<스위프트 기초>

**:: 명명법 / 콘솔로그 / 문자열 보간법 ::**

**1. 명명법**

* **Lower Camel Case (소문자로 시작)** : fuction, method, variable, constant

ex.) someVariableName

* **Upper Camel Case** (대문자로 시작): type(class, struct, enum, extension…)

ex.) Person, Point, Week

\* Swift는 모든 대소문자를 구분합니다.

**2. 콘솔로그 남기기**

* **print 함수** : 단순 문자열 출력
* **dump 함수** : 인스턴스의 자세한 설명(description 프로퍼티)까지 출력

**3. 문자열 보간법 (String Interpolation)**

* 프로그램 실행 중 문자열 내에 변수 또는 상수의 실질적인 값을 표현하기 위해 사용.
* \()

ex) \(변수명) 과 같이 사용하면 된다.

:: 상수와 변수 ::

**1. 상수와 변수 선언**

* **let**:상수 선언 키워드
* **var**:변수 선언 키워드

//값의 타입이 명확하다면 타입은 생략 가능

let 이름 = 값

var 이름 = 값

let constant: String = "차후에 변경이 불가능한 상수 let"

var variable: String = "차후에 변경이 가능한 변수 var"

variable = "변수는 이렇게 차후에 다른 값을 할당할 수 있지만"

// constant = "상수는 차후에 값을 변경할 수 없습니다" // 오류발생

**2. 상수 선언 후, 값 할당하기**

* 선언을 한 뒤, 나중에 값을 할당하려는 상수나 변수는 반드시 타입을 명시해야 합니다.

//나중에 할당하려고 하는 상수나 변수는 타입을 꼭 명시해주어야 합니다

let sum: Int

let inputA: Int = 100

let inputB: Int = 200

// 선언 후 첫 할당

sum = inputA + inputB

// sum = 1 // 그 이후에는 다시 값을 바꿀 수 없습니다, 오류발생

// 변수도 물론 차후에 할당하는 것이 가능합니다

var nickName: String

nickName = "yagom"

// 변수는 차후에 다시 다른 값을 할당해도 문제가 없지요

nickName = "야곰"

**3. 생각해보기**

다음과 같은 경우에 각 값은 상수와 변수 중 어느 것으로 선언하는 것이 더 좋을지 생각해 봅시다

OOO name = "yagom"

OOO numberToAdd = 5

OOO pi = 3.14195

OOO maxItemCount = 1000