



시작할게요! 제 세션에서는 여러분이 궁금한게 생기시면 언제든지 제 말을 끊고 손들어서 질문해주시면됩니다! 제가 기억력이 안좋아서 바로바로 질문해주지 않으시면 까먹을수도있으니까 바로바로 말 끊어버리고 질문주세요

CONTENTS



01

통신이란?

02

통신 종류

03

HTTP란?

04

안드로이드에선 어떻게?

05

실습하기!

06

HTTP 좀 더 둘러보기

오늘 6개 챕터로 진행을 할텐데 저번 설문에서 스터디가 조금 어렵다는 말이 있어서 최대한 기본적인 내용과 추상적인 설명으로 진행을 할게요
앞부분에선 조금 딱딱하지만 통신 종류나 http가 뭔지 얘기해보고 안드로이드에선 어떻게 사용할까 알아본뒤 바로 실습을 해볼거예요!



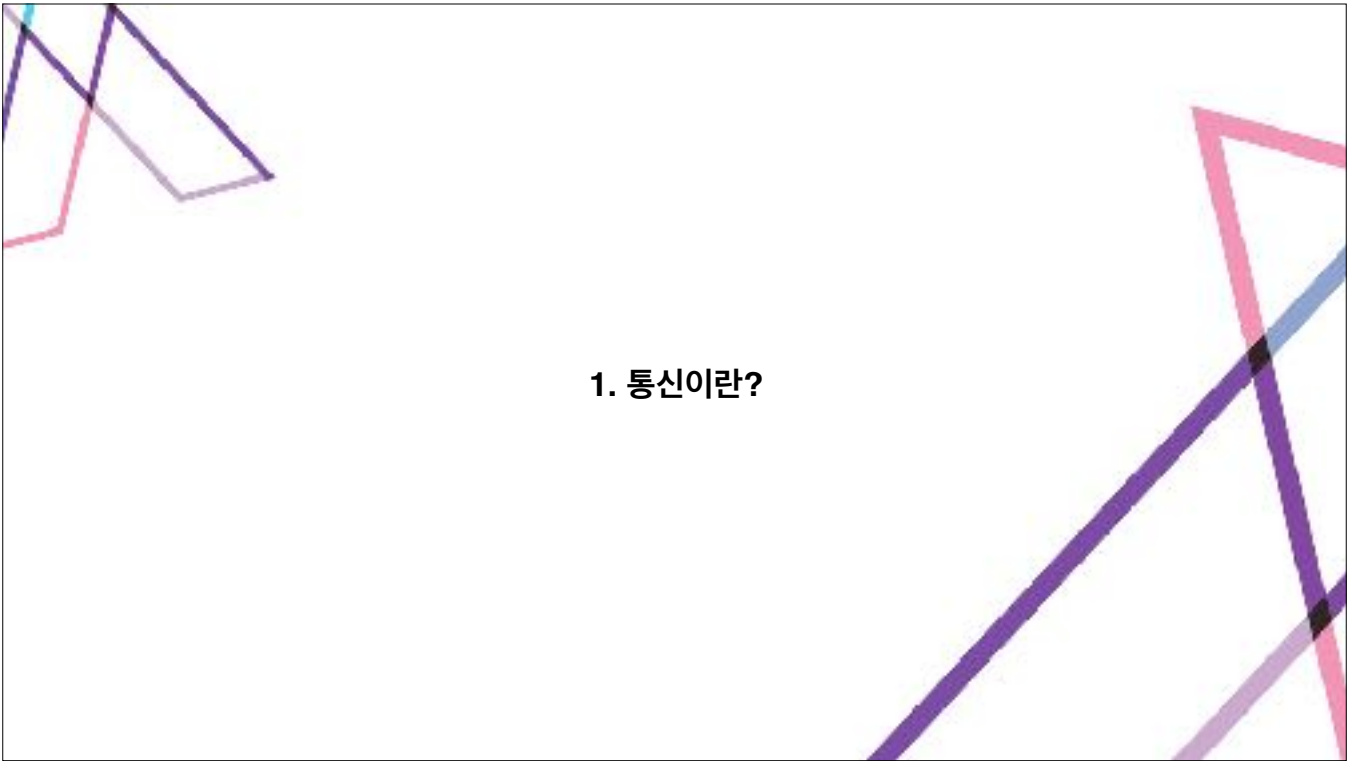
1. 통신이란?

자 그래서 이번주 스터디에서는 통신이란 주제에 대해서 배워볼게요. 시작하기에 앞서 오늘 스터디 목표부터 짚고갈게요.
여러분들은 오늘 스터디가 끝나면, 여태까지 배운 내용과 오늘 배운 내용만으로 조합해서 이따 실습시간때 이렇게 생긴 미세먼지경보앱을 만들어 볼 수 있을거예요.

01 통신이란?



재밋게 실습 준비했으니 앞부분이 조금 지루해도 잘 따라와주세요!



첫번째파트는 통신이란 무엇일까? 인데요

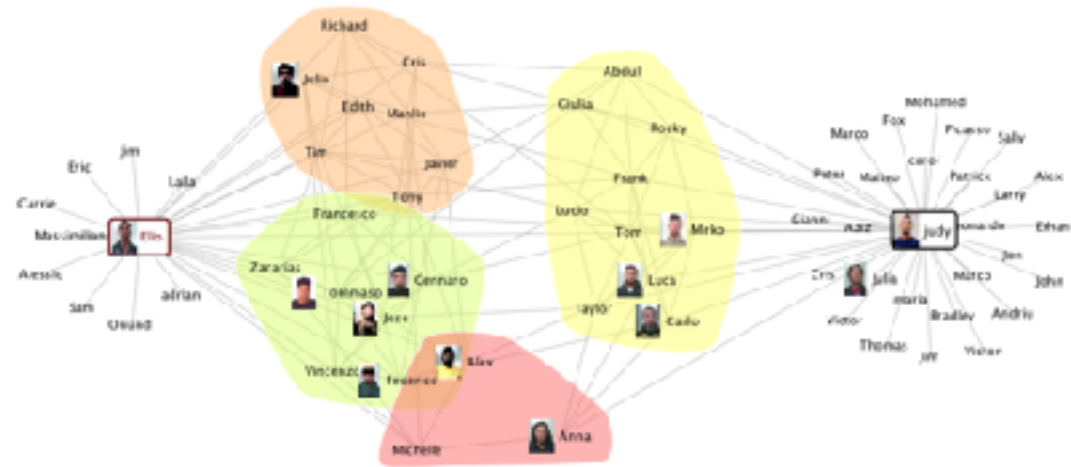
사전적 정의를 보면

소식을 전하는 것으로 여러가지 매체를 이용해 정보나 의사를 전하는것을 의미한다. 라고 합니다.

일상에서 예시를 찾아보면 좀 더 이해하기 쉬울것같은데

우리들이 개발하고 쓰는 스마트폰 앱에서 통신을 사용하는 경우가 어떤게있을까요?

01 통신이란?



네 그렇습니다. 조금만 생각해보면 우리가 보이는곳이나 안보이는곳 등등 이렇게 많은 부분에서 통신을 하고있어요
우리가 스마트폰을 사용하고있을때나 사용하지 않을때나 수많은 데이터가 통신을 통해 왔다갔다 하고 있어요

02 통신종류

이름	주용도	장점	단점
Bluetooth	오디오, 저전력통신	저전력, 저가격	느린속도, 작은거리
Socket	실시간통신	빠른통신속도	자원부담, 복잡성
Http	웹브라우저, API	connectless	stateless
Serial	유선 데이터	빠르고 안정적	유선연결이 필요

통신을 많이 하는만큼 다양한 통신방법이 있습니다.
우리가 개발하면서 접할 여지가 많은것들을 찾아보면
소켓 통신, 시리얼통신, 블루투스 통신, http 통신 등이 있겠네요

3. HTTP란?

03 HTTP란?

HTTP

(Hyper Text Transfer Protocol)

03 HTTP란?

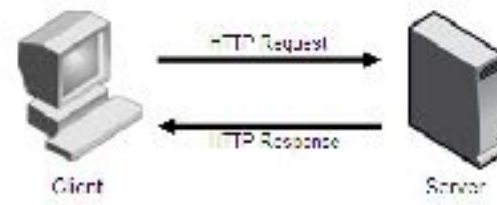
HTTP

(Hyper Text Transfer Protocol)

#stateless #connectless #getpost #headerbody #status

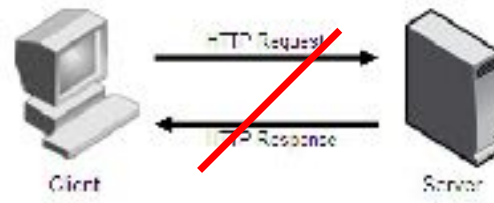
03 HTTP란?

#stateless #connectless



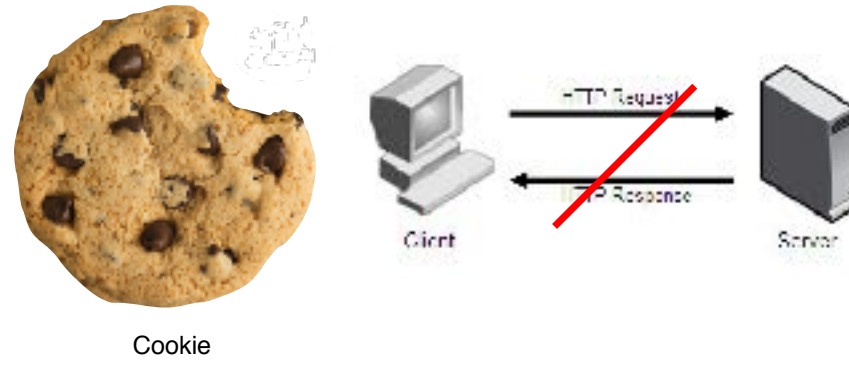
03 HTTP란?

#stateless #connectless



03 HTTP란?

#stateless #connectless



03 HTTP란?

#getpost

GET	데이터 검색
POST	데이터 추가(생성)
PUT	데이터 전송(생성/업데이트)
PATCH	데이터 업데이트
DELETE	데이터 삭제
OPTION	가능한 method 검색
...	

03 HTTP란?

#headerbody

Request(GET)

Header

GET http://www.naver.com/dust?lat=127&lng=93
User-Agent: Android; Host: www.naver.com

Request(POST)

Header

POST http://www.naver.com/dust
User-Agent: Android; Content-Type:application/form-urlencoded

Body

lat=127&lng=93

Response

Status : 200 ok
Content-Type: application/json Content-Length: 532
<pre>{ "lat":127, "lng":93, "weather":{ "temperature":16.3, "dust":5.3 } }</pre>

03 HTTP란?

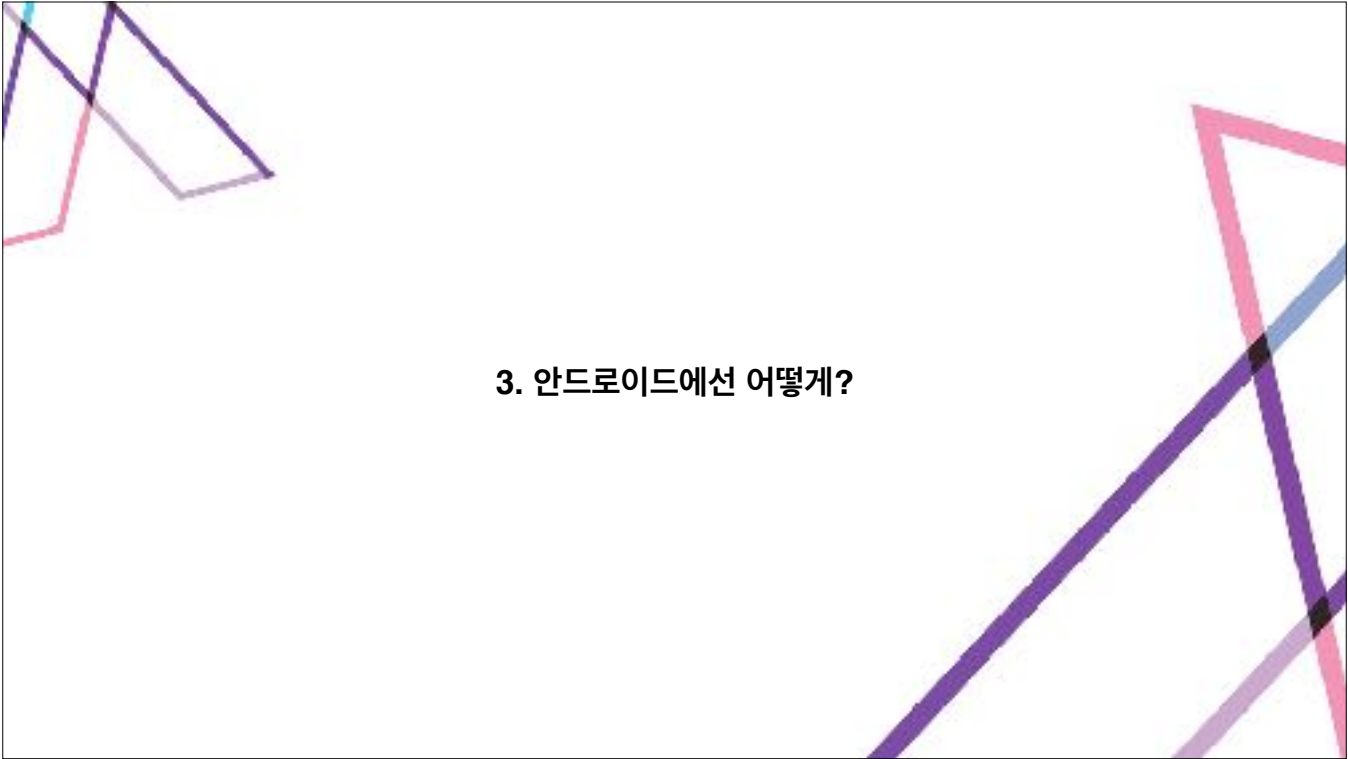
#statuscode

200	요청 성공
302	다른페이지로 이동
404	리소스를 찾을 수 없음
500	서버 내부 오류
504	게이트웨이 시간초과
...	

03 HTTP란?

#statuscode

성공 -	200	요청 성공
리다이렉트 -	302	다른페이지로 이동
클라이언트 실수 -	404	리소스를 찾을 수 없음
서버 실수 -	500	서버 내부 오류
	504	게이트웨이 시간초과
	...	

The image features several abstract, overlapping geometric lines in shades of purple, pink, and blue. These lines form various shapes, including triangles and polygons, primarily concentrated in the top-left and bottom-right corners of the frame. The central area is mostly white, providing a clear background for the text.

3. 안드로이드에선 어떻게?

04 안드로이드에선 어떻게?

HttpClient



Retrofit



Spring

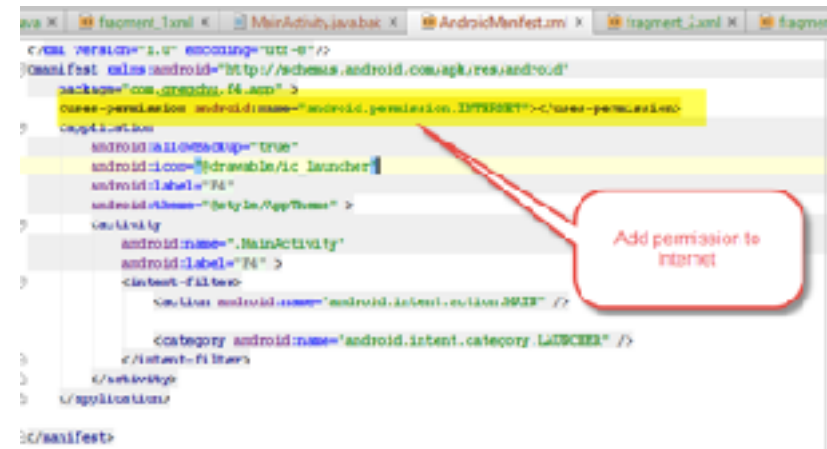
04 안드로이드에선 어떻게?



그리고 안드로이드에서 통신을 할때 주의해야할것들을 알아볼게요

04 안드로이드에선 어떻게?

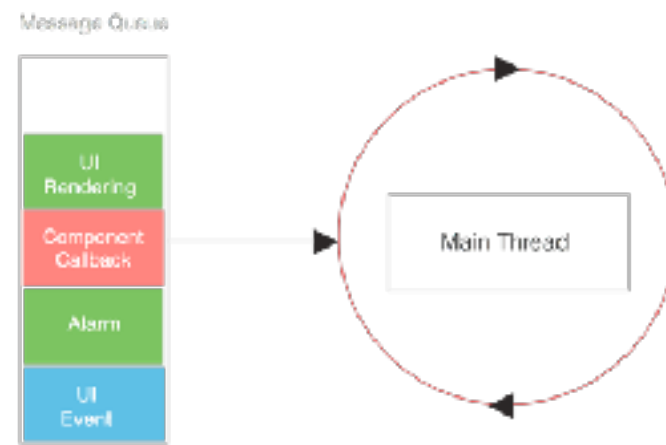
1. 권한(Permission)



```
1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3    package="com.greengrass.f4_app" >
4    <uses-permission android:name="android.permission.INTERNET" /></uses-permission>
5  </manifest>
6
7  <application
8    android:allowBackup="true"
9    android:icon="@drawable/ic_launcher"
10   android:label="@string/app_name"
11   android:theme="@style/AppTheme" >
12    <activity
13      android:name=".MainActivity"
14      android:label="@string/app_name" >
15      <intent-filter>
16        <action android:name="android.intent.action.MAIN" />
17        <category android:name="android.intent.category.LAUNCHER" />
18      </intent-filter>
19    </activity>
20  </application>
21 </manifest>
```

04 안드로이드에선 어떻게?

2. MainThread



04 안드로이드에선 어떻게?

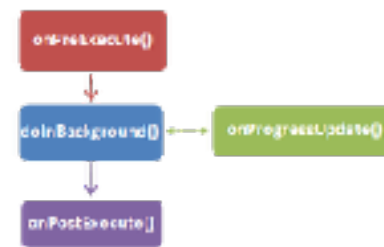


동기
synchronized



비동기
asynchronized

04 안드로이드에선 어떻게?



AsyncTask

Java8 Future

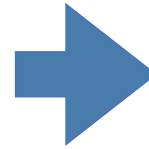


Thread + Handler

04 안드로이드에선 어떻게?

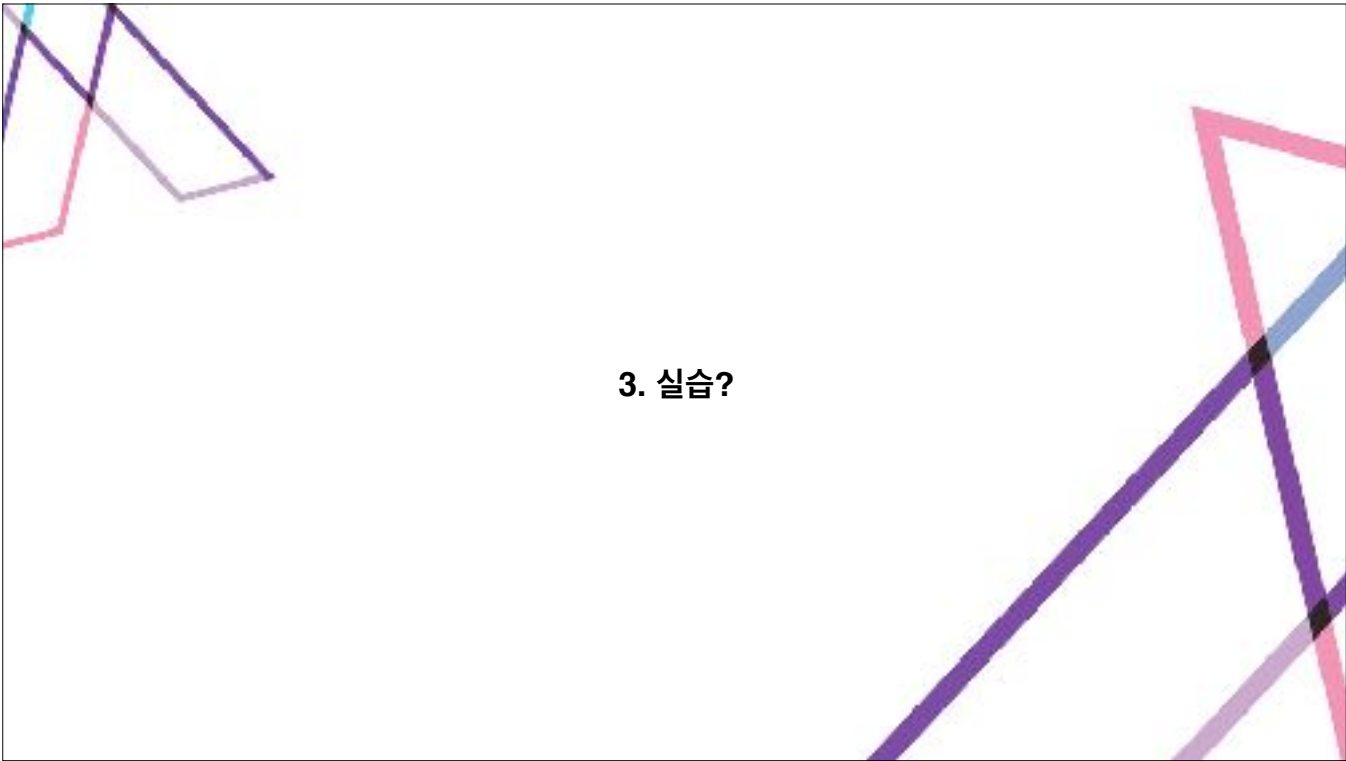
직렬화

```
{  
  "lat":127,  
  "lng":93,  
  "weather":{  
    "temperature":16.3,  
    "dust":5.3  
  }  
}
```



```
Class Weather {  
  public int lat;  
  public int lng;  
  public Weather weather;  
}
```

{ Gson }





그전에.. 좀 쉬어용~

실습1

Okhttp 맛보기

Request(GET)

Header

GET http://52.78.84.8:5000/api/users/{name}/friends
User-Agent: Android;

Response

Status : 200 ok
Content-Type: application/json Content-Length: 532
{ "friends": ["김정규", "임재은", "이동근", "신정아"], "name": "정성민" }

실습2

미세먼지앱 만들기

소스코드 : https://github.com/JSpiner/MASHUP_HTTP_STUDY

Request(GET)

Header

GET <http://52.78.84.8:5000/dust?lat={lat}&lng={lng}>

User-Agent: Android;

1. 미세먼지가 60 이상이면 '위험'
2. 미세먼지가 30 이상이면 '나쁨'
3. 미세먼지가 0 이상이면 '보통'
4. 서버는 10%확률로 에러가 납니다 ^^
5. 결과값은 랜덤인데 1분마다 바뀝니다.

Response

Status : 200 ok

Content-Type: application/json
Content-Length: 532

```
{
  "lat":127,
  "lng":93,
  "weather":{
    "temperature":16.3,
    "dust":5.3
  }
}
```

마무리