자가 경고 또는 액션시트에서 사용할 버튼과 버튼을 클릭했을 때 수행할 작업 구성

# UIAlertAction 프로퍼티

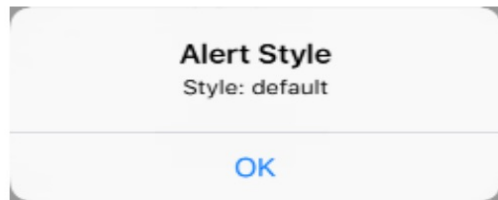
title	액션 제목
style	액션 스타일
isEnabled	액션 사용가능여부



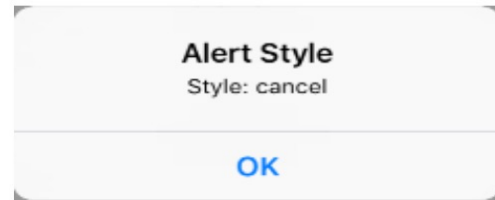
# UIAlertAction 스타일

default	기본 스타일
cancel	작업을 취소하거나 변경사항이 없을 경우
destructive	데이터가 변경되거나 삭제되어 돌이킬 수 없을 경우

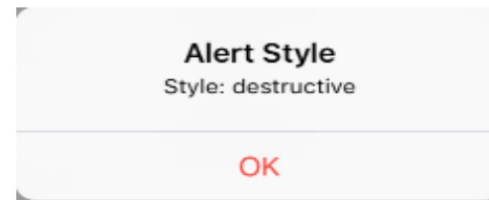
## 얼럿의 액션 스타일 종류



default



cancel



destructive

출처 : 부스트코스

# Alert vs ActionSheet

## Alert

- 중요한 행동 전 경고가 필요할 때
- 행동을 취소할 기회를 제공해야할 때
- 작업을 한번 더 확인하거나 삭제 등의 작업을 수행하거나 문제 사항을 알릴 때
- 결정이 필요한 중요한 정보를 표시할 때

## ActionSheet

- 사용자가 고를 수 있는 액션 목록이 여러 개일 경우
- 되돌리거나 동작이 중요하지 않을 시
- 새 작업 창을 열거나 종료여부 확인 시

The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern, layered effect. The word "Sudo" is centered in a green, sans-serif font.

Sudo

# Sudo

- ▶ 유닉스 혹은 그 계열 운영체제에서 슈퍼 유저로서 프로그램을 구동할 수 있도록 하는 프로그램
- ▶ 윈도우의 cmd를 우측마우스 클릭할 때 나타나는 '관리자 권한으로 실행' 과 같은 기능

The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern and dynamic visual effect. The shapes are layered, with some appearing more prominent than others, and they extend from the edges of the frame towards the center.

# CocoaPod

# CocoaPod

- ▶ 안드로이드의 Gradle, Maven과 유사
- ▶ iOS 라이브러리를 관리하는 툴

# Podfile

- ▶ 해당 앱에 종속되는 라이브러리, 버전 등에 대한 정보를 입력
- ▶ Nodejs의 package.json과 비슷

# CocoaPod 설치 및 프로젝트 연결

- ▶ `sudo gem install cocoapods`
- ▶ `pod init`
- ▶ `pod install`
- ▶ 프로젝트 폴더 내 xcworkspace 확장자를 가진 파일이 추가됨
- ▶ xcworkspace 실행



# Human Interface Guide

# Human Interface Guide

- ▶ 안드로이드의 Material Design과 유사
- ▶ 앱을 개발할 때 필요한 디자인과 동작을 포함한 여러 규칙을 통해 UI를 구성하는 방법에 대한 지침
- ▶ 자사의 제품에서 일관된 사용자 경험을 제공하기 위한 가이드

# HIG를 읽어야 하는 이유

- ▶ 앱 설계 시 사용자와 상호작용에 관해 생각하고 설계할 수 있음
- ▶ 개발자가 아닌 사용자의 입장에서 설계
- ▶ 앱의 일관성 유지
- ▶ 기준점을 제시하여 기획자, 디자이너와 협업 시 효율 증대
- ▶ 사용자에게 익숙한 환경을 제공하여 새로운 앱 사용 시 진입장벽을 낮출 수 있음

# HIG를 지키지 않았을 시 발생하는 문제

- ▶ 사용자에게 좋지 못한 경험을 가져다 줌
- ▶ 앱의 일관성을 해침
- ▶ 앱을 스토어에 등재할 시 승인 거절(Reject) 됨

# iOS 디자인 테마

- ▶ Deference(컨텐츠 존중)

좋은 화면 구성을 통해 사용자들이 콘텐츠를 이해하기 쉽도록 하자

- ▶ Clarity(명확함)

중요한 요소를 강조하자

- ▶ Depth(깊이)

보고 있거나 이동하려는 화면의 깊이감을 느끼도록 하자

# HIG 내용

App Architecture	아키텍처 설계
User Interaction	앱이 사용자와 상호작용하는 동작 설계
System Capabilities	시스템이 지원하는 여러 시스템 기능과 각 기능에 대한 설계
Visual Design	시각적인 요소 설계
Icons and Images	이미지, 아이콘, 시작화면에 관한 내용과 지침

# HIG 인터페이스 필수 요소

Bars	Bar의 역할 및 용도에 관한 설명과 바를 구현할 때
View	뷰의 종류와 역할에 관해 설명하고 뷰 설계 시
Controls	여러 가지 컨트롤 요소들에 관한 소개와 설계 시

The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern, layered effect. The shapes are concentrated on the left and right sides, leaving a large white central area.

# UIButton



# UIButton

- ▶ 안드로이드의 Button와 동일
- ▶ 버튼은 문자, 이미지, 배경이미지로 구성됨

# UIButton의 상태

- ▶ 버튼은 Default, Highlighted, Focused, Selected, Disabled 의 상태를 가지고 있으며 개별적인 설정을 해주어야 한다.
- ▶ 개별적으로 설정하지 않고 Default 상태만 설정할 경우 Xcode가 알아서 적절한 값을 삽입함

# UIButton 실습

The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern, layered effect. The text 'UILabel' is centered in a clean, sans-serif font.

UILabel

# UILabel

- ▶ 안드로이드의 TextView와 같음
- ▶ 글자를 표시할 때 사용

# UILabel 속성

속성	내용
text	표시할 문구
font	텍스트를 표시하는 데 사용되는 글꼴
textColor	텍스트 색상
textAlignment	글자 정렬

# UILabel 실습

# UIStackView



# UIStackview

- ▶ 안드로이드에는 기본으로 존재하지 않는 뷰
- ▶ 오토 레이아웃을 활용하여 디바이스의 사이즈나 모든 변화에 대응 가능한 동적인 UI 구성 가능

# UIStackView 속성

속성	내용
Axis	스택을 수직, 수평 중 설정
distribution	분배방법(X축)

- ▶ fill : 자동으로 뷰 크기 조절
- ▶ fillEqually: 균등하게 크기 유지
- ▶ fillProportionally: 비율에 맞춰 스택뷰 채움
- ▶ equalSpacing: 모든 뷰들이 균등한 간격으로 배열
- ▶ equalCentering: 모든 뷰들이 일정한 Center to Center 간격을 갖도록 배열

참고 : [UIStackView의 Distribution \(velog.io\)](https://velog.io/UIStackView-Distribution)

# UIStackView 실습

The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern, layered effect. The text 'UITextField' is centered in a clean, sans-serif font.

UITextField

# UITextField

- ▶ 안드로이드의 EditText와 동일
- ▶ 글자를 입력할 때 사용

# UITextField 속성

속성	내용
text	표시할 문구
font	텍스트를 표시하는 데 사용되는 글꼴
textColor	텍스트 색상
textAlignment	글자 정렬
placeholder	어떤 형식 또는 값을 입력하려고 알려주는 안내문구 혹은 텍스트 입력 예시 (안드로이드의 hint와 동일)

# UITextField 실습