

ios 입문 강의

3강. 연산자와 조건문

1. 지난 수업 개념 복습

(1) 변수

* 변수에 대하여

- * 변수란, 데이터를 담는 박스

* 변수 - Type

- * 변수 내용물의 형태
 - * 변수 타입에 따라 변수에 저장할 수 있는 값의 종류와 범위가 달라진다.

* 변수 - Name

- * 조금 전에 저장한 데이터는 어느 상자에 있는가?
- * 변수명을 지정 -> 컴퓨터와 사용자간의 데이터 사용을 위한 네이밍

* 변수 만드는데 필요한 것

* 변수 명 (Name) + 변수 타입 (Type)

- * 변수의 타입에 따라 메모리 공간 지정

* 변수의 키워드를 쓰는 이유

- * 스위프트는 컴파일 언어
 - * 컴파일 언어는 컴파일러가 프로그래밍 언어를 기계어로 번역
 - * -> 컴파일러에게 예약어(키워드)를 통해 어떤 변수를 만들 것인지, 함수를 만들 것인지 알려주는 것
 - * 띄어쓰기 자체도 문법의 일환으로 적용됨. 띄어쓰기도 정석대로 하는 것이 코드가 깔끔해진다.

* 변수와 상수의 활용 - 메모리

변수	상수
저장하는 값이 변할 수 있는 메모리 공간	변수와 달리 메모리 공간을 할당하지 않고 바로 사용
실행 중에 데이터가 변할 수 있다.	사용한 값이 메모리에 올라가지만 그 값을 바꿀 수 없다.
** 상수값은 프로그램 실행 중에 변경 불가능하게 만들었기 변할 일이 없다. => 메모리 절약	

* 프로그래밍에서 가장 중요한 것

* 변수(Variable)

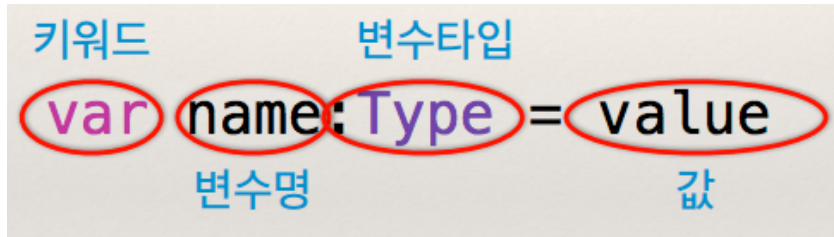
- * 데이터가공
- * 저장소 개념
- * 메모리에 저장됨

* 함수(Method)

- * 행동지시
- * 액션을 지칭
- * CPU가 연산 후, 변수에 집어넣는다 (메모리에 집어 넣는다)
 - * 일정한 행동을 끝내면 변수에 집어넣고
 - * 행동이 끝일 때에는 CPU에서 액션이 끝난다.

(2) Swift 문법과 함수

* Swift 문법 - 변수



* 변수 타입

`var name : Type = value` 변수

`let name : Type = value` 상수

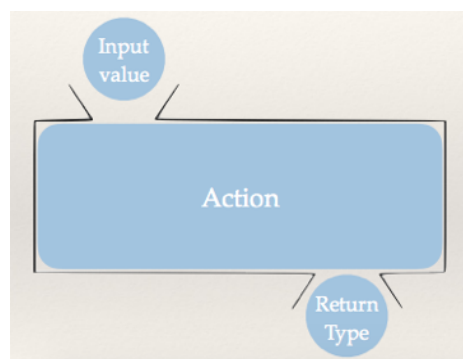
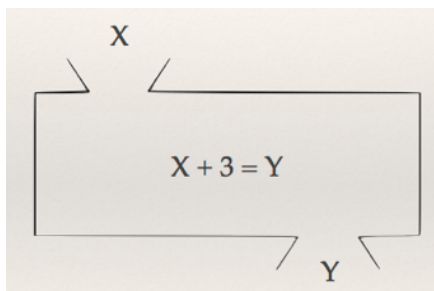
타입이름	타입	설명	Swift 문법 예제
정수	Int	1, 2, 3, 10, 100	<code>var intName : Int</code>
실수	Double	1.1, 2.35, 3.2	<code>var doubleName : Double</code>
문자열	String	"this is string"	<code>var stringName : String</code>
불리언	Bool	true or false	<code>var boolName : Bool</code>

* 값 지정하기

A diagram showing the Swift variable assignment syntax: `var number: Int = 3`. An arrow points from the value `3` to the variable `number`, indicating that the value is assigned to the variable.

* 우측에 있는 값을 좌측 변수에 대입

* 함수에 대하여



* 함수란, 입력을 통해 어떠한 일을 수행한 다음 결과물을 내어놓는 것

* 함수를 만들기 위해 필요한 것

* 함수명(Name) + 입력값(Input Value) + 함수내용(Action) + 결과타입(Return Type)

- * input값에 따라 output 값이 정해짐
- * 결과물에는 Type 을 정해줘야한다.

* Swift 문법 - Input 값

• No Input값 예제

```
func getAge() -> Int {  
    var age:Int = 22  
    return age  
}
```

• 다중 input값 예제

```
func sumNumber(num1:Int, num2:Int) -> Int {  
    var returnNum:Int  
    returnNum = num1 + num2  
    return returnNum  
}
```

* No Input값 예제

- * input이 없으면 () 빈 괄호

* 다중 Input값의 예제

- * input이 여러 개 이면 콤마(,)로 구분

(3) 함수 쉽게 만들기!

1. Input이 무엇일지 고민하기

* Input 변수가 필요한가? 함수 외부 변수를 사용할 것인가?

2. output이 무엇인지 고민하기

* Return Type이 필요한가? 외부변수에 데이터를 저장하면 되는가?

3. Input, output 고려해서 함수 내용 만들기

* 함수 내용을 만들며 필요하면 내부 변수와 외부 변수 만들기

* 내부 변수 (지역변수) - 해당 함수 내부 { } 안에서 사용 / 사용 후에는 함수가 마지막에 데이터를 지움

* 외부 변수 (전역변수) - 함수 내부 뿐 아니라 다른 함수, 외부에서도 사용 가능

클래스 안의 변수는 클래스 전체 어떤 함수를 써도 사용 가능

*클래스의 구조를 파악하라!

```
class classname
{
    var 변수1:String = "1"
    var 변수2:Int = 4

    func 함수1
    {
        var 내부변수:Double
        if(조건)
        {
            조건이 참일때 실행
        }
    }

    func 함수2
    {
        for name in list
        {
            반복문 내용 실행
        }
    }
}
```

😬 모든 행동(변수에 값 지정, 함수 호출, 객체 생성 등) 은

함수 안에서 이뤄져야 한다.

*함수 밖에서 불러진 행동은 컴파일 에러에 속한다.

* 들여쓰기에 따라 함수와 변수는 클래스 안에 들어가야 한다.

* 클래스 밖에서 함수와 변수를 만들어도 되긴 되나,

현재에는 사용하지 않을 것

(4) 간단한 문제 풀어보기

EX 1) 두 수를 더해서 결과값을 돌려주는 함수

1. Input Value
2. Output Type
3. function Name : addNum
4. 함수 내용

EX 2) 시험점수를 더하는 함수

1. Input Value
2. Output Type
3. function Name : addNum
4. 함수 내용

정답

1. 두 수를 더해서 결과값을 돌려주는 함수

1. Input Value : 정수형 변수 2개(num1, num2)
2. Output Type : Int
3. function Name: addNum
4. 함수 내용은?
(num1 + num2)

```
func addNum(num1:Int, num2:Int) ->Int  
{  
    return num1+num2  
}
```

2. 시험 점수를 더하는 함수

1. Input Value : 시험점수 1개 (score:Int)
2. Output Type : 없음(외부 변수에 데이터 저장)
3. function Name: addExamScore
4. 함수 내용은?
외부 변수(totalScore)에
score값 추가

```
func addExamScore(score:Int)  
{  
    totalScore = totalScore + score  
}
```

실습 1

Q 더하기 함수와 평균을 구하는 함수 만들기

1. 토탈값 변수 만들기 : 초기값 0
2. 카운트 변수 만들기 : 초기값 0
3. 더하기함수만들기 : 숫자를 하나씩 토탈 값에 저장하는 함수
ps. 더하기함수 내용 중 카운트의 숫자를 1 올려 주는 코드작성
4. 평균 구하는 함수 만들기 : 입력 값은 없으며 토탈을 카운트로 나눠서 리턴
5. 실행 하기 : viewDidLoad 함수 안에서 실행.
더하기 함수를 몇 번 실행 후 마지막에 지금까지 더한 숫자의 평균을 구하기

----- 코드 예시 -----

```
func plusFunction(num:Int) {  
    // total에 num으로 들어오는 수를 계속 더한다.  
    // total = total + num  
    total = total + num  
    count += 1  
}  
  
func averageFunction() -> Double {  
    //총 total값의 평균을 구하는 것  
    print(Double(total) / Double(count))  
    return Double(total) / Double(count)  
}
```

----- viewDidLoad ----- (함수 호출하는 곳)

```
override func viewDidLoad() {  
    super.viewDidLoad()  
  
    plusFunction(1) // total = 1 / count = 1  
    plusFunction(2) // total = 4 / count = 2  
    plusFunction(3) // total = 8 / count = 3  
  
    averageFunction()  
}
```

*string 조합

- string 병합

: string을 + 기호로 이을 수 있다.

```
var name:String  
name = "주" + "영민"
```

- interpolation (삽입) : \참조값

```
var name:String = "주영민"  
print("my name is \ (name) ")  
    \ ( )가 interpolation
```


2. 연산자

1) 산술 연산자

기호	예제	설명
+	$1 + 2 = 3$	더하기
-	$2 - 1 = 1$	빼기
*	$2 * 1 = 2$	곱하기
/	$10 / 3 = 3$	나누기
%	$10 \% 3 = 1$	나머지

2) 비교 연산자

기호	예제	설명
==	$A == B$	A와 B가 같다.
>=	$A >= B$	A가 B보다 크거나 같다
<=	$A <= B$	A가 B보다 작거나 같다.
>	$A > B$	A가 B보다 크다
<	$A < B$	A가 B보다 작다

* $>=$: 크거나 작다 🖐️ $=>$ 이렇게 순서 바뀌면 에러

3) 논리 연산자

기호	예제	집합	설명
&&	A조건 && B조건		A조건이 참이고, B조건이 참이면 참이다.
	A조건 B조건		A조건이나, B조건 둘중에 하나가 참이면 참이다.
!	!(A조건 B조건)		A B조건 of 반대

- * 논리의 타입 = Bool → true / False
- * 참 거짓으로 나타낼 것들의 조합
- * && : and → a, b 둘다 참일때 (교집합)
- * || : or → a이거나 b (합집합)
- * ! : not ⇒ a와 b의 반대 (여집합)

4) 추가 연산자

증감연산자	예제	설명
++	a++	1씩 추가
--	b--	1씩 빼기
복합연산자	예제	설명
+=	a += 1	a에 값을 더하기
-=	b -= 2	b에 값을 빼기
범위 연산자	예제	설명
a...b	3...10	a~b까지의 숫자
a..<b	0..<10	a~b까지 숫자중 b는 포함 안함

- * 증감 연산자 : ++ / -- (1씩 추가 / 빼기)
- * 복합 연산자 : += / -= (~값을 추가 / 빼기)
- * 범위 연산자 : a ...b (a부터~b까지) / a..<b (a부터 b까지 중 b는 포함 안함)

3. 조건문 (if- else문)

(1) 조건문 이란?

- * 일정한 조건이 충족될 때 선택적으로 실행
- * 종류 : if-else / switch문

(2) if - else문 - 조건 비교

```
if(조건) {  
    //조건이 만족되면 실행  
}else{  
    //조건이 만족되지 않을때 실행  
}
```

*조건값은 꼭 참거짓으로 나타나는 불리언으로 나타나야된다.

(3) 조건의 예시 → 조건문을 만드는 예들을 연습 해볼 것

- 문제 -

변수 name의 값이 "주영민"인가?

변수 age의 값이 31인가?

변수 height의 값이 176보다 큰가?

변수 weight의 값이 80보다 작은가?

- 답안 -

```
변수 name의 값이 "주영민"인가? (name == "주영민")  
변수 age의 값이 31인가? (age == 31)  
변수 height의 값이 176보다 큰가? (height > 176)  
변수 weight의 값이 80보다 작은가? (weight < 80)
```

* if - else 문과 조건 적용

```
var name:String = "주영민"  
  
if (name == "주영민")  
{  
    print("본인입니다.")  
}else  
{  
    print("본인이 아닙니다")  
}
```

실습 2 홀짝을 판별하는 함수 만들기

-> 내용을 조건문과 비교 연산자를 이용하여 만드시오.

1. Input Value
2. Output Type
3. function Name : addNum
4. 함수 내용

———— 실습 2 답안 예제 ————

1. Input Value : 정수
2. Output Type : Bool
3. function Name: isEvenNumber
4. 함수 내용은
입력받은 숫자를 2로 나눈 나머지가
0이면 짝수 0이 아니면 홀수

```
func isEvenNumber(num:Int) -> Bool
{
    if(num%2 == 0)
    {
        //짝수
        return 1
    }else
    {
        //홀수
        return 0
    }
}
```

(4) 조건의 조합

조건에 대한 여러가지 조합이 있을 경우의 연산자 사용을 연습해보자.

1. 이름이 “김뽀뽀”이고 나이가 31살인가?

* 주어진 이름과 나이가 동시에 만족

(name == “김뽀뽀” && age == 31)

2. 키가 170 이상이거나 몸무게가 60이상 인가?

* 주어진 키와 몸무게가 하나만 포함되어도 만족

(height >= 170 || weight >= 60)

* 조건의 조합을 이용한 예시

```
let height:Int = 177
let weight:Int = 63
if(height >= 170 && weight >= 60)
{
    print("소개팅 가능")
}else
{
    print("소개팅 불가")
}
```

(4) 추가 조건 else-if

```
let height:Int = 177
let weight:Int = 50
if(height >= 170 && weight >= 60)
{
    print("소개팅 가능")
}else if(height >= 170 || weight >= 60)
{
    print("소개팅 가능하나 물어봐야함")
}else
{
    print("소개팅 불가")
}
```

해당 코드 실행 진입 →

-> if 가 아니면, else if 적용 되는지 확인

-> else if 도 안되면 else

실습 3

성적 평균 입력시 해당 평균의 성적 등급을 매긴다

결과 : 100~90: 1등급 / 89~80:2등급 ...

1. Input Value : Double
2. Output Type : int
3. function Name : matchingGrade
4. 함수 내용

실습 예제

```
func matchingGrade(score:Double) -> Int {  
    if (100<=score && score<90)  
    {  
        return 1  
    }  
    else if (80<=score && score<70)  
    {  
        return 2  
    }  
    else  
    {  
        return 3  
    }  
}
```

** else if 는 많아질 수 있습니다!

실습 4

돈을 받아서 돈을 동전으로 거슬러주는 함수

결과: “거스름돈은 500원짜리 x개, 100원짜리 y개, 50원짜리 z개, 10원까지 a개 입니다.”

1. Input Value
2. Output Type
3. function Name : addNum
4. 함수 내용

실습 예제

```
func changeCoin(money:Int) {  
  
    var x:Int = 0  
    var y:Int = 0  
    var z:Int = 0  
    var a:Int = 0  
  
    // 계산 후에 남은 돈 저장할 변수  
    var change = 0  
  
    if (money / 500 > 0)  
    {  
        x = money / 500  
        change = money % 500  
    }  
    if (change / 100 > 0)  
    {  
        y = change / 100  
        change = change%100  
    }  
    if (change / 50 > 0)  
    {  
        z = change / 50  
        change = change%50  
    }  
    if (change / 10 > 0)  
    {  
        a = change / 10  
        change = change%10  
    }  
  
    print("거스름돈은 500원짜리 \(x)개, 100원짜리 \(y)개, 50원짜리 \(z)개, 10원까지 \(a)개 입니다.")  
}
```