

모바일 소프트웨어 설계

과제01

SimpleCalculator

분반: 00반

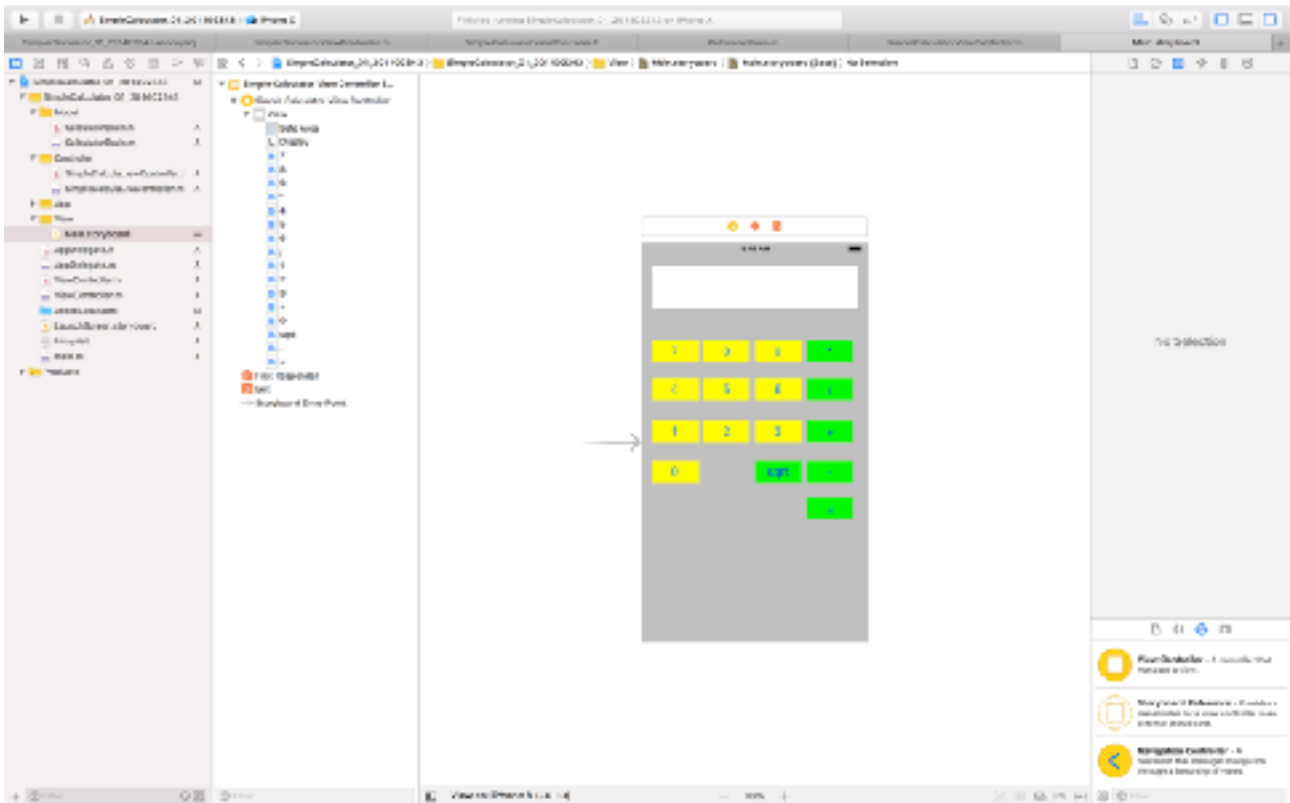
학번 : 201402343

이름 : 남진우

• 문제 정의

- Objective C 언어를 사용하여 SimpleCalculator를 구현한다.
- Button과 Outlet을 이해하고 Action을 추가하고 구현한다.

• 과제 설명



- Mainstoryboard 화면이다. 각 digit 과 operator의 버튼들과 그 화면을 보여주는 Label 로 이루어져 있다.

```
9  #import <UIKit/UIKit.h>
10
11 @interface SimpleCalculatorViewController : UIViewController
12
13     @property (weak, nonatomic) IBOutlet UILabel *display;
14     - (IBAction)digitPressed:(UIButton *)sender;
15     - (IBAction)operatorPressed:(UIButton *)sender;
16
17 @end
18
```

- SimpleCalculator.h 코드 캡처 화면이다. Main.storyboard 에서 digit들은 digitPressed Action 으로 가져오고 operator들은 operatorPressed Action으로 가져왔다. 각 버튼들이 눌릴때 같은 함수를 실행 할 수 있도록 여러가지들을 한번에 설정을 하였다.

```

◎ @interface SimpleCalculatorViewController ()
◎ @property (nonatomic) CalculatorBrain* brain;
◎ @property (nonatomic) BOOL userIsInTheMiddleOfTypingANumber;
15
16 @end

```

- SimpleCalculator.m 에 선언된 변수들이다. 이것들을 이용해서 숫자를 치고있는지와 중추적인 역할을 하게된다.

```

39 -(CalculatorBrain*)brain{ //Override 시켜준 것.
40     if(_brain == nil){
41         _brain = [[CalculatorBrain alloc] init];
42     }
43     return _brain;
44 }
45

```

- Override를 시켜준 부분이다. 그 이유는 brain이 nil일 때 새롭게 할당을 해주기 위해서이다.

```

◎ - (IBAction)digitPressed:(UIButton *)sender {
47     NSString *digit = sender.titleLabel.text;
48     if(self.userIsInTheMiddleOfTypingANumber){
49         self.display.text = [self.display.text stringByAppendingString:digit];
50     }
51     else{
52         self.display.text = digit;
53         self.userIsInTheMiddleOfTypingANumber = YES;
54     }
55 }
--

```

- digitPressed Action 함수이다. 이것은 숫자가 눌렸을 때 작동되는 함수이다. 현재 라벨에 있는 텍스트를 새롭게 변수로 저장해준뒤 조건문 안으로 들어간다. 만약에 숫자를 Typing하고 있었다면 그 뒤에 Append 를 시켜주고, 그렇지 않다면 현재 digit을 텍스트로 설정 한 후에 Typing boolean을 YES 로 바꾸어 준다.

```

68 - (IBAction)operatorPressed:(UIButton *)sender {
58     if(self.userIsInTheMiddleOfTypingANumber){
59         [self.brain setOperand:[self.display text] doubleValue];
60         self.userIsInTheMiddleOfTypingANumber = NO;
61     }
62     NSString *operation = [[sender titleLabel]text];
63     double result = [self.brain performOperation:operation];
64     self.display.text = [NSString stringWithFormat:@"%g",result];
65 }
66
67 @end

```

- operatorPressed 함수이다. 만약에 앞에 숫자가 있었다면, brain에 operand 로 설정 해주고 boolean타입의 변수를 NO로 변경해준다. doubleValue로 바꾼 이유는 라벨에 텍스트는 스트링 값이기 때문에 바꿔주었다. 그리고 performOperation이면, 즉 “=”이면 결과값을 디스플레이에 보여

```
#import <Foundation/Foundation.h>
```

```
@interface CalculatorBrain : NSObject
```

```

- (void) setOperand:(double) newOperand;
- (double) performOperation: (NSString*)operation;

```

```
@end
```

준다.

- CalculatorBrain.h 화면이다. 새로운 operand와 계산을 실행하는 performOperation을 설정한다.

```

#import "CalculatorBrain.h"
@interface CalculatorBrain()
@property (nonatomic)double operand;
@property (nonatomic)double waitingOperand;
@property (nonatomic)NSString * waitingOperation;

-(void)performWaitingOperation;
@end

@implementation CalculatorBrain
-(double)performOperation:(NSString *)operation{
    if([operation isEqual:@"sqrt"]){
        self.operand = sqrt(self.operand);
    }
    else{
        [self performWaitingOperation];
        self.waitingOperation = operation;
        self.waitingOperand = self.operand;
    }
    return self.operand;
}

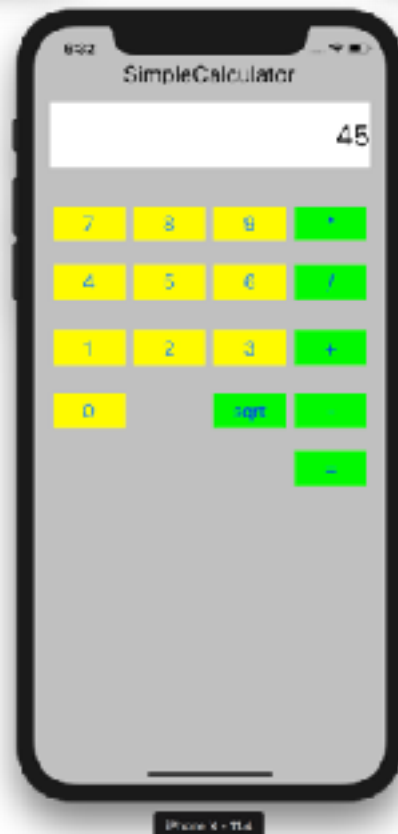
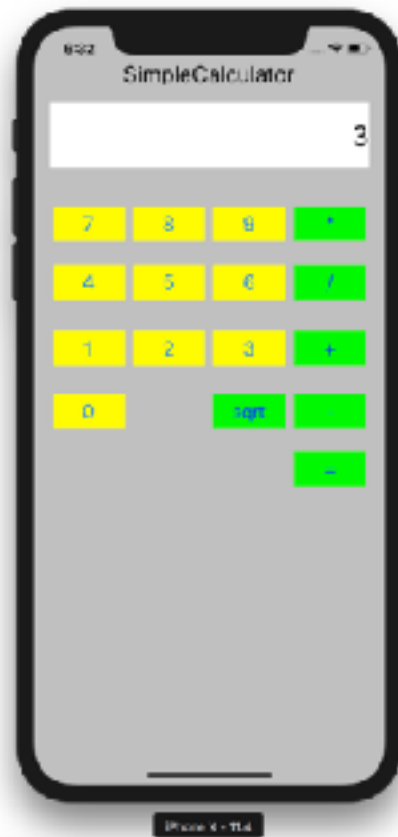
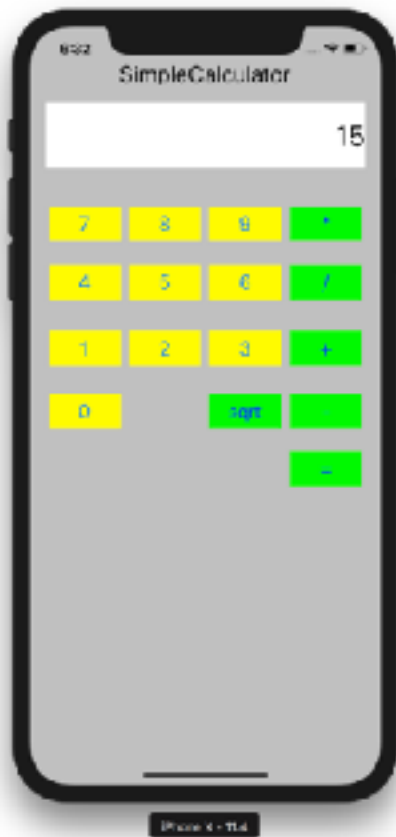
-(void) performWaitingOperation{
    if([self.waitingOperation isEqual:@"+"]){
        self.operand = self.waitingOperand + self.operand;
    }
    else if([self.waitingOperation isEqual:@"-"]){
        self.operand = self.waitingOperand - self.operand;
    }
    else if([self.waitingOperation isEqual:@"*"]){
        self.operand = self.waitingOperand * self.operand;
    }
    else if([self.waitingOperation isEqual:@"/"]){
        self.operand = self.waitingOperand / self.operand;
    }
}

@end

```

CalculatorBrain.h 부분에서 계산이 진행되는 함수들이다. 만약에 sqrt이면 Math 라이브러리에 내장되어 있는 제곱근에 관련된 함수를 실행하고, 그렇지 않다면 밑에 함수를 실행하고 operand와 operation 관련 해서 설정을 해준다. 그리고 밑에 함수는 각 연산마다의 계산하는 내용인데 waitingOperand 와 현재 operand를 각각 연산하는 형식으로 한다.

- 실행화면



15 * 3 의 연산을 한 캡처 화면이다.