# Catatan belajar android

1. Agar Gradle Build Offline, gradle online akan sangat lambat saat proses percobaan running android di android studio. Gradle akan terus mengkoneksikan ke internet (do not know what it doing)

Buka *C:\Users\Wind\.gradle*

Buat file dengan nama *gradle.properties*

Isi :

*org.gradle.daemon=true*

1. Proses running emulator di android studio cukup cepat, emulator ini tidak perlu di matikan atau diclose pada saat running karena anda dapat melakukan compile ulang hanya dengna menekan tombol



1. Walkthrough :

* Membaca dan memahami konsep pemograman dan pola desain di android

Konsep yang perlu dipahami :

1. View di android
2. Memahami layout view dengan berbagai jenis layout dan view (relative, linear, GridView, ListView, ScrollView, [Recycler View](http://javatechig.com/android/android-recyclerview-example))

Referensi :

<http://developer.android.com/guide/topics/ui/declaring-layout.html>

good read :

<http://www.androidhive.info/2011/07/android-layouts-linear-layout-relative-layout-and-table-layout/>

1. Membuat Fragment di Android

<http://www.vogella.com/tutorials/AndroidDialogs/article.html>

<https://www.raywenderlich.com/117838/introduction-to-android-fragments-tutorial>

1. Memahami berbagai konsep dasar pemograman (activity, broadcast activity, service dsb)
2. Memahami event handling di android
3. Layout dan Styling di android
4. Simulasi aplikasi sederhana (app kontak address) dengan menggunakan sqlite.

Referensi :

<http://www.vogella.com/tutorials/AndroidSQLite/article.html>

1. Memproses Json dari server

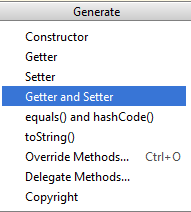
Memproses json dari server mesti menggunakan library **Gson** anda dapat mencari versi terbaru dengan keyword “Gradle gson package search” kemudian tambahkan paket terbaru tersebut di app.gradle

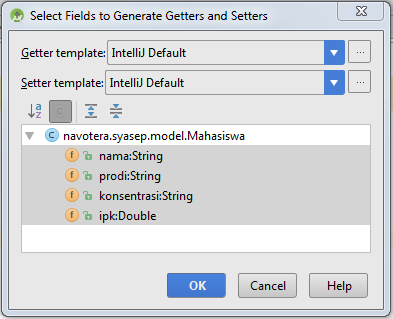
Untuk dapat memproses json yang merupakan tipe String dari server, maka anda harus membuat object terlebih dahulu. Misalnya mahasiswa.java, untuk membuat setter dan getter di android studio anda mesti menuliskan terlebih dahulu variable2nya misalnya

**public class** Mahasiswa {  
 **public** String **nama**;  
 **public** String **prodi**;  
 **public** String **konsentrasi**;  
 **public** Double **ipk**;

tinggal setter dan getter kan…

untuk membuat setter dan getter secara otomatis di android studio anda dapat menekan tombol **alt + insert** blok seluruh variable member kemudian **ok**.





Memproses json array ke array list object

<http://www.leveluplunch.com/java/examples/convert-json-array-to-arraylist-gson/>

Android is controller – view –adapter pattern

Controller dalam android dikenal dengan istilah Activity dimulai dari Main\_activity ditulis dengan Java. Main Activity akan bertindak sebagai PARENT ACTIVITY atau controller utama secara default. Untuk mengubahnya anda dapat mengubah pada file manifest :

*<?***xml version="1.0" encoding="utf-8"***?>*<**manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.example.wind.helloworld"**>  
  
 <**application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:supportsRtl="true"  
 android:theme="@style/AppTheme"**>  
 <**activity android:name=".MyActivity"**>  
 <**intent-filter**>  
 **<action android:name="android.intent.action.MAIN" />**  
  
 <**category android:name="android.intent.category.LAUNCHER"** />  
 </**intent-filter**>  
 </**activity**>  
  
  
 <**activity  
 android:name=".DisplayMessageActivity"  
 android:label="@string/title\_activity\_display\_message"  
 android:parentActivityName="com.example.wind.helloworld.MyActivity"** >  
 <**meta-data  
 android:name="android.support.PARENT\_ACTIVITY"  
 android:value="com.example.wind.helloworld.MyActivity"** />  
 </**activity**>  
  
 <**activity  
 android:name=".TestDatabaseActivity"  
 android:label="@string/title\_test\_database\_activity"  
 android:parentActivityName="com.example.wind.helloworld.MyActivity"** >  
 <**meta-data  
 android:name="android.support.PARENT\_ACTIVITY"  
 android:value="com.example.wind.helloworld.MyActivity"** />  
 </**activity**>  
  
  
 </**application**>

## Mengubah json agar bisa digunakan oleh aplikasi (menggunakan jsonUtil dan Gson)

1. Json sederhana

Format Json sebagai berikut :

|  |
| --- |
| {  "nama":"MUHAMMAD ELMI",  "nim":"A03090038" ,  "prodi":"D3 AKUNTANSI" ,  "konsentrasi" :"Akuntansi",  "ipk":"2.17"  } |

Mahasiswa mahasiswa = Jsonutil.*fromJsonToObj*(json, Mahasiswa.**class**);  
TextView ipk = (TextView)findViewById(R.id.***nilai\_ipk***);  
ipk.setText(Double.*toString*(mahasiswa.getIpk()));

1. Json Bentuk Array :
   1. [
   2. {
   3. "name":"Beagle",
   4. "colors":["black","white","tan"]
   5. },
   6. {
   7. "name":"Dalmation",
   8. "colors":["white","black"]
   9. }
   10. ]

Buat object terlebih dahulu

**class** Dog {

String name;

String[] colors;

}

Ingin mengubah menjadi object array :

JsonReader reader = **new** JsonReader(**new** StringReader(json));  
reader.setLenient(**true**);  
Gson gson = **new** Gson();

Dog[] dogs = gson.fromJson(reader, Dog[].**class**);

Ingin mengubah menjadi bentuk List<Object> :

List<PengumumanModel> mpengumumanList = **new** Gson().fromJson(reader, **new** TypeToken<List<PengumumanModel>>(){}.getType());

# Void di method return value ?

Seringkali kita memperoleh method default dari suatu library tanpa bias kita kembalikan nilainya contohnya

Memanggil api server dengan menggunakan library optimusHTTP.

**public void** getPengumuman() {  
 String serverParam = ***SERVER*** + **"pengumuman"**;  
 OptimusHTTP client = **new** OptimusHTTP();  
 client.enableDebugging();  
 client.setMethod(OptimusHTTP.***METHOD\_GET***);  
  
  
 ArrayMap<String, String> params = **new** ArrayMap<>();  
  
  
 ArrayList<HttpReq> refHttpReqList = **new** ArrayList<>();  
  
 client.makeRequest(**this**.**activity**, serverParam, params, **new** OptimusHTTP.ResponseListener() {  
 @Override  
 **public void** onSuccess(String json) {  
  
 JsonReader reader = **new** JsonReader(**new** StringReader(json));  
 reader.setLenient(**true**);  
 Gson gson = **new** Gson();  
 List<PengumumanModel> mpengumumanList = **new** Gson().fromJson(reader, **new** TypeToken<List<PengumumanModel>>() {  
 }.getType());  
  
   
  
 }  
  
 @Override  
 **public void** onFailure(String msg) {  
 System.***out***.println(msg);  
  
 }  
 });  
  
  
}

seperti yang dilihat pada hasil dari pemanggilan method makeRequest diatas hanya menyediakan onSuccess dan kita tidak akan bias mengembalikan nilai dari getPengumuman misalnya dengan menuliskan

**public String getPengumuman()**

kode diatas pasti akan error karena tidak hasil makeRequest tidak pasti (bias onSuccess atau onFailure), caranya adalah kita membuat interface yang dapat digunakan untuk callback

**public interface** HttpCallback {  
 **void** onSuccess(String result);  
}

------------------------------------

**public void** getPengumuman(**final** HttpCallback callback) {  
 String serverParam = ***SERVER*** + **"pengumuman"**;  
 OptimusHTTP client = **new** OptimusHTTP();  
 client.enableDebugging();  
 client.setMethod(OptimusHTTP.***METHOD\_GET***);  
  
  
 ArrayMap<String, String> params = **new** ArrayMap<>();  
 ArrayList<HttpReq> refHttpReqList = **new** ArrayList<>();  
  
 client.makeRequest(**this**.**activity**, serverParam, params, **new** OptimusHTTP.ResponseListener() {  
 @Override  
 **public void** onSuccess(String json) {  
  
 JsonReader reader = **new** JsonReader(**new** StringReader(json));  
 reader.setLenient(**true**);  
 Gson gson = **new** Gson();  
 List<PengumumanModel> mpengumumanList = **new** Gson().fromJson(reader, **new** TypeToken<List<PengumumanModel>>() {  
 }.getType());  
  
 *//interface* callback.onSuccess(json);  
  
 }  
  
 @Override  
 **public void** onFailure(String msg) {  
 System.***out***.println(msg);  
  
 }  
 });  
  
  
}

Cara memanggilnya di Activity adalah sebagai berikut :

Server httpClient = **new** Server(**this**);  
httpClient.getPengumuman(**new** VolleyCallback() {  
 @Override  
 **public void** onSuccess(String result) {  
 System.***out***.println(result);  
 System.***out***.println(**"contacting srv"**);  
 }  
});

## Membuat Activity baru

Untuk dapat membuat activity baru yang akan anda tampilkan maka anda harus melakukan langkah berikut :

1. Buat satu file *TestDatabaseActivity.java* pada folder **java->nama\_project\_anda**
2. Tambahkan atau register activity tersebut pada file manifest.xml

<**activity  
 android:name=".TestDatabaseActivity"  
 android:label="@string/title\_test\_database\_activity"  
 android:parentActivityName="com.example.wind.helloworld.MyActivity"** >  
 <**meta-data  
 android:name="android.support.PARENT\_ACTIVITY"  
 android:value="com.example.wind.helloworld.MyActivity"** />  
 </**activity**>

1. Anda dapat meredirect ke activity yang baru saja anda buat dengan cara menuliskan kode berikut (bias dengan redirect pada event onclick dan apa saja).



View diisi dengan bahasa XML pada folder res/layout/

Adapter merupakan penghubung Activity/Controller ke berbagai sumber data seperti lokal SQLite atau Restserver.

* Hello world dengan toast dan hello world dengan textview
* Buat Form dan simpan ke database
* Menggunakan data dari REST Server localhost (akses rest dengan IP local **bukan** localhost)
* Notifikasi dari GCM (Google Cloud Message)
* Menggunakan cross app seperti google map dan lain2
* Mencoba membuat rancangan app real.

1. **Pahami konsep dasar Synchronous vs. asynchronous in android…**

**Good article :**

1. [**http://code.hootsuite.com/asynchronous-android-programming-the-good-the-bad-and-the-ugly/**](http://code.hootsuite.com/asynchronous-android-programming-the-good-the-bad-and-the-ugly/)
2. [**https://realm.io/news/android-threading-background-tasks/**](https://realm.io/news/android-threading-background-tasks/)
3. [**https://developer.android.com/reference/android/os/AsyncTask.html**](https://developer.android.com/reference/android/os/AsyncTask.html)

Progmming sync cenderung terkait dengan konsep multithread, async dalam pemahaman yang singkat adalah pendekatan memanfaatkan proses thread yang lain. Misalnya UI menggunakan thread 1 maka dengan async logic pemograman berjalan dengan menggunakan thread 2, dalam prakteknya misalnya anda menuliskan script yang akan difungsikan mengkoneksikan dengan server online / REST Server. Maka agar tidak mengganggu aktivitas UI (tidak memfokuskan pada 1 thread proses saja, yang akan mengakibatkan aplikasi terkesan tidak responsive / hang) maka anda mesti menggunakan pendekatan async. Namun sync dapat juga dipertimbangkan (pendekatan ini menggunakan 1 thread saja sebagai proses utama yang menjalankan seluruh app logic). Pada prakteknya google memblokir segala proses yang akan membuat UI tidak responsive. Google menyediakan fitur *handler*  untuk memfasilitasi komunikasi async dengan UI

Pendekatan sync ini bermanfaat jika anda ingin memblokir aktivitas user. Misalnya anda membuat app yang menghubungi server untuk konfirmasi transaksi, maka untuk mencegah user mengubah data pembelian maka perlu dibuatkan blocking terhadap sehingga dapat menyelesaikan proses transaksi.

So kapan menggunakan sync dan async

Referensi konsep dasar penjelasan kode AsyncTask di Android : <http://code2concept.blogspot.co.id/2015/01/asynctask-android-example.html>

People on web always tell me that android is thread-safe… what the hell is that means :

<http://stackoverflow.com/questions/261683/what-is-meant-by-thread-safe-code>

im getting stuck when trying dismiss dialog on different thread here the reference for answer :

<https://www.google.com/search?q=dismiss+dialog+on+main+thread+android&ie=utf-8&oe=utf-8&client=firefox-b-ab>

# [Is main thread the same as UI thread?](http://stackoverflow.com/questions/3261370/is-main-thread-the-same-as-ui-thread)

<http://stackoverflow.com/questions/3261370/is-main-thread-the-same-as-ui-thread>

|  |  |
| --- | --- |
|  | **UI Thread** and **Main Thread** are same only in **Android**.  The **Main thread**, that is responsible for handling the **UI events** like Draw, Listen and receive the **UI events**.  Ans also it is responsible for interact with running **components** of the **UI toolkit** for the corresponding application that belongs to.  When an User event occurs in the application, the **Main thread** \*  need to add the event in the queue -> intimate about the event to appropriate View -> change the state of the view -> redraw the view according to the state changes -> waiting for the response for the particular event action -> after intimated and event action completed need to delete the event in the queue.  \*  The above every actions are handled by the **Main thread** (Not only the above operation, it is a one of the operation handled by the **UI Thread**), So if our application fails to respond the event about **5 seconds** android will shows the error **"not responding"**.  So only it is widely suggested to do the light processes in the **UI thread**.  Hope this answer is somewhat detail and helpful to the new android bees like me. I just shared what i learned about UI Thread. If i went wrong in anywhere please don't hesitate to recorrect me. |

# [Handler vs AsyncTask vs Thread What the difference ?](http://stackoverflow.com/questions/6964011/handler-vs-asynctask-vs-thread?rq=1)

<http://stackoverflow.com/questions/6964011/handler-vs-asynctask-vs-thread?rq=1>

|  |
| --- |
| If you look at the source code of AsyncTask and Handler, you will see their code is written purely in Java. (Of course, there are some exceptions, but that is not an important point.)  So there is no magic in AsyncTask or Handler. They just make your job easier as a developer.  For example: If Program A calls method A(), method A() could run in a different thread with Program A.You can easily verify it using:  Thread t = Thread.currentThread();  int id = t.getId();  Why you should use a new thread? You can google for it. Many many reasons.  So, what is the difference between Thread, AsyncTask, and Handler?  AsyncTask and Handler are written in Java (internally they use a Thread), so everything you can do with Handler or AsyncTask, you can achieve using a Thread too.  What can Handler and AsyncTask really help you with?  The most obvious reason is communication between the caller thread and the worker thread. (*Caller Thread*: A thread which calls the *Worker Thread* to perform some task. A Caller Thread does not necessarily have to be the UI thread). Of course, you can communicate between two threads in other ways, but there are many disadvantages (and dangers) due to thread safety issues.  That is why you should use Handler and AsyncTask. They do most of the work for you, you just need to know what methods to override.  The difference between Handler and AsyncTask is: Use AsyncTask when *Caller thread* is a *UI Thread*. This is what android document says:  AsyncTask enables proper and easy use of the UI thread. This class allows to perform background operations and publish results on the UI thread without having to manipulate threads and/or handlers  I want to emphasize on two points:  1) Easy use of the UI thread (so, use when caller thread is UI Thread).  2) No need to manipulate handlers. (means: You can use Handler instead of AsyncTask, but AsyncTask is an easier option).  There are many things in this post I haven't said yet, for example: what is UI Thread, or why it's easier. You must know some method behind each kind and use it, you will completely understand why..  @: when you read the Android document, you will see:  Handler allows you to send and process Message and Runnable objects associated with a thread's MessageQueue  They may seem strange at first. Just understand that each thread has each message queue (like a to do list), and the thread will take each message and do it until the message queue is empty (just like you finish your work and go to bed). So, when Handler communicates, it just gives a message to caller thread and it will wait to process. Complicated? Just remember that Handler can communicate with the caller thread in a safe way. |

This tutorial about synctask, handler and runnable thread is great : <http://www.101apps.co.za/articles/using-threads-tutorial.html>

1. Kenapa harus terpisah antara string.xml dan layout.xml

Sebenarnya bias saja tidak dipisahkan misalnya text pada tombol di android dituliskan langsung pada parameter android:text = “your text” tapi kesulitannya adalah apabila anda melakukan pengembangan android dengan view yang beragam misalnya ada yang khusus untuk 5 inch, 4 inch, 4.5 inch, tablet, desktop sehingga aka nada banyak layout.xml yang anda miliki sebanyak ragam kekhususan ukuran layout. Akan sangat merepotkan apabila tiap layout.xml yang banyak tersebut anda harus mengedit satu2. Dengan adanya strings.xml cukup satu parameter yang anda edit.

1. You know design a template is exhausting, rumit dan melelahkan (ini karena kita kurang terbiasa dalam mendesain dan belum tahu best practice dalam desain) ….. kabar baiknya kita bisa download ready template xml di situs berikut (jika kita buru2 tentu saja):

* <https://github.com/andreasschrade/android-design-template>
* <http://wsdesign.in/>

1. Mengkoneksikan dan memperoleh data dari REST Server :
   1. Anda harus menambahkan library / repositories dari maven karena script ini akan menggunakan library OptimusHTTP yang tersedia di mavenCentral. Menambahkan maven sebagai sumber respositories telah dijelaskan pada tips yang dituliskan pada bagian dari tutorial ini.
   2. Script adalah sebagai berikut :

**public void** connectREST(View view)  
{  
 String SERVER = **"http://jsonplaceholder.typicode.com/posts/1"**;  
 OptimusHTTP client = **new** OptimusHTTP();  
 client.enableDebugging();  
 client.setMethod(OptimusHTTP.***METHOD\_GET***);  
  
  
 *//parameter* ArrayMap<String, String> params = **new** ArrayMap<>();  
 params.put(**"email"**, **"abc@abc.com"**);  
 params.put(**"pass"**, **"abc"**);  
  
  
  
 *//make request* ArrayList<HttpReq> refHttpReqList = **new** ArrayList<>();  
 **try** {  
 *// makeRequest() returns the reference of the request made  
 // which can be used later to call the cancelReq() if required  
 // if no request is made the makeRequest() returns null* HttpReq req = client.makeRequest(MainActivity.**this**, SERVER, params, **responseListener**);  
 **if** (req != **null**)  
 refHttpReqList.add(req);  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 }  
  
}  
  
**private final** OptimusHTTP.ResponseListener **responseListener** = **new** OptimusHTTP.ResponseListener() {  
 @Override  
 **public void** onSuccess(String msg) {  
 System.***out***.println(msg);  
 Toast.*makeText*(getApplicationContext(), msg, Toast.***LENGTH\_LONG***).show();  
 }  
  
 @Override  
 **public void** onFailure(String msg) {  
 System.***out***.println(msg);  
 }  
};

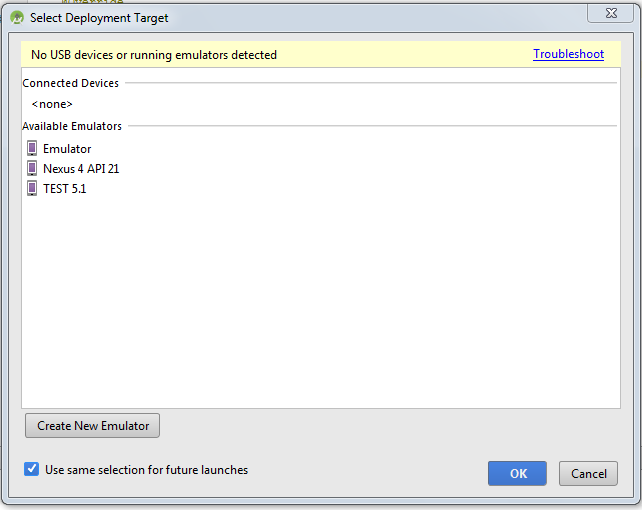
Memilih library http di android

Good reference : <https://packetzoom.com/blog/which-android-http-library-to-use.html>

**TroubleShoot :**

1. Emulator Android tidak menampilkan daftar emulator : *Nothing to show.*

|  |
| --- |
|  |

Saat Memulai emulator anda akan ditampilkan dialog berikut: 

Daftar emulator yang sudah anda setting sebelumnya

Namun saat tertentu anda tidak akan ditampilkan daftar emulator, kenapa ? Bug….

Anda hanya akan ditampilkan “*Nothing to show”* dengan log Unable to Get ADB Version.

**Solusinya** adalah dengan

* PC anda harus online internet (Pada poin ini belum ditemukan cara offline untuk menjalankan emulator)
* Coba lagi, ada kemungkinan ADB tidak jalan
* Merestart kembali android studio atau langkah terakhir adalah menginstall ulang android studio.

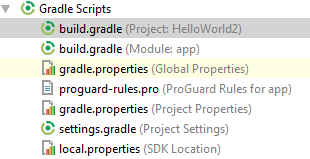
1. Onclick event error dengan pesan android “Unfortunately, [APP NAME] has stopped” . ini terjadi karena untuk meregister Onclick event di view harus diletakkan dibawah.. berikut contoh :

<**Button  
 style="?android:attr/buttonStyleSmall"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Test Toast"  
 android:id="@+id/button"  
 android:layout\_below="@+id/sql\_lite\_menu"  
 android:layout\_alignRight="@+id/restful\_menu"  
 android:layout\_alignEnd="@+id/restful\_menu"  
 android:backgroundTint="#954c4c"  
 android:onClick="toastAlert"**/>

**(NOT TESTED, TRY IT)**

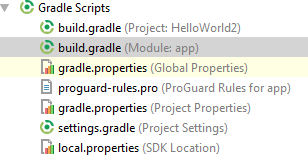
1. Menambahkan library dari Maven

Setting di file build.gradle (Project : Your Project Name)



buildscript {  
 repositories {  
 jcenter()  
 maven { url **'http://repo1.maven.org/maven2'** }  
 }  
 dependencies {  
 classpath **'com.android.tools.build:gradle:2.1.0'** *// NOTE: Do not place your application dependencies here; they belong  
 // in the individual module build.gradle files* }  
}  
  
allprojects {  
 repositories {  
 jcenter()  
 mavenCentral()  
 }  
}  
  
task clean(**type**: Delete) {  
 delete rootProject.buildDir  
}

dan pada build.gradle (Module:app)



apply **plugin**: **'com.android.application'**android {  
 compileSdkVersion 23  
 buildToolsVersion **"23.0.3"** defaultConfig {  
 applicationId **"com.example.wind.helloworld"** minSdkVersion 15  
 targetSdkVersion 23  
 versionCode 1  
 versionName **"1.0"** }  
 buildTypes {  
 release {  
 minifyEnabled **false** proguardFiles getDefaultProguardFile(**'proguard-android.txt'**), **'proguard-rules.pro'** }  
 }  
}  
  
dependencies {  
 compile fileTree(**dir**: **'libs'**, **include**: [**'\*.jar'**])  
 testCompile **'junit:junit:4.12'** compile **'com.android.support:appcompat-v7:23.3.0'** compile **'com.android.support:design:23.1.1'** compile **'com.github.nisrulz:optimushttp:1.0.0'**}

**Pada bagian ini**

**menambahkan paketnya;**

**Referensi :**

<http://stackoverflow.com/questions/16671277/maven-dependencies-with-android-studio-gradle>

**Konsep dasar :**

1. Apa itu context ?

* <http://stackoverflow.com/questions/3572463/what-is-context-on-android>
* <http://stackoverflow.com/questions/37384156/why-is-context-needed-here/37384265?noredirect=1#comment62279114_37384265>

1. Toast atau alert pesan:

* Toast.makeText(); hanya akan berjalan pada class activity atau turunannya.
* Cara menampilkan apabila diketikkan pada class activity adalah sebagai berikut:
* Toast.makeText(this, “message”, Toast.***LENGTH\_LONG***).show();

Atau

Toast.makeText(getApplicationContext(),”message”,Toast.***LENGTH\_LONG***).show();

1. Seringkali kita ingin melihat pesan dalam sistem (tidak ditampilkan di UI), misalnya ingin melihat status koneksi dengan server (ingat di log sistem bukan di UI android) maka dapat dilakukan dengan menggunakan pendekatan java biasanya dengan menuliskan

//shortcut yakni sout

System.***out***.println(string\_pesan)

Dimana kita dapat melihat isi pesan ini yang telah anda tuliskan, anda dapat melihat hasil sout tersebut di tab logcat dibawah ini:

|  |
| --- |
|  |

# ARCHICTECTURAL PATTERN IN ANDROID DEVELOPING

<http://www.tinmegali.com/en/model-view-presenter-android-part-1/>

# Java Corner

**What do the 3 dots in the following method mean?**

public void myMethod(String... strings){

// method body

}

**ANSWER**

|  |
| --- |
| It means that zero or more String objects (or an array of them) may be passed as the parameter(s) for that function.  See the "Arbitrary Number of Arguments" section here: <http://java.sun.com/docs/books/tutorial/java/javaOO/arguments.html#varargs>  In your example, you could call it as any of the following:  myMethod(); // Likely useless, but possible  myMethod("one", "two", "three");  myMethod("solo");  myMethod(new String[]{"a", "b", "c"}); |

Ref : <http://stackoverflow.com/questions/3158730/java-3-dots-in-parameters>

**What does <> in some parameter means ?**

Cannot answer it please read here : <http://www.tutorialspoint.com/java/java_generics.htm>

**Best practice resume in android development :**

[**https://github.com/futurice/android-best-practices**](https://github.com/futurice/android-best-practices)

**example :** [**https://github.com/ajitsing/ExpenseManager**](https://github.com/ajitsing/ExpenseManager)

it means generic type, in other programming language there are not any restricted type on method parameter for example so u can put any type on method param. In java there is that restriction so it would be difficult to put non exact type on some method, generic type is the solution for that problem. Read reference for more.

**What is LIST <> in java ?**

[**http://tutorials.jenkov.com/java-collections/list.html**](http://tutorials.jenkov.com/java-collections/list.html)

**Membuat class java agar dapat menggunakan konteks app yang sedang dijalankan oleh android :**

Seringkali kita ingin memisahkan antara class tertentu yang memiliki fungsi atau object tertentu misalnya database klass khusus. Maka kesulitannya adalah perlu kita mendefinisikan klass activity tersebut agar dapat digunakan oleh klass object khusus. Caranya begini :

<http://stackoverflow.com/questions/30746141/using-getapplicationcontext-in-another-simple-class-not-in-mainactivity-java>