

과 목 명 : iOS프로그래밍실무

학 과 : 컴퓨터소프트웨어

학 번 : 201822048

     반 : A

이 름 : 황재하

제 출 일 : 2023.04.12

담당교수 : 한성현 교수님

실습.1

// 옵셔널 체이닝이란?

// 옵셔널 값에 접근하기 위해 물음표(?)를 사용하는 문법입니다.

// 만약 접근하려는 속성, 메서드, 혹은 서브스크립트가 옵셔널 값을 반환한다면, 물음표를 이용하여 체이닝을 계속할 수 있습니다.

class Person {

var name: String

var age: Int?

init(name: String) {

self.name = name

}

}

let person: Person? = Person(name: "John")

let age = person?.age

실습. 2

// 옵셔널 변수 선언시 자료형 뒤에 ?나 !추가

var x : String? = "Hi"//Hi지우고도 실습 print(x, x!)

print(x, x!) // 강제 언래핑 ! 사용, x가 nil이면 크래쉬 발생

if let x = x { // 옵셔널 바인딩 : 안전하게 옵셔널을 푸는 방법

print(x)

}

let b = x!.count

print(type(of:b),b)

let b1 = x?.count

print(type(of:b1),b1, b1!)

let c = x ?? ""

print(c)

실습. 3

스위프트의 String에서 많이 사용하는 프로퍼티와 메소드

프로퍼티

count: 문자열의 길이를 반환합니다.

isEmpty: 문자열이 비어있는지 여부를 반환합니다.

startIndex: 문자열의 시작 인덱스를 반환합니다.

endIndex: 문자열의 마지막 인덱스 다음 위치를 나타내는 인덱스를 반환합니다.

unicodeScalars: 문자열의 유니코드 스칼라 값으로 이루어진 콜렉션을 반환합니다.

unicodeScalars.count: 문자열에서 유니코드 스칼라 값의 수를 반환합니다.

unicodeScalars.indices: 문자열의 유니코드 스칼라 값의 인덱스를 포함하는 범위를 반환합니다.

메서드

contains(\_:): 문자열이 특정 문자열을 포함하고 있는지 여부를 반환합니다.

hasPrefix(\_:): 문자열이 특정 문자열로 시작하는지 여부를 반환합니다.

hasSuffix(\_:): 문자열이 특정 문자열로 끝나는지 여부를 반환합니다.

lowercased(): 문자열의 모든 문자를 소문자로 변환한 새로운 문자열을 반환합니다.

uppercased(): 문자열의 모든 문자를 대문자로 변환한 새로운 문자열을 반환합니다.

replacingOccurrences(of:with:): 문자열에서 특정 문자열을 다른 문자열로 바꾼 새로운 문자열을 반환합니다.

split(separator:maxSplits:omittingEmptySubsequences:): 문자열을 특정 구분자를 기준으로 나누어 배열로 반환합니다.

trimmingCharacters(in:): 문자열의 특정 문자나 문자 집합을 제거한 새로운 문자열을 반환합니다.

components(separatedBy:): 문자열을 특정 구분자를 기준으로 나눈 배열을 반환합니다.

실습. 4

r x: Int = 10

class Person {

var name: String

var age: Int

init(name: String, age: Int) {

self.name = name

self.age = age

}

}

let kim: Person = Person(name: "Kim", age: 20)

print(kim.age) // 20

let han: Person? = Person(name: "Han", age: 25)

// 강제 언래핑으로 접근, 크래쉬 위험

print(han!.age) // 25

// 옵셔널 체이닝으로 접근, 크래쉬 위험 없음 하지만 옵셔널 형으로 반환함

print(han?.age) // Optional(25)

// 일반형으로 반환하는 방법 1 : 옵셔널 체이닝 한 것을 한 번더 강제 언래핑

print((han?.age)!) // 25

// 일반형으로 반환하는 방법 2 : 옵셔널 바인딩 if let

if let hanAge = han?.age {

print(hanAge) // 25

} else {

print("nil")

}

실습.5

  


