

# AVAudioRecorder & System Sound Services

사운드 콘텐츠 응용

Dept. of Multimedia Science, Sookmyung Women's University, prof. JongWoo Lee

## Index

- AVAudioRecorder란?
  - 예제 따라하기(AudioRecorder)
- System Sound Service란?
  - 예제 따라하기(SysSound)

# AVAudioRecorder

## AVAudioRecorder란?

- 녹음을 위한 아주 간단한 인터페이스  
(다음시간에 또다른 방법인 조금더 복잡한 Audio Queue Services에 대해 배울 예정)
- 디스크에 녹음 데이터를 저장해야 함
- NSMutableData object에 녹음할 수 없음
- audio recorder를 사용하면
  - Record until the user stops the recording
  - Record for a specified duration
  - Pause and resume a recording
  - Obtain input audio-level data that you can use to provide level metering

- AVAudioRecorder Class

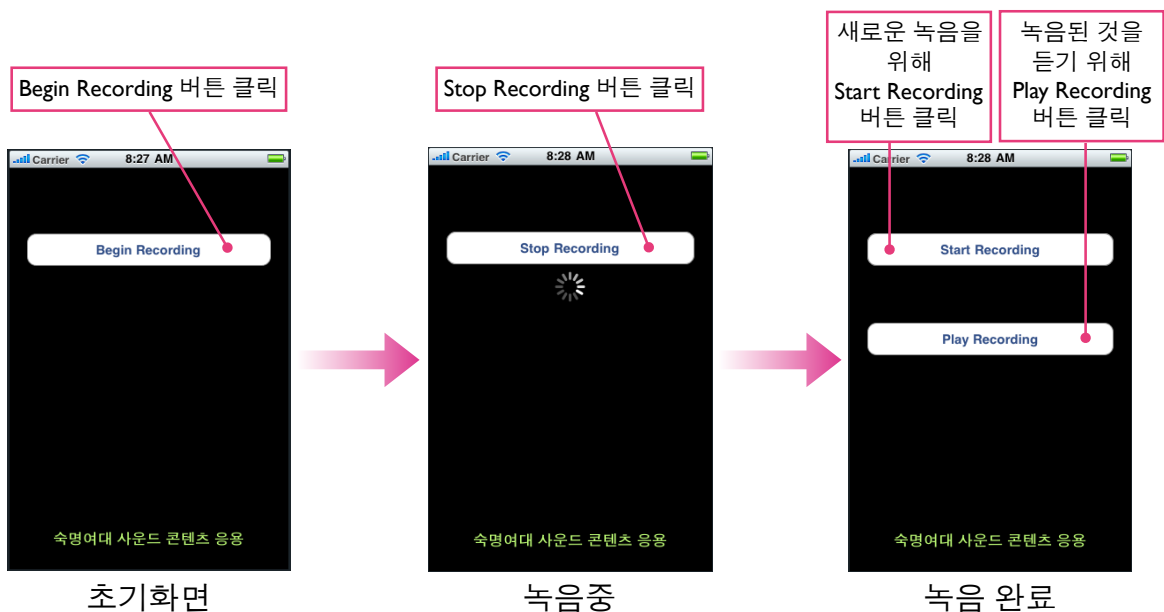
Inherits	NSObect
Conforms to	NSObect(NSObject)
Framework	/System/Library/Frameworks/AVFoundation.framework
Availability	Available in iOS 3.0 and later
Declared in	AVAudioRecorder.h

이제

3주차에서 배웠던 *AVAudioSession*과  
이전 주에 배웠던 *AVAudioPlayer*,  
그리고 *AVAudioRecorder*를 사용하여  
간단한 녹음기를 만들어 보자!

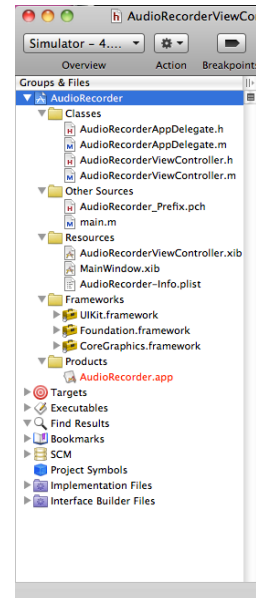
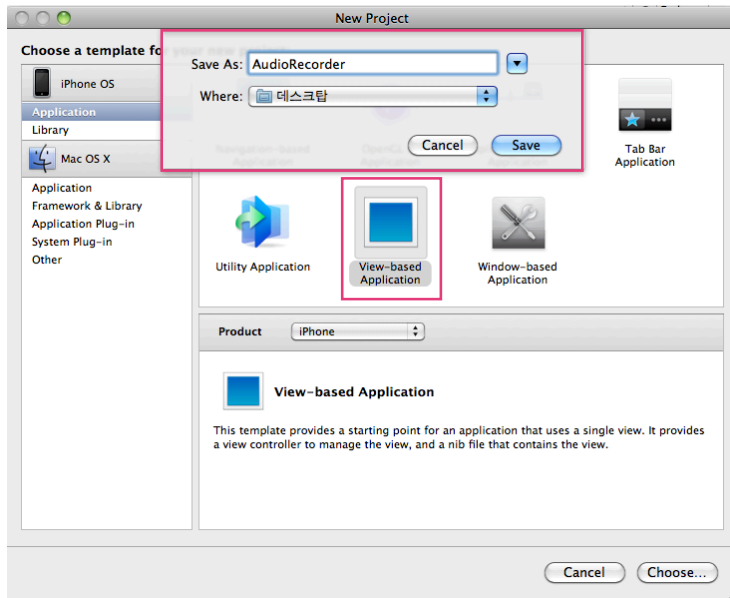
## AVAudioRecorder 예제

- 소스코드명 : AudioRecorder
- 예제 흐름



# Xcode\_프로젝트 생성

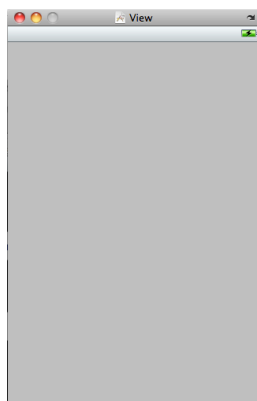
- Xcode에서 file-New Project
- View-based Application 선택후 choose버튼 클릭
- AudioRecorder라는 프로젝트 이름을 주고 저장할 곳을 선택하고 Save버튼 클릭



프로젝트 생성 후  
기본으로  
생성되는파일들

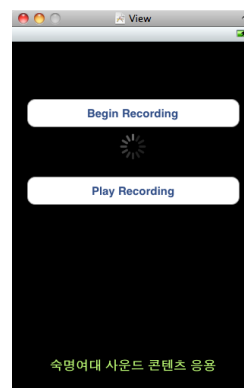
# Interface Builder\_UI 구성

- 생성된 파일 중 AudioRecorderViewController.xib를 클릭
- Interface Builder가 실행되며 View 창에 UI구성

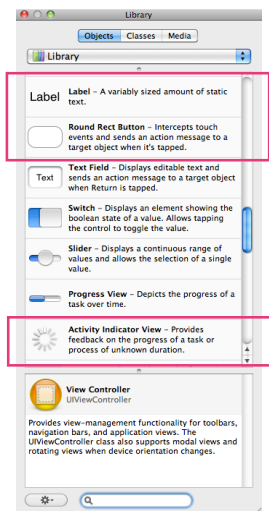


초기화면

1. View Attributes 창의 background에서 배경을 검정색으로 변경
2. Library 창에서 Round Rect Button 2개 Activity Indicator View와 Label을 각각 끌어와 내용(글자)을 삽입



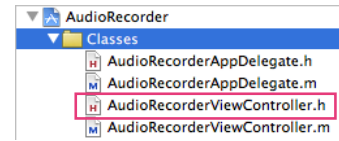
구성 예



다음단계: Interface BuilderView에 구성한 UI 객체를 Xcode와 연결해보자!

# Xcode\_AudioRecorderViewController.h

- Classes에 자동으로 생성된 4개의 파일에서 AudioRecorderViewController.h파일을 수정
- 이곳에서 Interface Builder에서 생성했던 UI들을 선언



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

#import <AVFoundation/AVFoundation.h>
#import <CoreAudio/CoreAudioTypes.h>

@interface record_audio_testViewController : UIViewController <AVAudioRecorderDelegate> {

    IBOutlet UIButton * btnStart;
    IBOutlet UIButton * btnPlay;

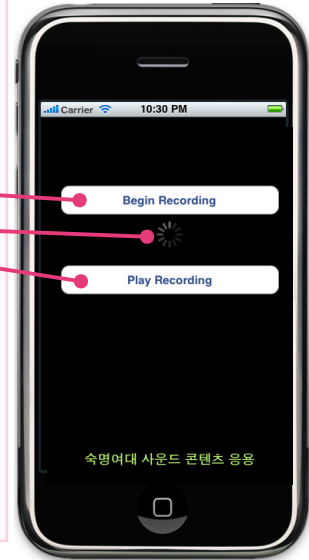
    IBOutlet UIActivityIndicatorView * actSpinner;
    BOOL toggle; // begin/start 버튼의 활성화 설정을 위한

    NSURL * recordedTmpFile;
    AVAudioRecorder * recorder;
    NSError * error;

}

@property (nonatomic, retain)IBOutlet UIActivityIndicatorView * actSpinner;
@property (nonatomic, retain)IBOutlet UIButton * btnStart;
@property (nonatomic, retain)IBOutlet UIButton * btnPlay;

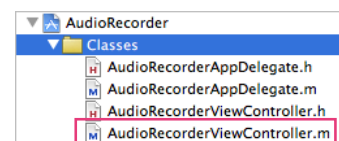
//사용자 정의 함수
- (IBAction) start_button_pressed; //시작 버튼이 눌렀을 때 액션 함수 선언
- (IBAction) play_button_pressed; //재생 버튼이 눌렀을 때 액션 함수 선언
@end
```



다음단계: 실제 함수들을 구현해보자!

# Xcode\_AudioRecorderViewController.m

- Classes에 자동으로 생성된 4개의 파일에서 AudioRecorderViewController.m파일을 수정
- 실제 사용될 함수들을 구현



## I. viewDidLoad{ }함수

```
#import "AudioRecorderViewController.h"

@implementation AudioRecorderViewController
@synthesize actSpinner, btnStart, btnPlay;

//nib파일에서 뷰가 로딩되는 상황을 구현
- (void)viewDidLoad {
    [super viewDidLoad];

    toggle = YES; //toggle을 true mode로 시작
    btnPlay.hidden = YES; //첫 실행에서는 play button을 숨김

    //AVAudioSession object의 instance를 생성
    AVAudioSession * audioSession = [AVAudioSession sharedInstance];

    //재생되고 녹음되는 동안의 audioSession을 세팅
    //AudioSession 카테고리를 AVAudioSessionCategoryPlayAndRecord로 설정
    [audioSession setCategory:AVAudioSessionCategoryPlayAndRecord error: &error];
    [audioSession setActive:YES error: &error]; //세션을 활성화
}

}
```

# Xcode\_AudioRecorderViewController.m계속

## 2. start\_button\_pressed{ }함수

```
-(IBAction) start_button_pressed{
    if(toggle)
    {
        toggle = NO;
        [actSpinner startAnimating];
        [btnStart setTitle:@"Stop Recording" forState: UIControlStateNormal ];
        btnPlay.enabled = toggle;
        btnPlay.hidden = !toggle;

        //recording session의 시작
        //dictionary object를 setup
        NSMutableDictionary* recordSetting = [[NSMutableDictionary alloc] init];
        [recordSetting setValue : [NSNumber numberWithInt:kAudioFormatAppleIMA4] forKey:AVFormatIDKey];

        //녹음을 위해 사용되는 임시 파일을 생성
        recordedTmpFile = [NSURL fileURLWithPath:[NSTemporaryDirectory() stringByAppendingPathComponent:
            [NSString stringWithFormat:@"%f.%f@", [NSDate timeIntervalSinceReferenceDate] * 1000.0, @"caf"]]];

        //Setup the recorder to use this file and record to it.
        recorder = [[ AVAudioRecorder alloc] initWithURL:recordedTmpFile settings:recordSetting error:&error];

        //녹음을 시작하기 위해 recorder를 사용
        [recorder setDelegate:self];

        //오디오 파일을 생성하고 녹음을 위해 시스템을 준비
        //녹음 프로세스를 시작하고 초기화를 위해 이를 호출
        [recorder prepareToRecord];

        //실제 녹음을 시작
        [recorder record];
    }
}
```

소스코드 다음장에 계속

# Xcode\_AudioRecorderViewController.m계속

## 2. start\_button\_pressed{ }함수 (계속)

```
else
{
    toggle = YES;
    [actSpinner stopAnimating];
    [btnStart setTitle:@"Start Recording" forState:UIControlStateNormal ];
    btnPlay.enabled = toggle;
    btnPlay.hidden = !toggle;

    //Stop the recorder.
    [recorder stop];
}
}
```

## 3. play\_button\_pressed{ }함수

```
-(IBAction) play_button_pressed{
    //play 버튼을 눌렀을 때
    //지금 막 녹음된 파일을 재생하기 위한 AVAudioPlayer을 셋업
    AVAudioPlayer * avPlayer = [[AVAudioPlayer alloc] initWithContentsOfURL:recordedTmpFile error:&error];
    [avPlayer prepareToPlay];
    [avPlayer play];
}
```

# Xcode\_AudioRecorderViewController.m계속

## 4. viewDidLoad{ }함수

```
- (void)viewDidLoad {
    //main view에서 보유하고 있는 subview들을 해제
    // e.g. self.myOutlet = nil;
    //임시 파일을 삭제
    NSFileManager * fm = [NSFileManager defaultManager];
    [fm removeItemAtPath:[recordedTmpFile path] error:&error];

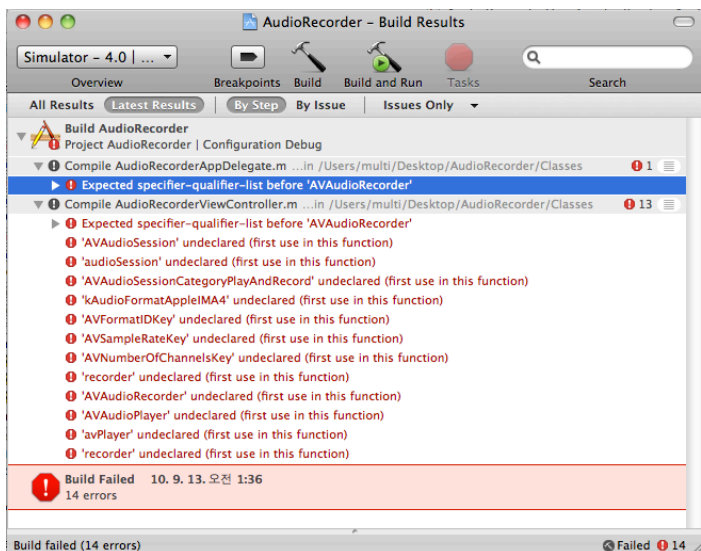
    //보유한 object들에 dealloc호출
    [recorder dealloc];
    recorder = nil;
    recordedTmpFile = nil;
}
```

## 오류가 난다!

- 코드를 작성한 후 Build and Run
- 그러나.... 오류가 난다!!

AVAudioSession  
audioSession  
AVAudioSessionCategoryPlayAndRecord  
kAudioFormatAppleIMA4  
AVFormatIDKey  
등등이 선언이 되어있지 않다.

이를 위해 Framework를 추가!  
- AVFoundation.framework  
- CoreAudio.framework

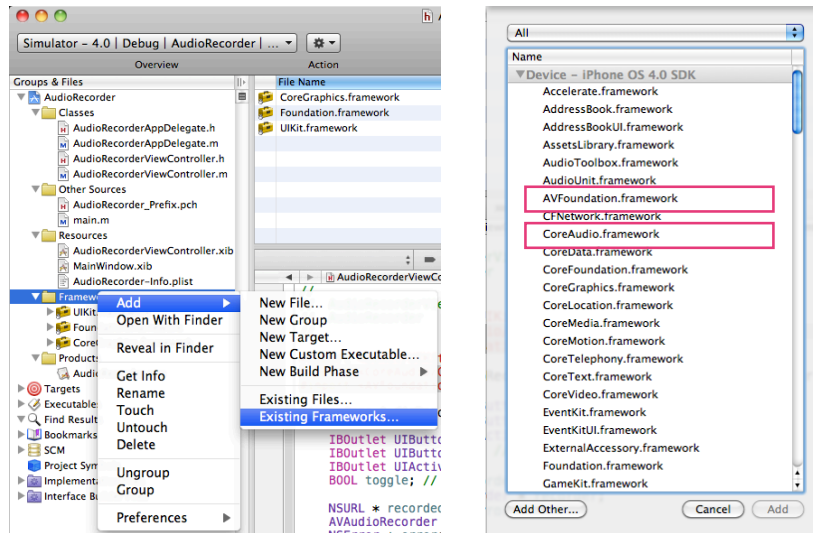


# Framework 추가하기

- AudioRecorderViewController.h에 다음을 import하고

```
#import <CoreAudio/CoreAudioTypes.h>
#import <AVFoundation/AVFoundation.h>
```

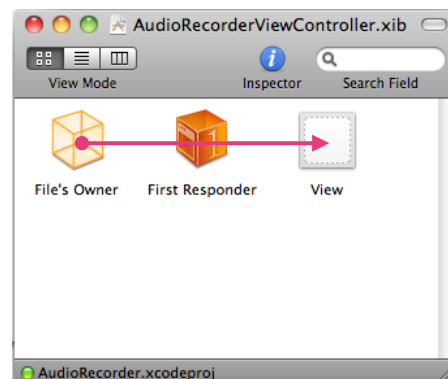
- Groups & Files의 Frameworks에 CoreAudio, AVFoundation을 Add



# Xcode와 Interface Builder 연결하기(1/4)

- AudioRecorderViewController.xib파일을 열어 Interface Builder에서 설정해야 하는 것들
  1. View와 File's Owner를 연결한다.
  2. xib 창의 File's Owner의 Outlet을 연결한다.
  3. Begin Recording과 Play Recording 버튼을 연결한다.
  4. Activity Indicator View 속성을 지정한다.

1. View와 File's Owner 연결
  - ✓ File's Owner에 속한 View라는 것을 알려주기 위함
  - ✓ File's Owner 아이콘을 control 키를 누르면서 View로 연결

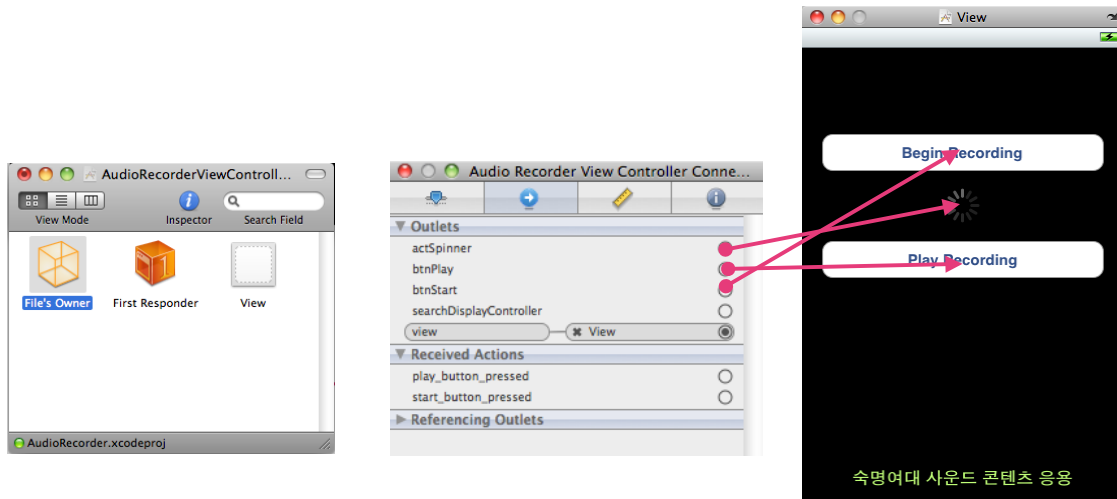




# Xcode와 Interface Builder 연결하기(2/4)

## 2. xib 창의 File's Owner의 Outlet을 연결한다.

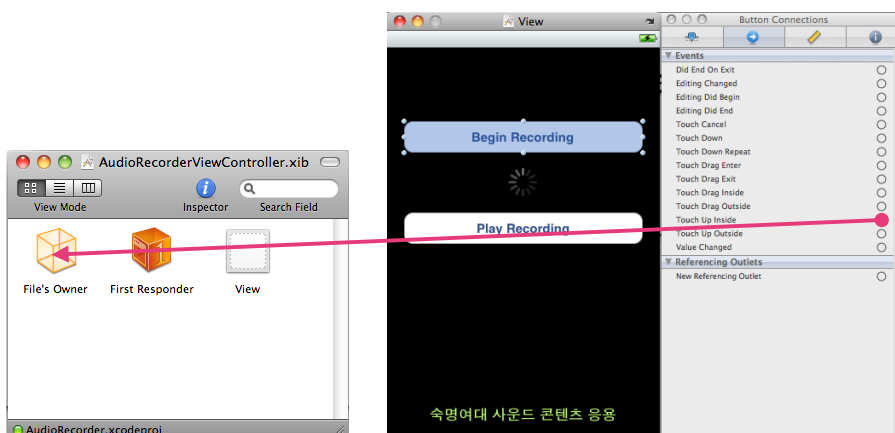
- ✓ File's Owner가 선택된 상태에서
- ✓ Inspector창의 두번째 탭 Audio Recorder View Controller Connections에 Xcode에서 정의했던 Outlet의 목록이 보임
- ✓ 각 항목을 드래그하여 View 창의 버튼과 Activity Indicator에 연결



# Xcode와 Interface Builder 연결하기(3/4)

## 3. Begin Recording과 Play Recording 버튼을 연결

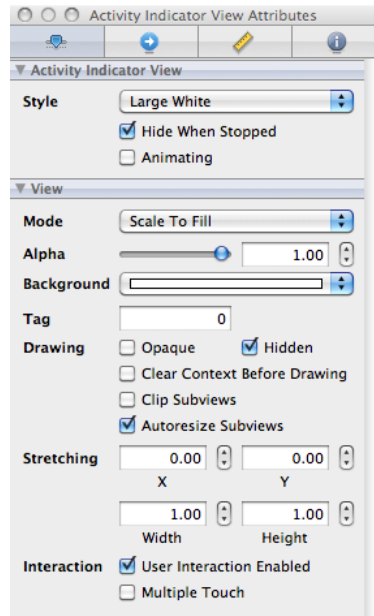
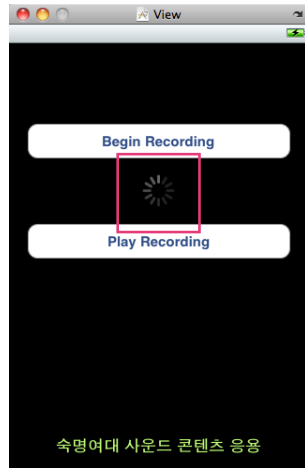
- ✓ Begin Recording이 선택된 상태에서
- ✓ Inspector창의 두번째 탭 Button Connections의 Touch Up Inside의 오른쪽 동그라미를 클릭하여 File's Owner 아이콘으로 드래그
- ✓ 작은 회색메뉴가 나타나면 start\_button\_pressed를 선택
- ✓ 같은 방법으로 Play Recording 버튼이 선택된 상태에서 Touch Up Inside를 File's Owner의 play\_button\_pressed로 연결



# Xcode와 Interface Builder 연결하기(4/4)

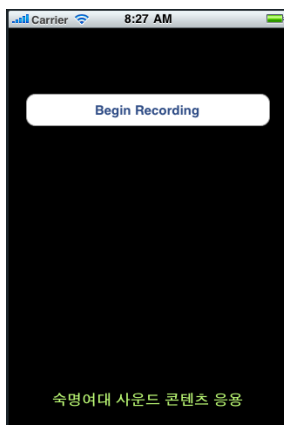
## 4. Activity Indicator View의 속성을 지정

- ✓ Activity Indicator View가 선택된 상태에서
- ✓ 정지된 상태에서는 보이지 않게 Hide When Stopped에 체크

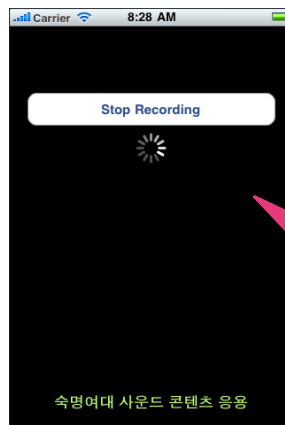


## 실행해봅시다!

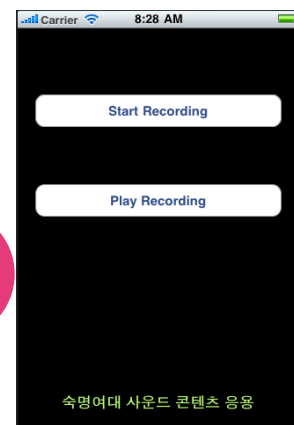
- Xcode로 돌아와 Build & Run을 눌러 실행



초기화면



녹음중



녹음 완료& 재생

Q. Simulator를 돌렸지만 녹음을 어떻게....?  
iMac에 마이크가 어디에 있나요?



여기있습니다!  
보이시나요?  
이곳에 대고 크~게 말해보세요!

# System Sound Service

## System Sound Service(1/3)

- 버튼을 클릭하는 것과 같은 사용자 인터페이스 사운드 효과를 재생하는 서비스
- 기기의 진동을 호출하는 기능을 포함
- level, positioning, audio session과 같은 것이 필요하지 않는 짧은 사운드를 재생하기 위함
- 제한사항
  - 30초 이하의 사운드
  - Linear PCM이나 IMA4(IMA/ADPCM)포맷
  - .caf, .aif 또는 .wav 파일로 패키징 되어야 함
- playback method
  - AudioServicesPlaySystemSound 함수(일반적인 짧은 사운드 효과 재생)
  - AudioServicesPlayAlertSound 함수(경고로서의 짧은 사운드 재생)
- Declared in | AudioServices.h in Audio Toolbox framework

## System Sound Service(2/3)

- Functions by Task

Creating and Disposing of System Sound Objects	AudioServicesCreateSystemSoundID
	AudioServicesDisposeSystemSoundID
Playing Sounds	AudioServicesPlayAlertSound
	AudioServicesPlaySystemSound
Adding and Removing System Sound Callbacks	AudioServicesAddSystemSoundCompletion
	AudioServicesRemoveSystemSoundCompletion

- AudioServicesPlaySystemSound 함수

- very simply play short sound files
- Sounds play at the current system audio volume, with no programmatic volume control available
- Sounds play immediately
- Looping and stereo positioning are unavailable
- Simultaneous playback is unavailable: You can play only one sound at a time

- iPhone에서 진동을 호출할 수 있음(시뮬레이터에서 구동 불가)

- AudioServicesPlaySystemSound 함수에서 kSystemSoundID\_Vibrate 상수를 호출

## System Sound Service(3/3)

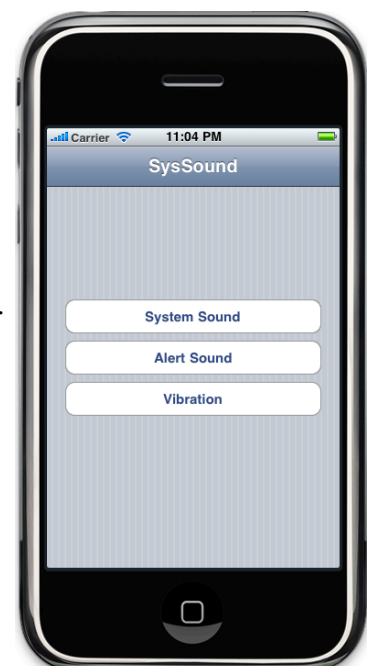
- AudioServicesPlaySystemSound 함수로 시스템 사운드를 재생하기 위해

1. sound ID object를 생성하고
2. then, 재생 함수를 호출한다.

- 소스코드명: SysSound

- 예제설명

- tap.aif 파일을 사용하여
- 시스템사운드와 경고 사운드 재생 메소드의 사용법을 익혀본다.

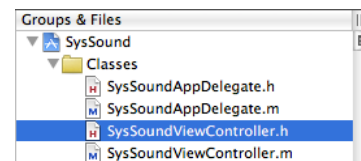


# Xcode\_프로젝트 생성

- Xcode에서 file-New Project
- View-based Application 선택후 choose버튼 클릭
- AudioRecorder라는 프로젝트 이름을 주고 저장할 곳을 선택하고 Save버튼 클릭
- AudioRecorder와 같은 방법으로 Interface Builder에서 SysSoundViewController.xib의 레이아웃을 구성
- AudioToolbox.framework 추가

# Xcode\_SysSoundViewController.h

- Classes에 자동으로 생성된 4개의 파일에서 SysSoundViewController.h파일을 수정



```
#import <UIKit/UIKit.h>
#include <AudioToolbox/AudioToolbox.h>

@interface SysSoundViewController : UIViewController {

    CFURLRef    soundFileURLRef;
    SystemSoundID soundFileObject;
}

@property (readwrite) CFURLRef    soundFileURLRef;
@property (readonly) SystemSoundID soundFileObject;

//UI와 객체와 연결하는 액션 메소드
- (IBAction) playSystemSound: (id) sender;
- (IBAction) playAlertSound: (id) sender;
- (IBAction) vibrate: (id) sender;

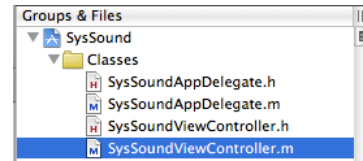
@end
```



다음단계: 실제 함수들을 구현해보자!

# Xcode\_SysSoundViewController.m

- Classes에 자동으로 생성된 4개의 파일에서 SysSoundViewController.m파일을 수정
- 실제 사용될 함수들을 구현



## 1. viewDidLoad{ }함수

```
#import "SysSoundViewController.h"

@implementation SysSoundViewController

@synthesize soundFileURLRef;
@synthesize soundFileObject;

- (void) viewDidLoad {
    [super viewDidLoad];

    // Provide a nice background for the app user interface.(애플의 설명 nice background)
    self.view.backgroundColor = [UIColor groupTableViewBackgroundColor];

    // 오디오 소스 파일을 위한 URL을 생성
    NSURL *tapSound = [[NSBundle mainBundle] URLForResource:@"tap"
                                                            withExtension:@"aif"];

    // CFURLRef instance로서 URL을 저장
    self.soundFileURLRef = (CFURLRef) [tapSound retain];

    // 사운드 파일을 표현하기 위한 시스템 사운드 객체를 생성
    AudioServicesCreateSystemSoundID (
        soundFileURLRef,
        &soundFileObject
    );
}
```

# Xcode\_SysSoundViewController.m계속

## 2. 사용자 액션 함수 정의

```
// System Sound button을 눌렀을 때의 액션
- (IBAction) playSystemSound: (id) sender {
    AudioServicesPlaySystemSound (soundFileObject);
}

// Alert Sound button을 눌렀을 때의 액션
- (IBAction) playAlertSound: (id) sender {
    AudioServicesPlayAlertSound (soundFileObject);
}

// Vibrate button을 눌렀을 때의 액션, 시뮬레이터에서는 작동하지 않음
- (IBAction) vibrate: (id) sender {
    AudioServicesPlaySystemSound (kSystemSoundID_Vibrate);
}

//메모리 해제
- (void) dealloc {
    AudioServicesDisposeSystemSoundID (soundFileObject);
    CFRelease (soundFileURLRef);
    [super dealloc];
}
```

# Xcode와 Interface Builder 연결 & 실행

- SysSoundViewController.xib파일을 열어 Interface Builder에서 설정해야 하는 것들
  1. View와 File's Owner를 연결한다.
  2. xib 창의 File's Owner의 Outlet을 연결한다.
  3. System Sound,Alert Sound와 Vibration 버튼을 연결한다.
- Xcode에서 실행시켜보자.
  - wav파일로도 실행시켜보기  
(소스코드 내 test.wav 파일이 있음)

