

S H O U T · O U R · P A S S I O N · T O G E T H E R

inno SOPT

안드로이드파트 2nd

SHOUT OUR PASSION TOGETHER
SOPT

Contents

22기 SOPT
Android 1st



01

Activity

컴포넌트
Activity?
생명주기

02

Fragment

Fragment?
생명주기

03

실습

오늘 할 일
진짜 오늘 할 것

04

과제^{^^}

세미나 때 공개

01

Activity

01 Activity

01
컴포넌트



- **Activity**
 - UI를 구성하는 컴포넌트
 - 매니페스트에 등록 되어 있어야 함
 - 안드로이드 어플리케이션은 최소 1개 이상의 액티비티가 있어야 함
 - 쉽게 생각하면 사용자가 보는 화면

- **Service**
 - 백그라운드에서 실행되는 컴포넌트
 - 모든 서비스는 Service클래스 상속 받음
 - 한 번 실행되면 다른 어플로 이동하더라도 종료되지 않음
 - Ex 뮤직 플레이어

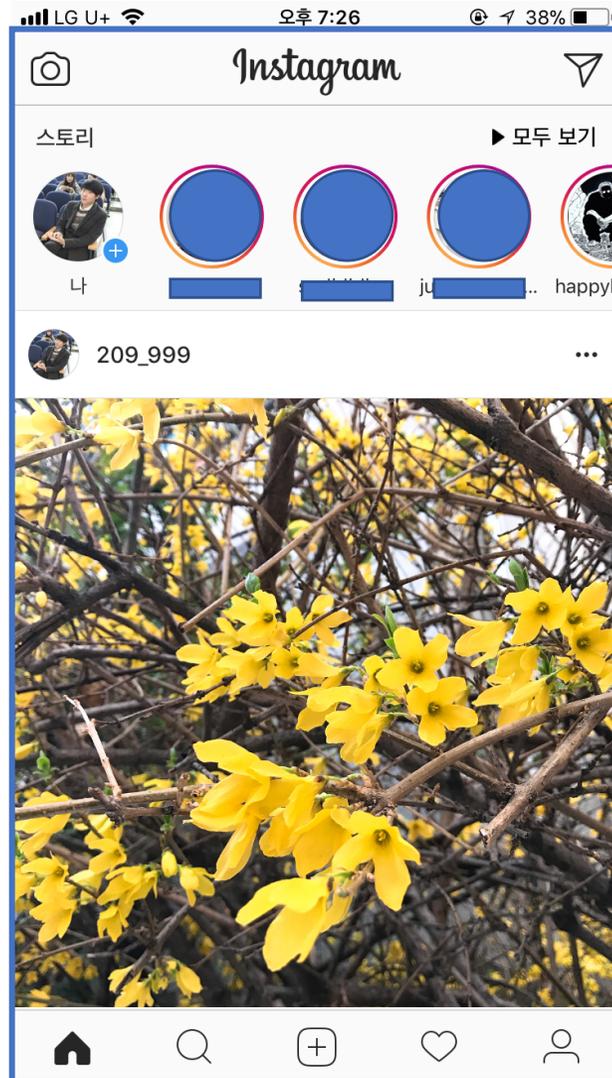
- **BroadCast Receiver**
 - Broadcast는 시스템에서 발생하는 이벤트이며 이것의 Receiver은 특정 Broadcast에 반응하는 컴포넌트
 - 신호만 기다릴 뿐 별도의 UI를 거치지 않는다
 - 배터리 부족, 네트워크 전송이 대표적인 예

- **Content Provider**
 - 어플리케이션간 데이터 공유를 위한 컴포넌트
 - 유일한 합법적 데이터 공유 장치
 - 주소록 Content Provider를 통해 제공

- **Intent**
 - 4대 컴포넌트 간 통신 수단
 - 대표적으로 startActivity()가 있다.
 - 인텐트를 통하여 다른 어플리케이션 활성화 가능

01 Activity

02 Activity?



01 Activity

03
생명주기



메소드	동작	다음 메소드
OnCreate()	액티비티가 생성될 때 호출되며 사용자 인터페이스 초기화에 사용됨.	onStart()
OnRestart()	액티비티가 멈췄다가 다시 시작되기 바로 전에 호출됨.	onStart()
OnStart()	액티비티가 사용자에게 보여지기 바로 직전에 호출됨.	onResume() or onStop()
OnResume()	액티비티가 사용자와 상호작용하기 바로 전에 호출됨.	onPause()
OnPause()	다른 액티비티가 보여질 때 호출됨. 데이터 저장, 스레드 중지 등의 처리를 하기에 적당한 메소드.	onResume() or onStop()
OnStop()	액티비티가 더이상 사용자에게 보여지지 않을 때 호출 됨. 메모리가 부족할 경우에는 onStop()이 호출되지 않을 수도 있음.	onRestart() or onDestroy()
OnDestroy()	액티비티가 소멸될 때 호출됨. finish() 메소드가 호출 되거나 시스템이 메모리 확보를 위해 액티비티를 제거할 때 호출됨.	없음

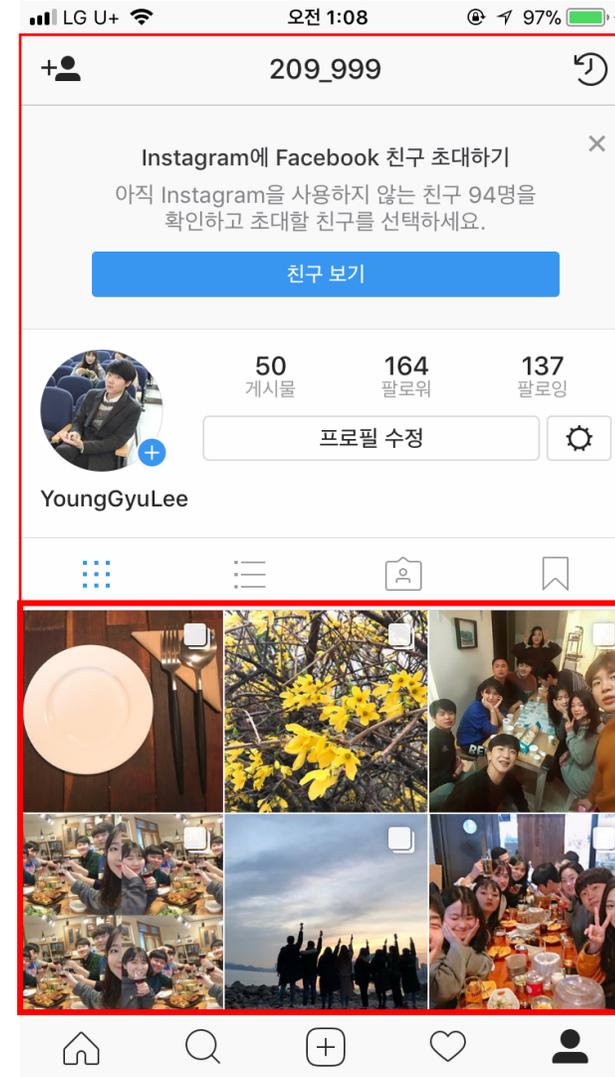
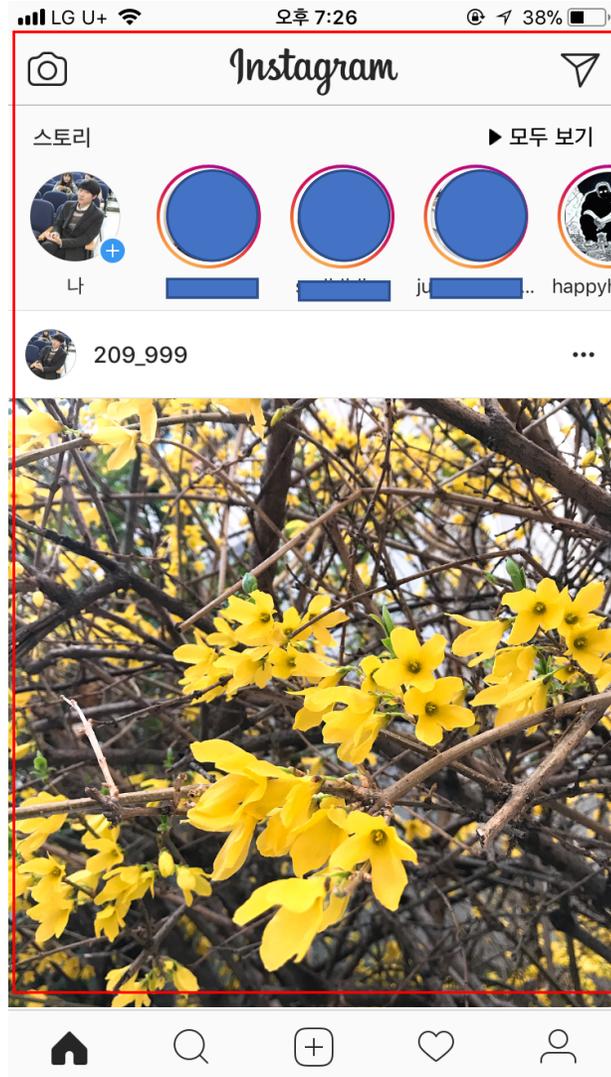
02

Fragment

02 Fragment

01
Fragment?

SHOUT OUR PASSION TOGETHER
SOPT



02 Fragment

01
Fragment?



- 일종의 하위 Activity
- 자체 생명주기
- 자체 입력 이벤트
- 값 전달은 **Bundle**로 함(Activity가 Intent로 한 것과는 대조적)
- **Tag**를 달 수 있다.
- **FragmentManager**를 통해 관리

FragmentManager

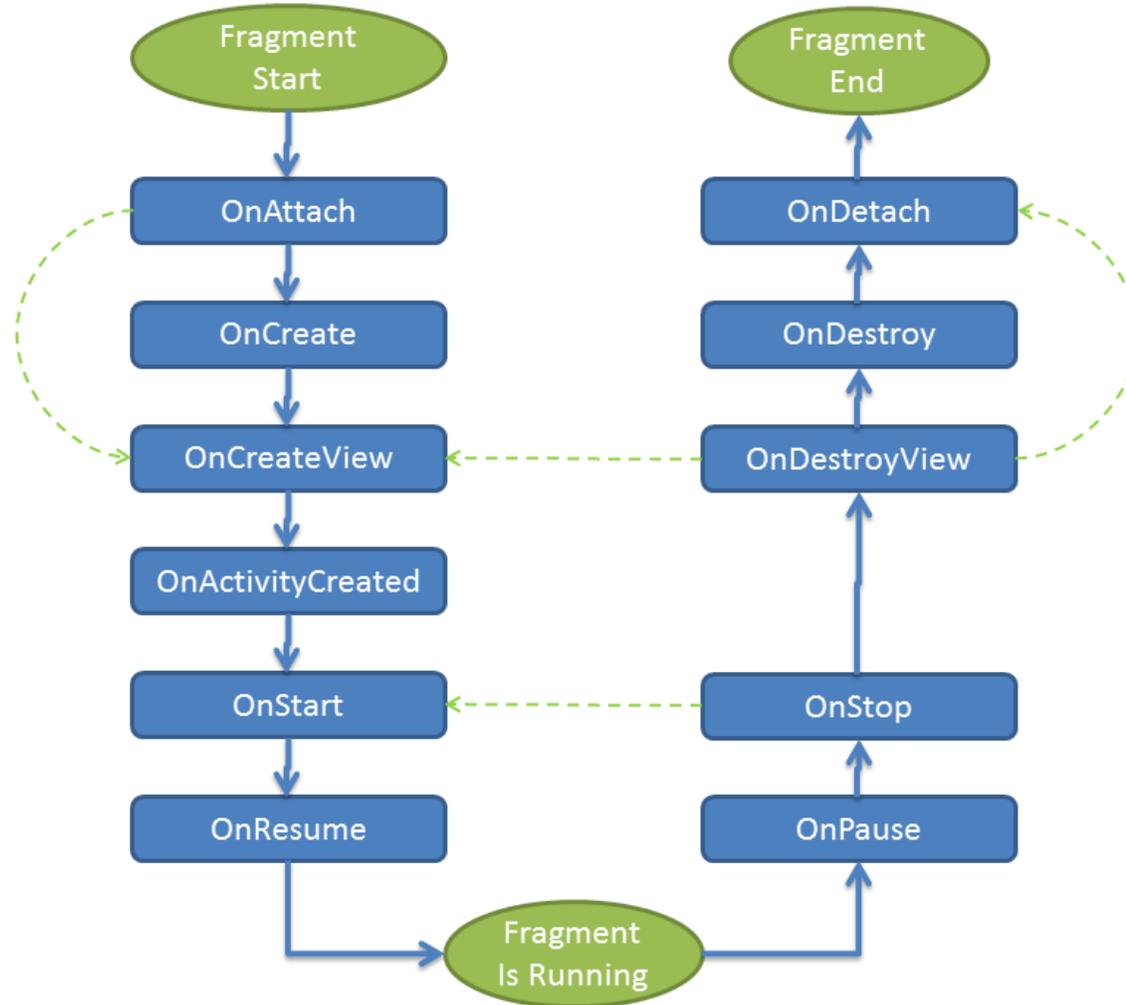
- Fragment 관리를 목적으로 하는 클래스
- **FragmentManager** 클래스를 통해 **FragmentManagerTransaction** 객체를
 - **FragmentManagerTransaction**을 통해 동적 관리
- 코틀린에서는 **supportFragmentManager**를 통해 생성

FragmentManager

- Fragment **추가, 삭제, 전환**을 도와주는 클래스
- **addToBackStack()** 함수를 호출해서 교체되는 Fragment를 back stack에 저장
- 항상 마지막에 변경사항 적용을 위해 **commit()**을 해야 함.

02 Fragment

02
생명주기



03

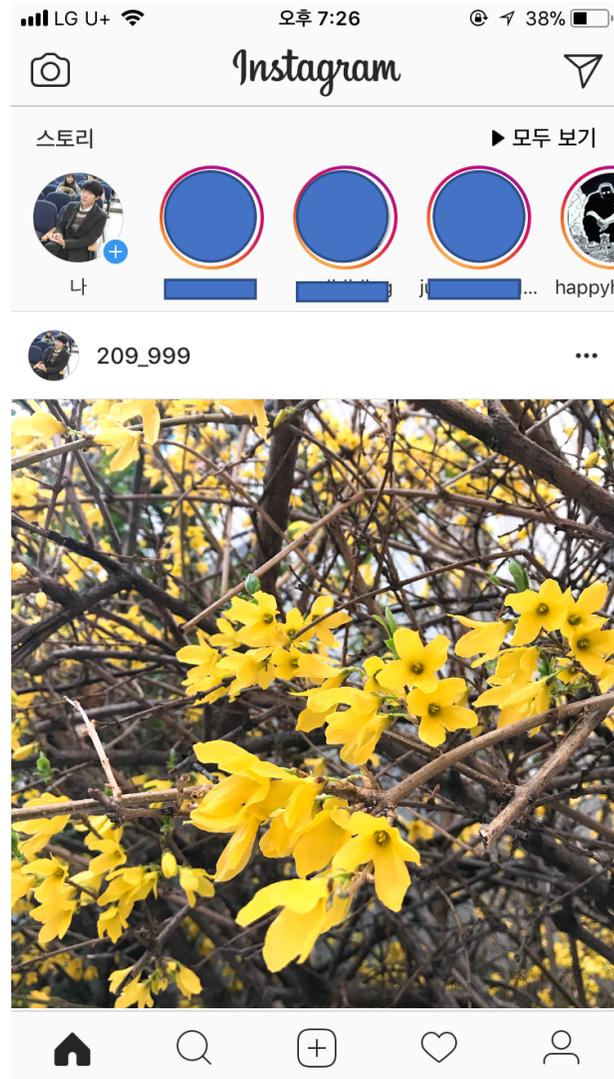
실습

또 눈치 채셨겠지만..

03 실습

01
오늘 할 일(세상..)

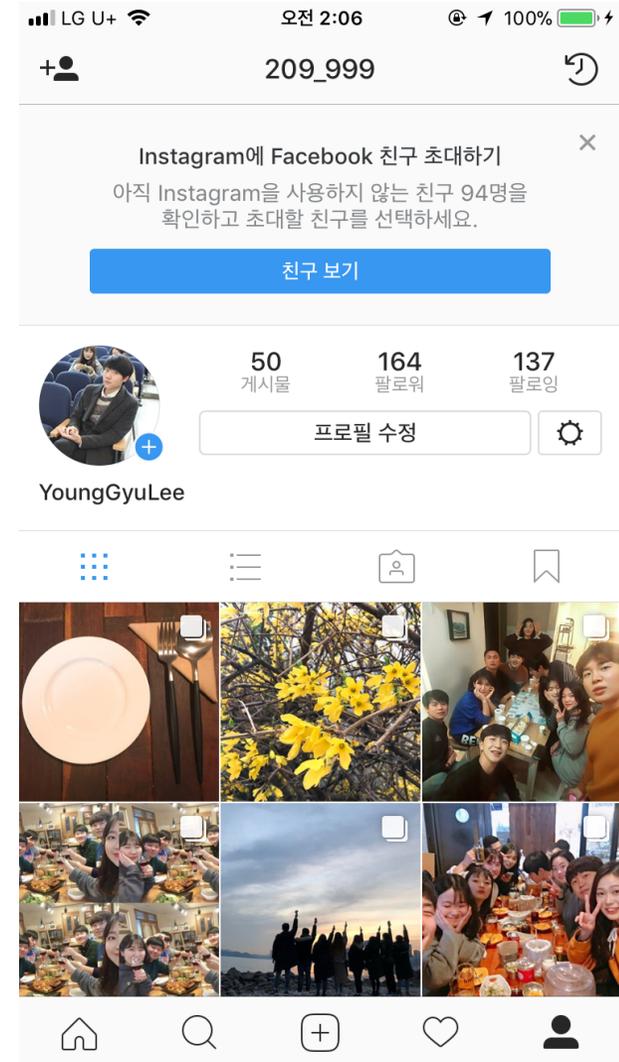
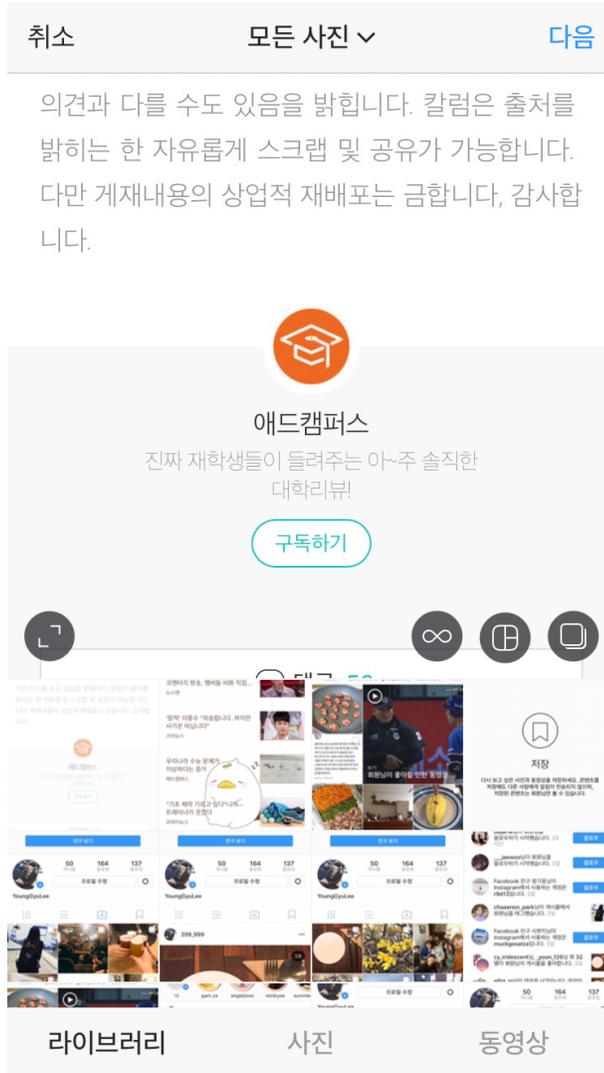
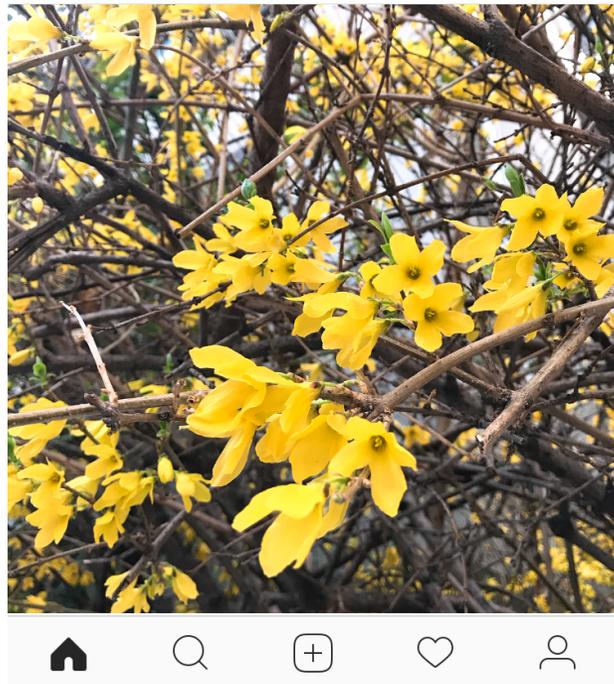
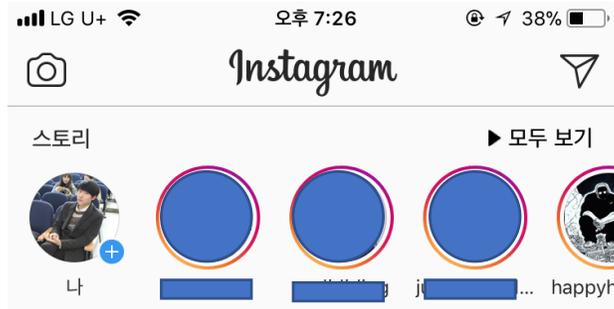
SHOUT OUR PASSION TOGETHER
SOPT



또 인스타그램을 만들 것입니다.

03 실습

01 오늘 할 일(오마이...)



단! 화면 전환 위주로!

03
실습

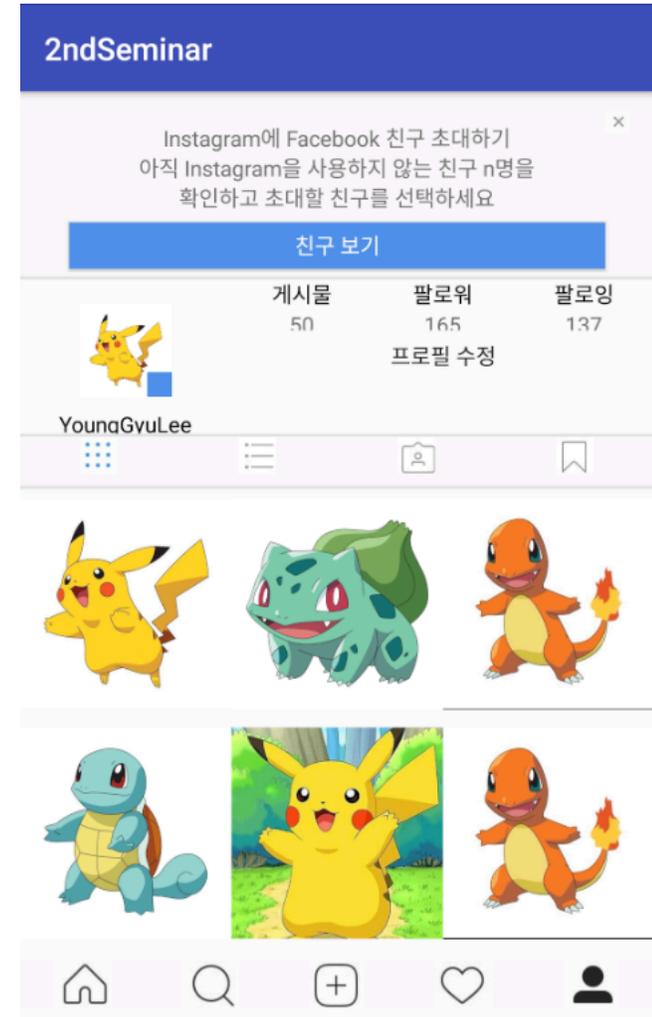
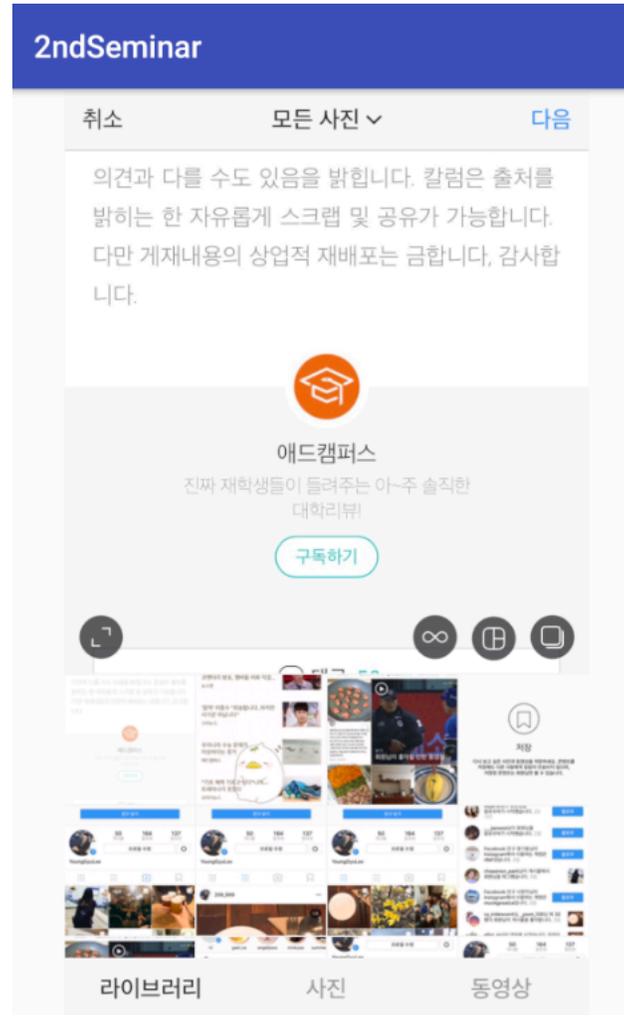
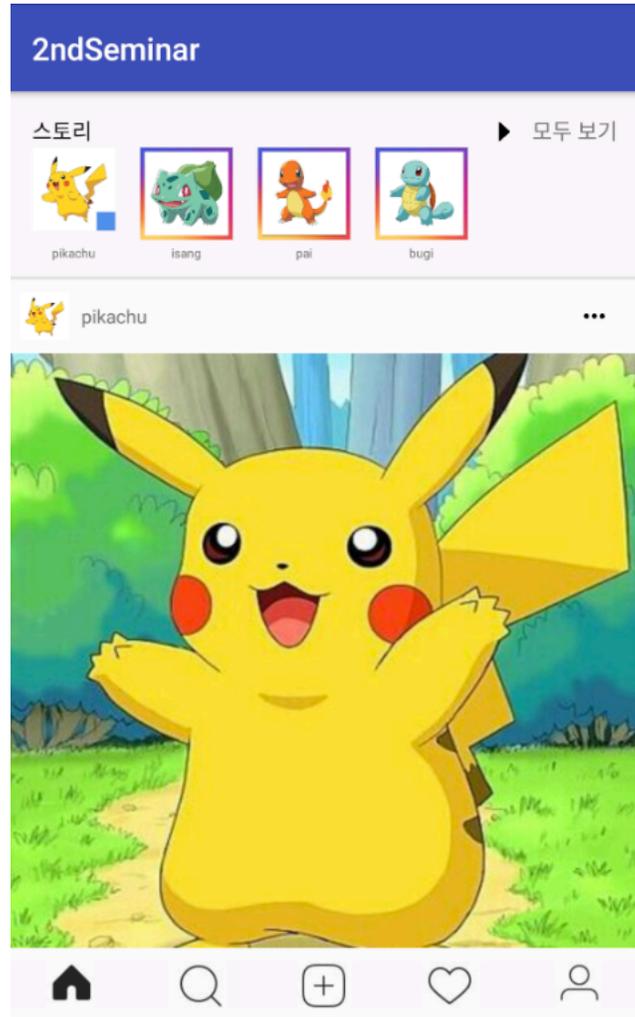
01
오늘 할 일



물론

03 실습

02
진짜 오늘 할 것(두둥..!)



딱 요정도..!(간단하쥬?^~^)

그럼 오늘도 힘차게!

그전에..!

잠시 쉬어여 😊

04

과제 ^~^

S H O U T · O U R · P A S S I O N · T O G E T H E R

inno SOPT

끼움~~

SHOUT OUR PASSION TOGETHER
SOPT