

# Tool Box

---

- 각 치수 전환 함수 만들기

1. inch to cm, cm to inch (2개)
2. m2 to 평, 평 to m2 (2개)
3. 화씨 to 섭씨, 섭씨 to 화씨 (2개)
4. 데이터량 (KB to MB, MB to GB) (2개)
5. 시간(hhmmss)을 받으면 초단위로 변경, 초단위를 시간으로 변경  
ex) 11320 >> 4400초

---

# 계산기 만들기

---

# 평균 구하기

---

- 국어, 수학, 과학, 영어, 사회 과목 총합 구하기
- 점수 평균 구하기

# 학점 구하기

---

- 컴퓨터기초, 알고리즘, 자료구조, 운영체제의 학점 구하기
- 전체 학기의 학점 구하기

“과목이 더 늘어 난다면 어떻게 해야될까요?”

학기별 평균값을 알고싶다면요?

“나 뿐이 아닌 친구의 평균도 구하려면  
어떻게 해야될까요?”

# 프로그래밍 패러다임

---

- 절차 지향형 프로그래밍
- 객체지향형 프로그래밍
- 함수형 프로그래밍
- 논리형 프로그래밍
- 등등...

# Swift는

---

- 객체지향형, 함수형, 프로토콜 지향 프로그래밍을 모두 다룰수 있는 언어이다.
- iOS프레임 워크는 객체 지향형 프로그래밍으로 구현되어 있어 객체 지향형으로 프로그래밍으로 하도록 권장한다.

# 객체지향형 프로그래밍 방법론

---

객체 지향 프로그래밍은 컴퓨터 프로그램을 명령어의 목록으로 보는 시각에서 벗어나 여러 개의 독립된 단위, 즉 "객체"들의 모임으로 파악하고자 하는 것이다. 각각의 객체는 메시지를 주고받고, 데이터를 처리할 수 있다.



# 게임을 만들어 봅시다.

---

- 만약 내가 게임을 만든다고 한다면 어떻게 해야할까요?

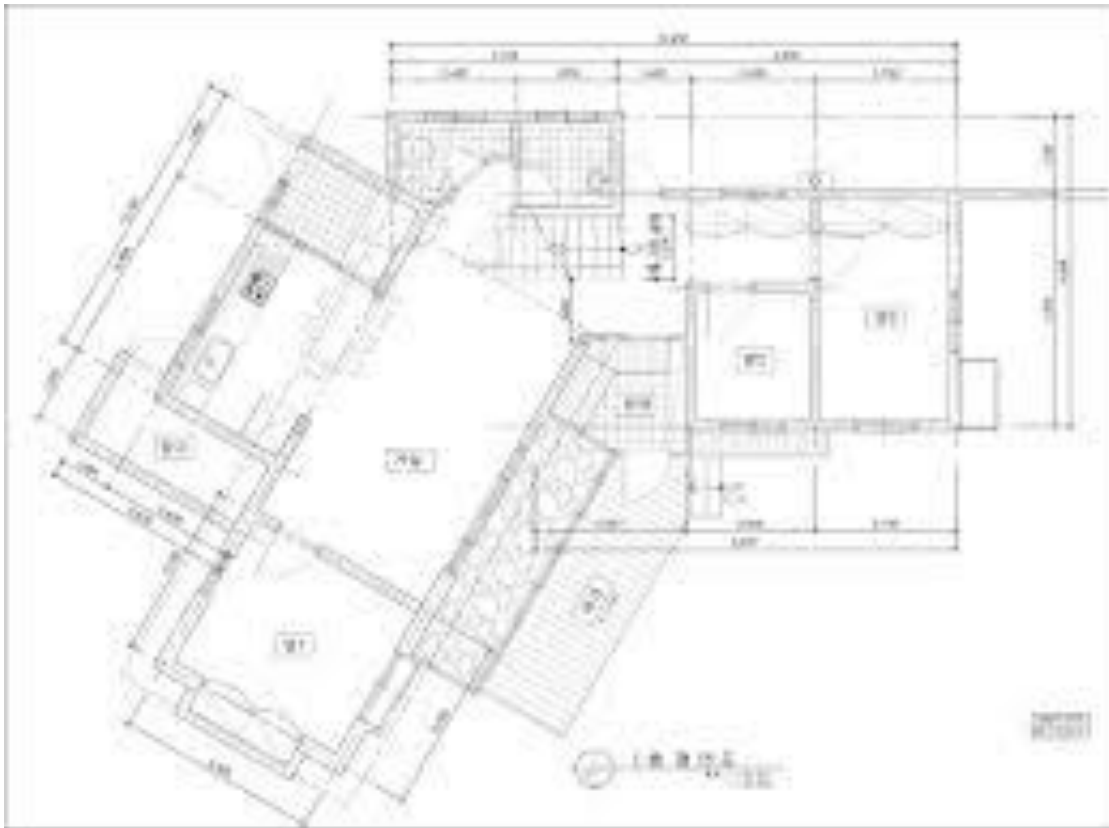
# 기본 구성 요소

---

- **클래스(Class)** - 같은 종류(또는 문제 해결을 위한)의 집단에 속하는 속성(attribute)과 행위(behavior)를 정의한 것으로 객체지향 프로그램의 기본적인 사용자 정의 데이터형(user define data type)이라고 할 수 있다. 클래스는 프로그래머가 아니지만 해결해야 할 문제가 속하는 영역에 종사하는 사람이라면 사용할 수 있고, 다른 클래스 또는 외부 요소와 독립적으로 디자인하여야 한다.
- **객체(Object)** - 클래스의 인스턴스(실제로 메모리상에 할당된 것)이다. 객체는 자신 고유의 속성(attribute)을 가지며 클래스에서 정의한 행위(behavior)를 수행할 수 있다. 객체의 행위는 클래스에 정의된 행위에 대한 정의를 공유함으로써 메모리를 경제적으로 사용한다.
- **메서드(Method), 메시지(Message)** - 클래스로부터 생성된 객체를 사용하는 방법으로서 객체에 명령을 내리는 메시지라 할 수 있다. 메서드는 한 객체의 서브루틴(subroutine) 형태로 객체의 속성을 조작하는 데 사용된다. 또 객체 간의 통신은 메시지를 통해 이루어진다.

# 클래스 vs 객체

---



# 클래스 vs 객체

---



# 클래스 vs 객체

---

- 여러분은 어떤 차이를 느끼시나요?

# 객체지향형 프로그래밍 방법론

---

객체 지향 프로그래밍은 컴퓨터 프로그램을 명령어의 목록으로 보는 시각에서 벗어나 여러 개의 독립된 단위, 즉 "객체"들의 모임으로 파악하고자 하는 것이다. 각각의 객체는 메시지를 주고받고, 데이터를 처리할 수 있다.

# 객체지향형 프로그래밍 특징

---

- 추상화
- 캡슐화
- 은닉화
- 상속성
- 다형성

# Swift Class Architecture

```
class ClassName : superClass
{
    var vName1 = "1"
    var vName2 = 4

    func fName1() - > Any
    {

    }

    func fName2(_ ani:Bool)
    {

    }
}
```

<CalssName.swift>



# 변수 & 함수

---

- 변수 : 프로그램에서 데이터의 저장공간을 담당
- 함수 : 프로그램이 실행되는 행동을 담당

# 정리

---

- 객체지향 프로그래밍은..  
변수와, 함수로 이뤄진 클래스파일을 기초로 하며  
이 클래스 파일을 메모리에 적재되면 객체화가 되고  
객체들의 속성과 명령어를 수행 하면서 프로그램이 실행된다.

# Project 1. 계산기 만들기

---

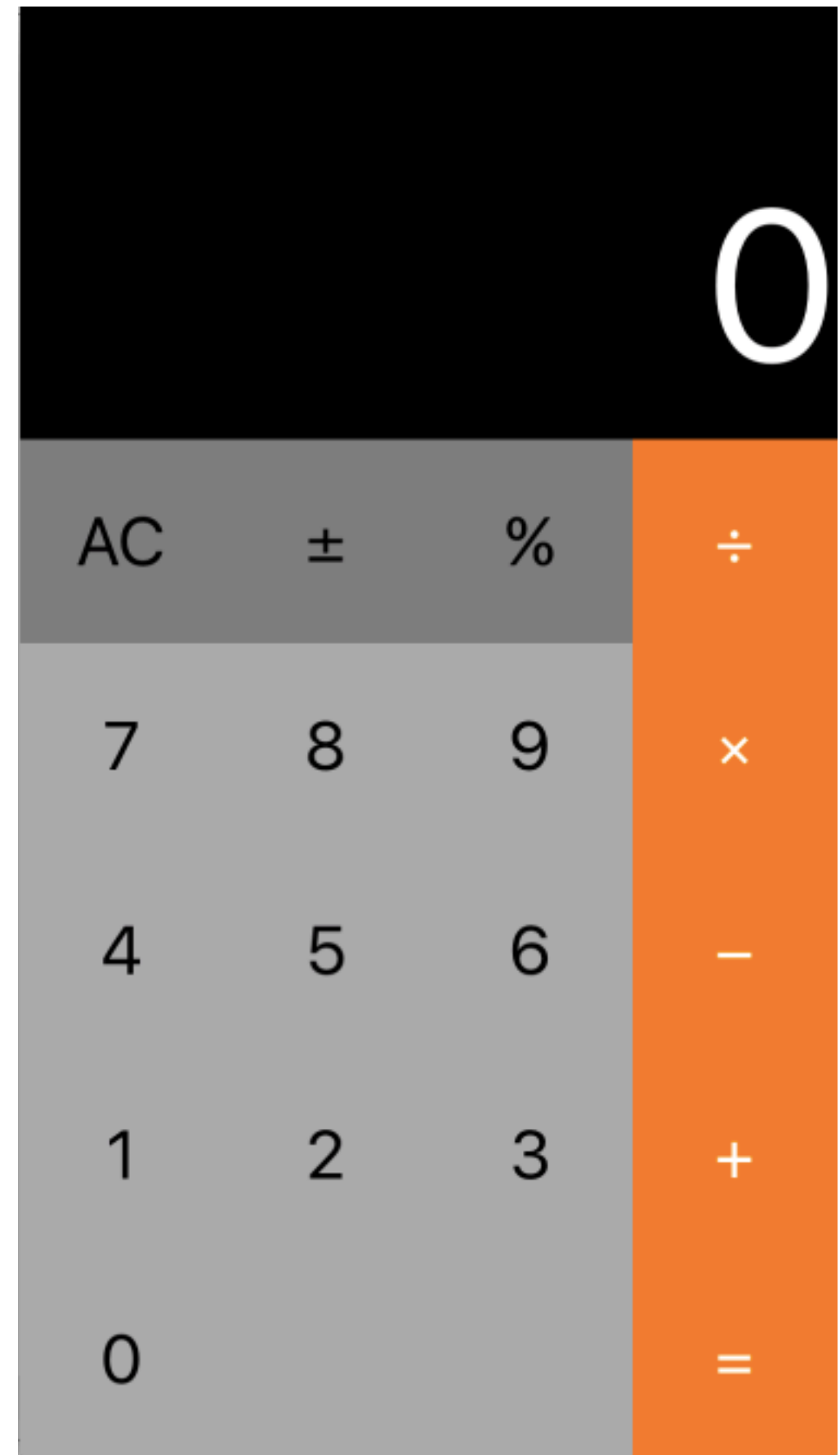
Step 1. UI만들기

Step 2. UI - Class 연결

Step 3. 버튼 액션 만들기

Step 4. 계산기 완성

# Step 1. Make UI



# 속성 UI만들기

---

!!!!같이 해보아요~~