مقدمة عن نظام الأندرويد

التفاصيل كتب بواسطة: AMANSAG الزيارات: 923

أندرويد هو نظام مجاني ومفتوح المصدر مبني على نواة لينكس صمم أساسًا للأجهزة ذات شاشات اللمس كالهواتف الذكية والحواسب اللوحية, يتم تطوير الأندرويد من قبل التحالف المفتوح للهواتف النقالة الذي تديره شركة جوجل.

اشترت جوجل نظام أندرويد من المطورين الأساسيين للنظام في سنة 2005، في حين كان الإعلان الرسمي عنه في 5 نوفمبر 2007 بالتزامن مع انطلاق التحالف المفتوح للهواتف النقالة.

طرحت جوجل نظام أندرويد كنظام مفتوح المصدر تحت رخصة أباتشي، مشروع أندرويد مفتوح المصدر (AOSP) هو الفريق المسؤول عن تطوير وتحديث وإصلاح النظام.

لدى الأندرويد مجتمع ضخم من المطورين الذين يقومون بكتابة وتطوير البرامج والتطبيقات للأندرويد، ويعتمدون بشكل أساسي على الكتابة بلغة الجافا.

جوجل بلاي وهو متجر متاح علي الإنترنت وكتطبيق على هواتف أندرويد يدار بواسطة جوجل. في سبتمبر 2012، وصل عدد التطبيقات المتواجدة بالمتجر 675,000 تطبيق وعدد التحميلات 25 مليار تحميل

يعد الأندرويد حاليًا النظام الأكثر انتشاراً بـ 500 مليون جهاز مفعل ونسبة مشاركة بالسوق بنسبة 62% والأسرع انتشارًا ب1.3 مليون هاتف جديد يفعل كل يوم

كان هذا تعريفا وجيزا من ويكيبيديا لنظام الندرويد الذي اكتسح مجال الهواتف الذكية.

في ما سيأتي ,سأحول أن أقدم دروسا لبرمجة تطبيقات الأندرويد،دروسا متسلسلة بدءا بتثبيت البرامج الأساسية مرورا بشرح لأهم عناصر بيئة العمل و فتح أول مشروع لنا ثم نمر بعدها إلى سلسلة من الدروس نشرح فيها كيفية استعمال مكونات نظام الأندرويد،لكن قبل هذا و ذاك لابد من الإشارة إلى أنه لابد من معرفة أساسيات لغة الجافا التي ستكون لغة البرمجة التي سنبرمج بها تطبيقاتنا.

أترككم مع بقية الدروس.

تحميل و تثبيت البرامج اللازمة

التفاصيل كتب بواسطة: AMANSAG الزيارات: 302

السلام عليكم و رحمة الله .

في فقرتنا هذه ، سنتطرق إلى كل الإعدادات التي لابد منها لبدء برمجة تطبيقات الأندرويد .

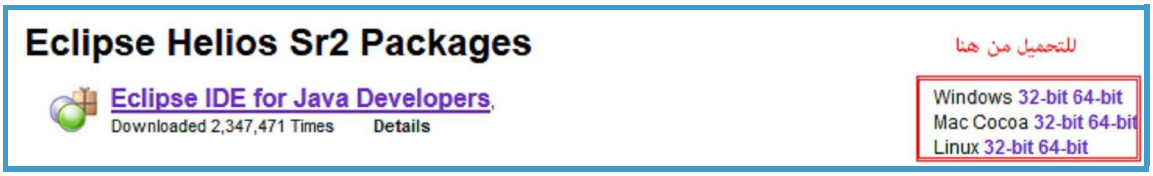
-

1- التحميل :

لأن java هي اللغة التي سنستعملها للبرمجة ، لابد من إعداد جهازنا لأجل ذلك ، و بالتالي كان لزاما تثبيت العناصر التالية :

- JRE أو Java Runtime Environement الضرورية لتشغيل أي تطبيق من تطبيقات الجافا.
- JDK أو Java Development Kit الضرورية لترجمة الأكواد إلى البيتكود ... يمكن تحميل المنصتين من هنا .

- Eclipse وهو محرر البرمجة الذي سنعمل به وستجدونه من هنا(Eclipse IDE for Java Developers).



2-التثبيت :

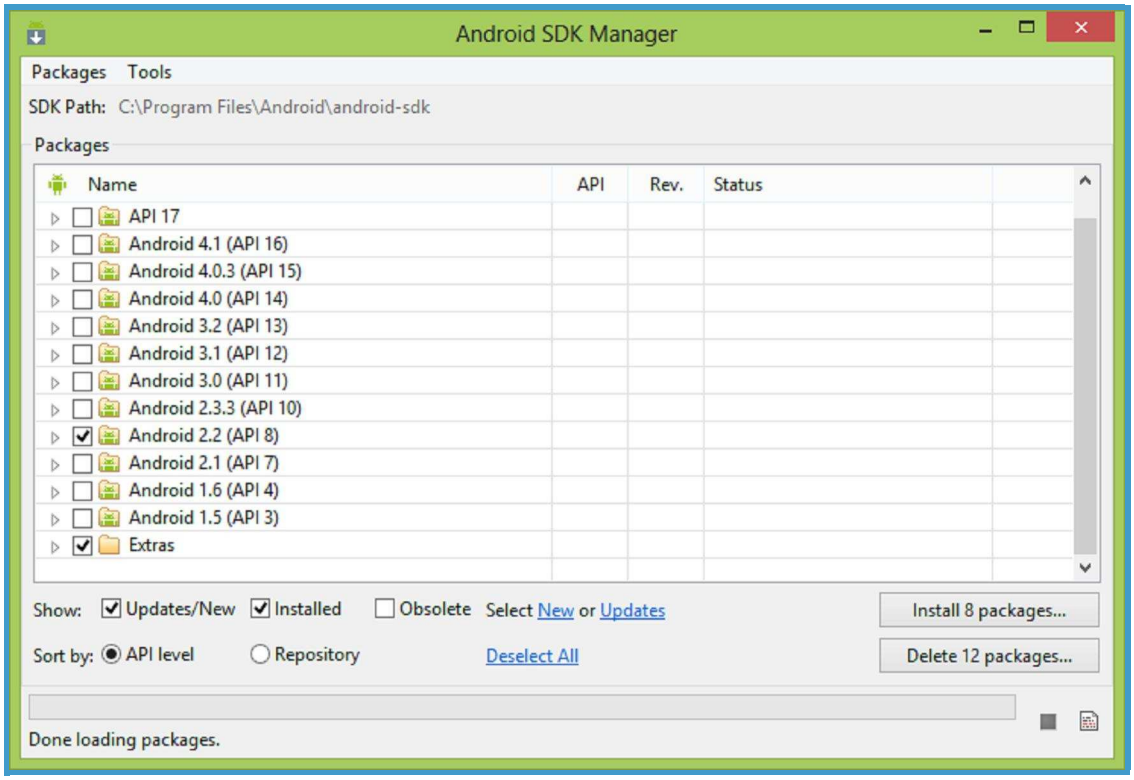
بالنسبة للمنصتين يكفي تتبع الخطوات وسيتم التثبيت بشكل تلقائي.أما بالنسبة لبيئة البرمجةب برنامج Eclipse فيكفي استخراج المجلد من الملف المضغوط ووضعه في المكان الذي تريده،كما يمكن إنشاء مختصر للملف Eclipse.exe على سطح مكتبك لكي يسهل الوصول إليه.

3- تحميل Android SDK :

الـ SDK هو مجموعة من الأدوات التي تمكن المبرمجين من برمجة تطبيقات خاصة بمنصة معينة ، بالتالي ف Android SDK هو مجموعة من الأدوات التي تمكن من برمجة تطبيقات خاصة بالأندرويد فقط.

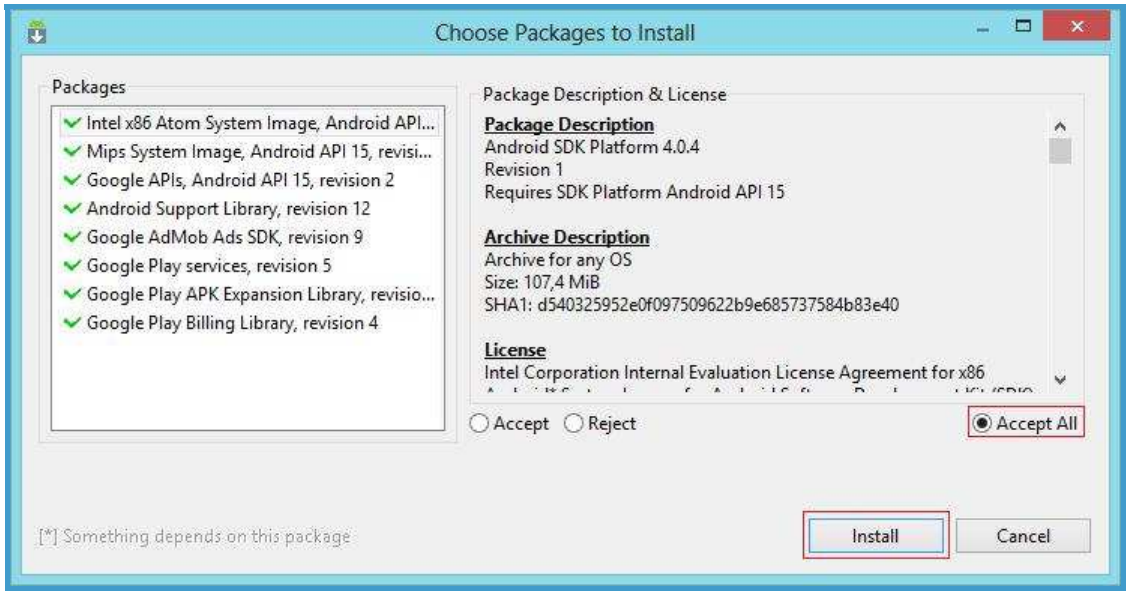
للحميل يمكنكم الذهاب إلى هذه الصفحة ثم النقر على USE AN EXISTING IDE ثم DOWNLOAD THE SDK TOOLS . بعد التحميل تقومون بالتثبيت .

عند التشغيل سترون شاشة على هذا الشكل :



هنا يمكنكم إختيار الحزمات التي تريدون تحميلها ،كل حزمة خاصة بنسخة من نسخ الأندرويد،وكل نسخة تمتاز بإضافات عن سابقاتها.في البداية يمكنكم اختيار الحزمتين Android 2.2 و Extras .
لكن إن كنا نريد أن نبرمج تطبيقا لأجهزة تشتغل تحت سخة متقدمة لابد من تحميل النسخ المناسبة.

ننقر على الزر Install ...packages وستظهر لنا نافذة أخرى :

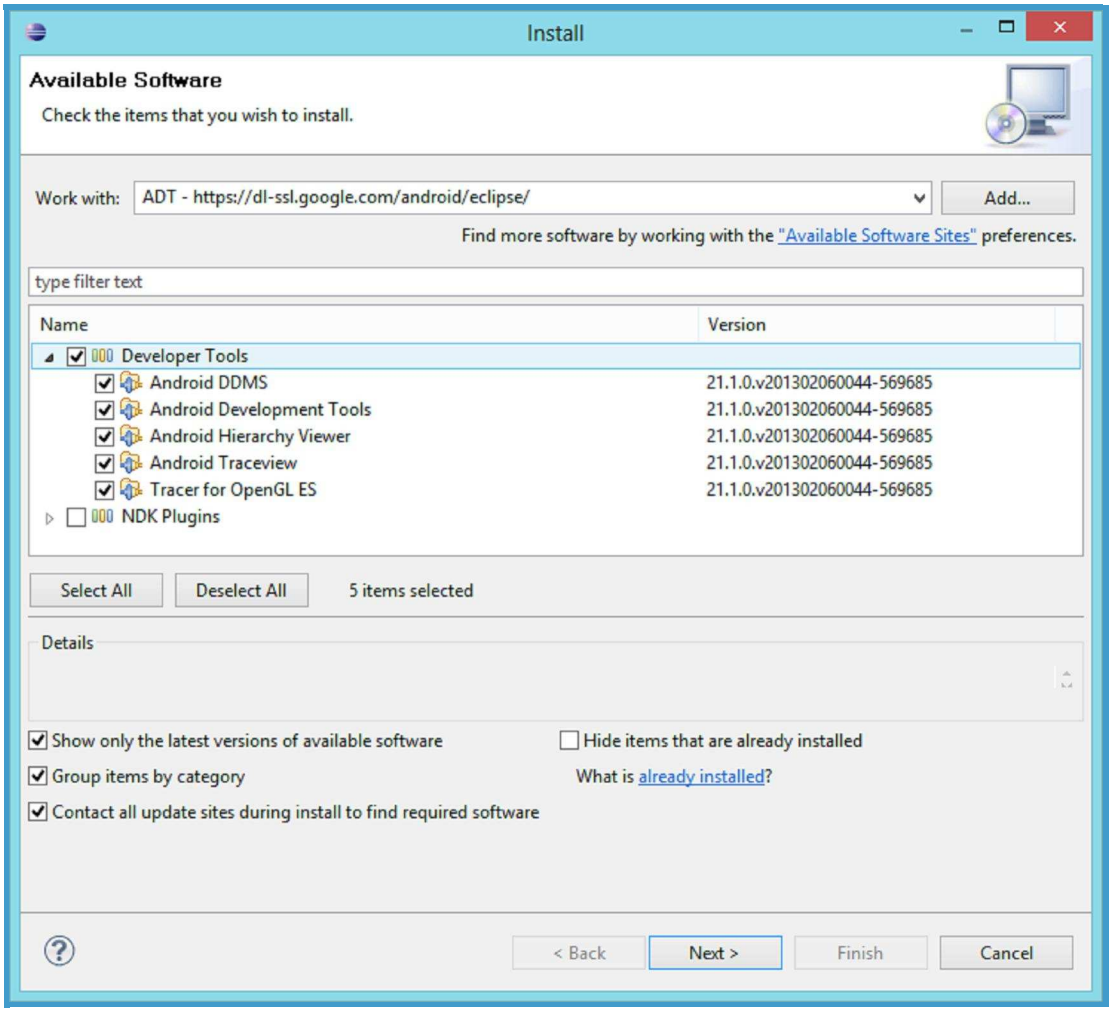


نختار Accept All ثم ننقر على Install .

تنقص مرحلة أخيرة لإتمام الإعدادات وهي:

4- تثبيت ADT :

من المعلوم أن Eclipse ليس معدا تمما لبرمجة تطبيقات الأندرويد و إنما لبرمجة تطبيقات الجافا،لكن هناك برامج مساعدة(Add-on) تمكنه من بينها ADT.من أجل ذلك نقوم أولا بتشغيل Eclipse ثم نذهب إلى القائمة Help ثم Install new softwares و ستظهر نافذة ننقر على الزر Add ثم تظهر نافذة صغيرة نكتب عليها الرابط التالي الذي سيتم التحميل منه `https://dl-ssl.google.com/android/eclipse/` ثم ننقر على OK، ننتظر قليلا ريثما يتم إيجاد الأدوات ...



نختار Developer Tools ثم ننقر على Next إى أن يكتمل التحميل .

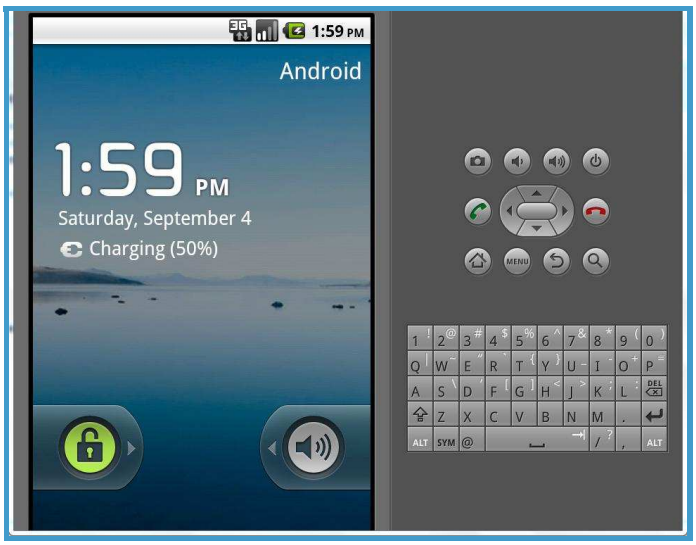
ربما ستواجهون بعض الأخطاء لكن يمكنكم المحاولة بكتابة الرابط السابق كما يلي `http://dl-ssl.google.com/android//eclipse` (كتابة http دون s).

التحميل و التثبيت سيتكلف بها جهازكم .

هنا تنتهي فقرة الإعدادات التي لابد من المرور منها لبرمجة تطبيقات الأندرويد.

محاكى الأندرويد

التفاصيل كتب بواسطة: Ahmed Amansag الزيارات: 195



السلام عليكم و رحمة الله و بركاته.

محاكي الاندرويد : Android Virtual Device

هو برنامج يقوم بمحاكاة الهواتف االتي تعمل تحت نظام الأندرويد أو برنامج الاندرويد للكمبيوتر، هذا يعني كله أنك لن تحتاج إلى هاتف أو لوح إلكتروني لكي تبرمج تطبيقا ما،لكن هذا لا

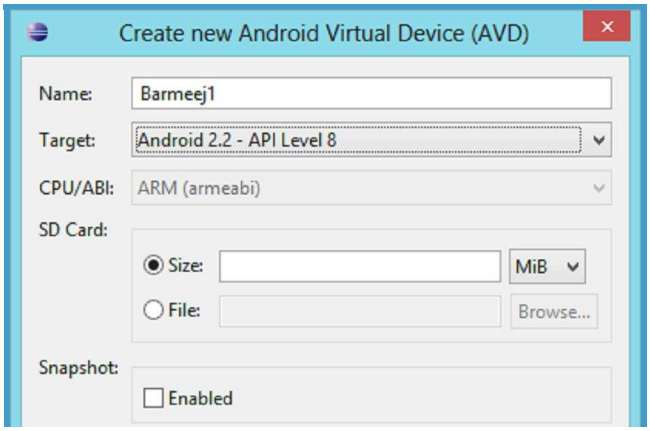
يعني بتاتا أنه يمكن الإستغناء عن الهواتف او الألواح ، فهناك تطبيقات لايكفي تشغيلها على المحاكي,كتطبيقات الـ GPS أو التطبيقات التي تستخدم حساسات الحرارة أو السرعة أو المجال المغناطيسي للهواتف.أما بالنسبة للذين يمتلكون هاتفا أولوحا فيمكنهم تجريب تطبيقاتهم عليها فهي تمتاز بالسرعة عكس المحاكي الذي سنجد انه بطيئ جداااا .

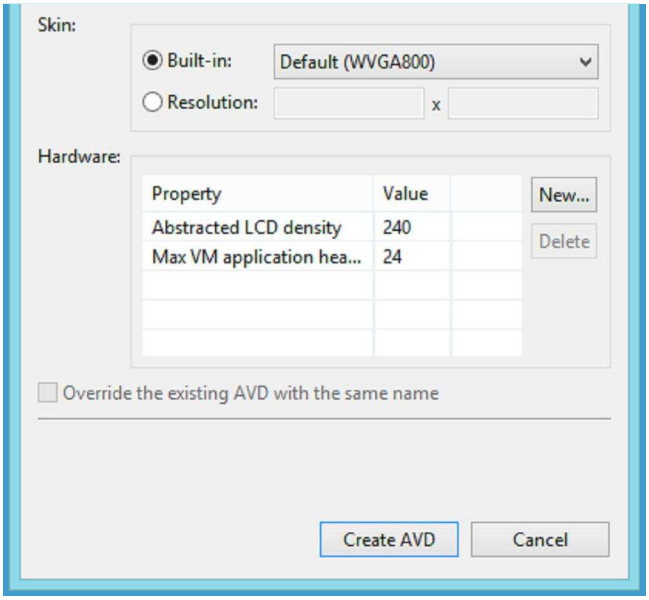
نفتح برنامج Eclipse و ستكونون قد لاحظتم أنه تم إضافة زرين خاصين بالأندرويد في شريط الأدوات هما كالتالي:

خاص بفتح الـ Android SDK لتحديث أو إضافة حزمات...

خاص بإعدادت المحاكيات (لأن لكل نسخة أندرويد محاكي الاندرويد الخاص بها).

ننقر على الزر الأخير الخاص بالمحاكي ستظهر لنا نافذة إعدادات المحاكيات ، سنضيف محاكي الاندرويد بالنقر على الزر New وبعدها تظهر نافذة لإضافة المحاكي





نملأ Name بما نشاء ثم Target بالنسخة المستهدفة (نسخة الأندرويد) ثم ننقر على الزر Create AVD .

يمكنكم تشغيل محاكي الاندرويد بالنقر على الزر start ثم الزر Launch ثم .....تنتظرون قليلا و ستتمكنون من التجوال في المحاكي .

ستلاحظون أن محاكي الاندرويد ثقيل جداا،لذلك ينصح عدم فتحه و إغلاقه في كل مرة سنجرب فيها تطبيقنا أي نفتحه مرة واحدة ولا نغلقه و مع مرور الوقت ستزداد سرعته .

مفاهيم أساسية قبل البدء

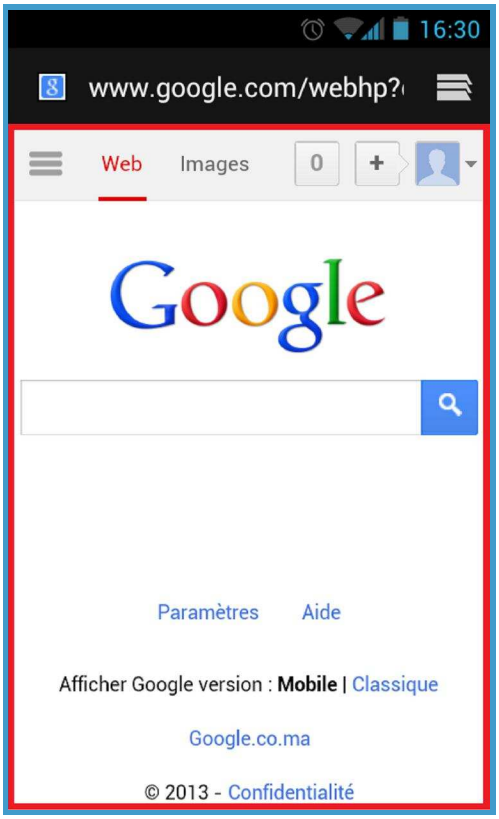
التفاصيل كتب بواسطة: Ahmed Amansag الزيارات: 226

السلام عليكم ورحمة الله و بركاته.

قبل أن نهاجم لوحة المفاتيح و نمر إلى البرمجة ، لابد من تسليط الأضواء على بعض العناصر المهمة في أي تطبيق من تطبيقات الأندرويد و كذا دورة حياة تطبيق أندرويد.

مفهوم الـ Activity :

الـ Activity هي ذلك المركب الأساسي لكل تطبيقات الأندرويد ، والتي تحتوي على الأقل على مكون واحد هو ال View (حيث يمكن أن يكون زرا أو حقل إدخال أو ....) لو لاحظنا كمثال هذا التطبيق :

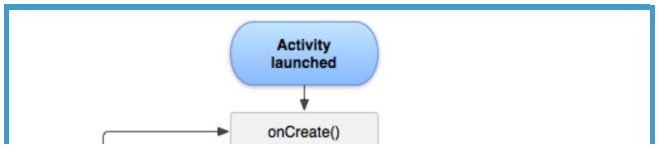


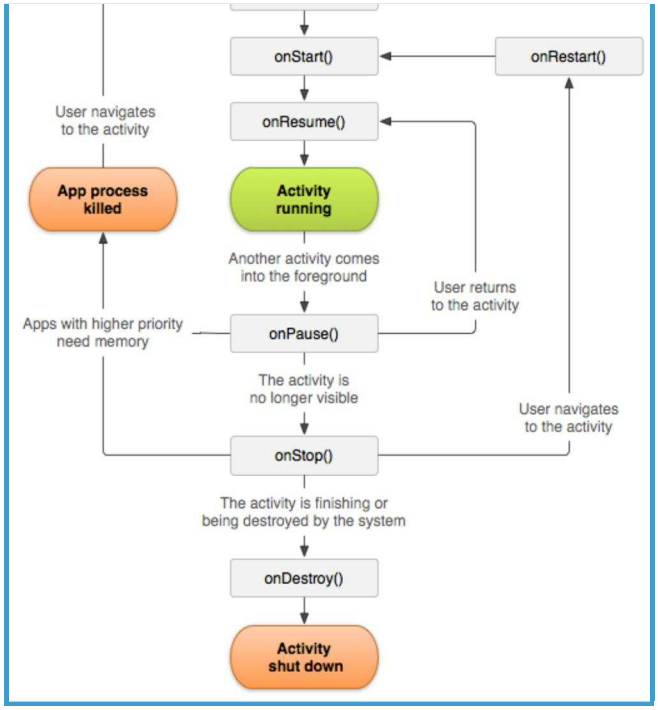
الإطار الأحمر هو حدود الـ Activity و كما ترون فهي تحتوي على عدة مكونات (View) مرتبة بطريقة معينة(سنرى ذلك في درس قادم).

كل تطبيق أندرويد يمكن أن يضم أكثر من Activity يمكن الإنتقال بينها و سنرى لاحقا ذلك في درس خاص.

دورة حياة ال Activity :

لكل Activity دورة حياة لابد من المرور منها منذ إنشائها إلى تدميرها،ويمكن تلخيص دورة حياة تطبيق اندرويد أو بالأحرى الActivity في المخطط التالي:





لدى الـActivity ثلاث حالات في دورة الحياة :

1. أن تكون في حالة اشتغال (نشيطة) Running وهو عندما تكون في الواجهة.
2. أن تكون في حالة توقف(معلقة) Paused ويكون ذلك عندما تفقد ما نسميه focus أي لا تبقى في الواجهة لكنها مازالت تعمل.
3. أن تكون في حالة إيقاف Stopped ويكون ذلك عندما يتم حجبها من طرف Activity أخرى .لكنها مازالت تحتفظ بكل المعلومات غير أنها لم تعد مرئية للمستخدم.وعندما سيكون هناك خصاص في الذاكرة سيتم تدميرها من طرف النظام.

بالمناسبة جميع الـ Activities ترث من الفئة (Class) التالية : Activity

مفهوم ال View :

نقصد به كل المكونات التي نصادفها عادة في تطبيقات الأندرويد من أزرار (Button) وحقول أدخال(EditText) و غيرها... وكلها ترث من الفئة التالية : View.

والآن سنمر إلى كيفية إنشاء مشروع أندرويد .

أول مشروع أندرويد

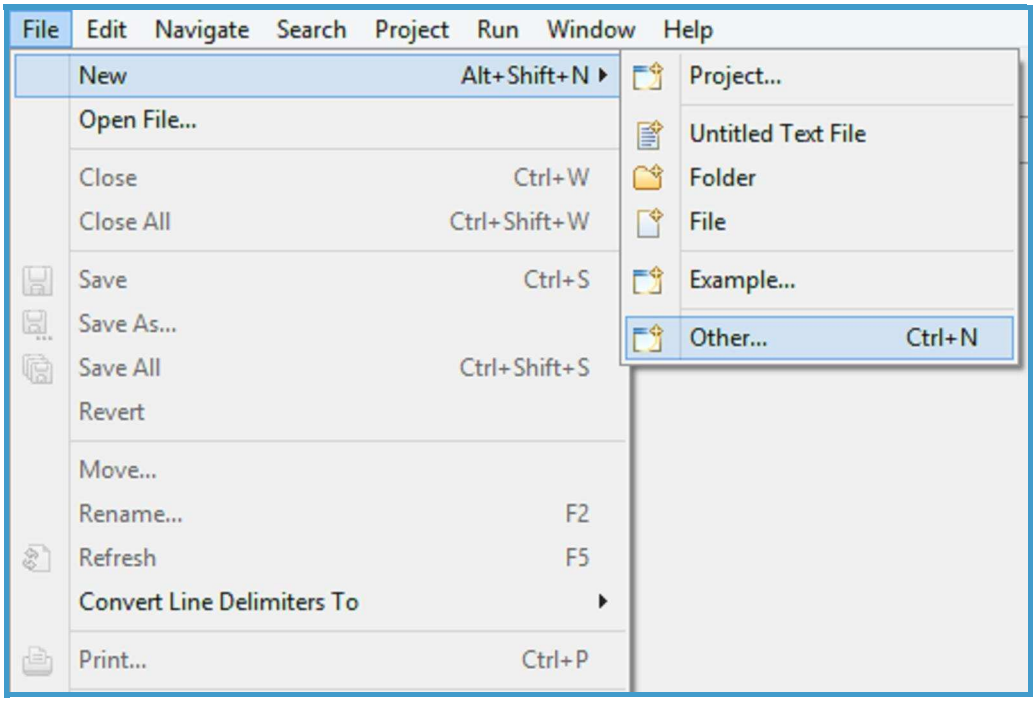
التفاصيل كتب بواسطة: Ahmed Amansag الزيارات: 1141

السلام عليكم و رحمة الله.

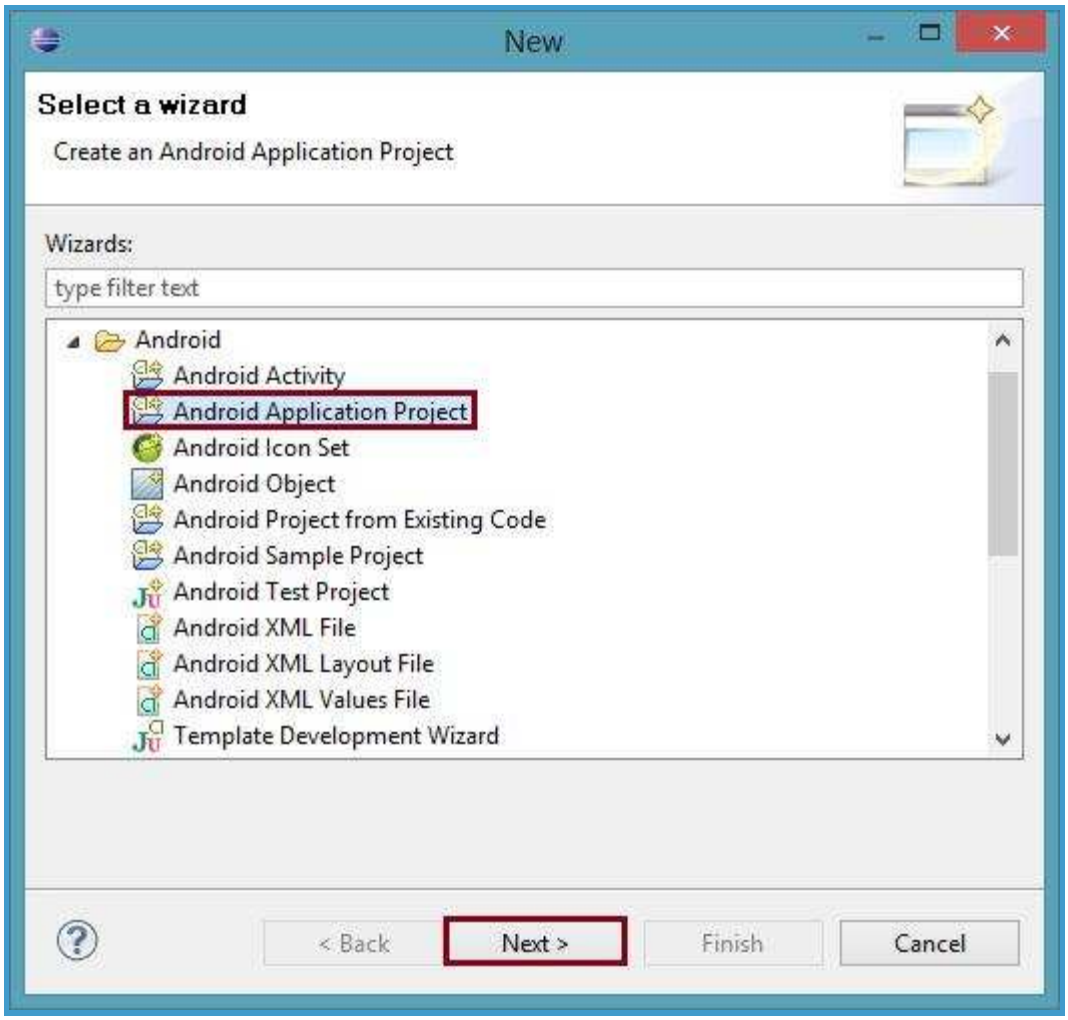
سنمر الآن إلى كيفية إنشاء أول مشروع أندرويد، سنتتبع الخطوات التالية:

فتح برنامج Eclipse :

نختار القائمة File كما يلي:



وستظهر النافذة التالية :

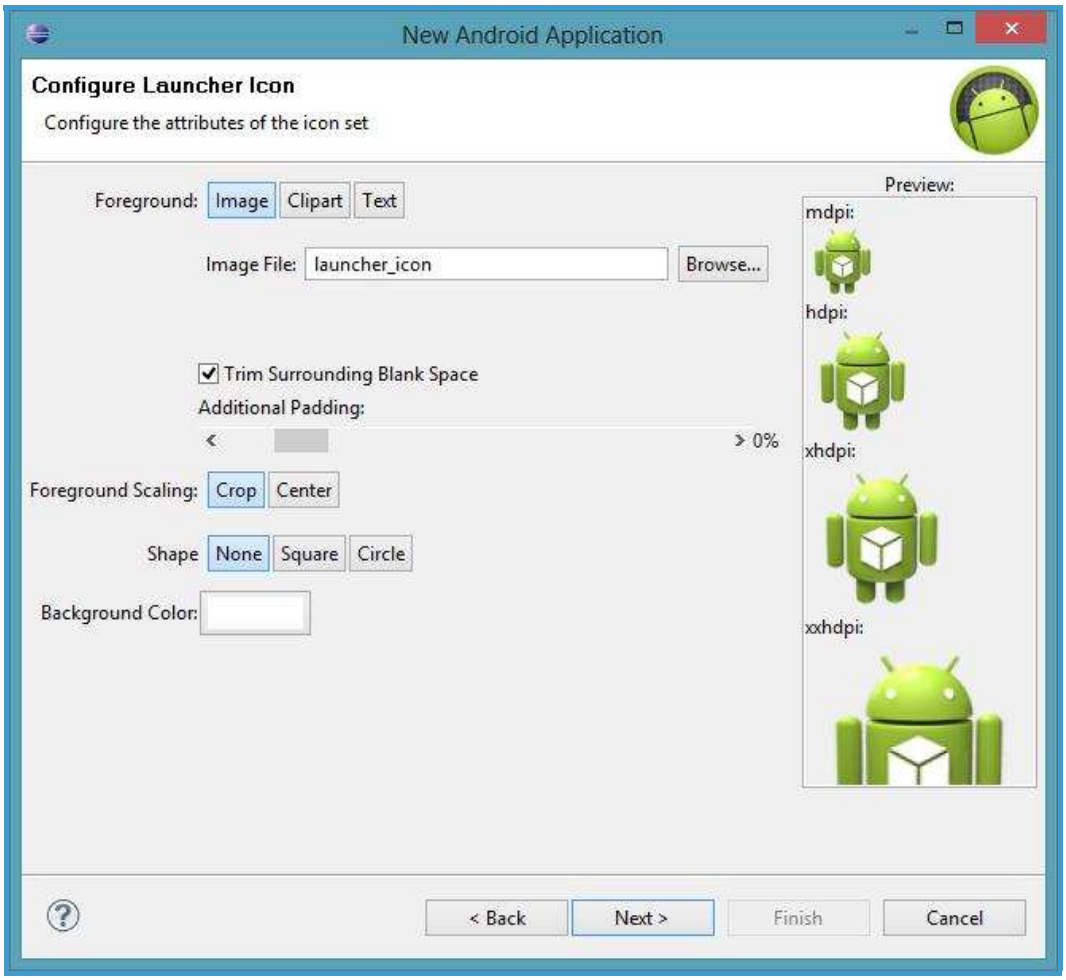


نختار Android Application Project ثم ننقر الزر next وستظهر النافذة الآتية :

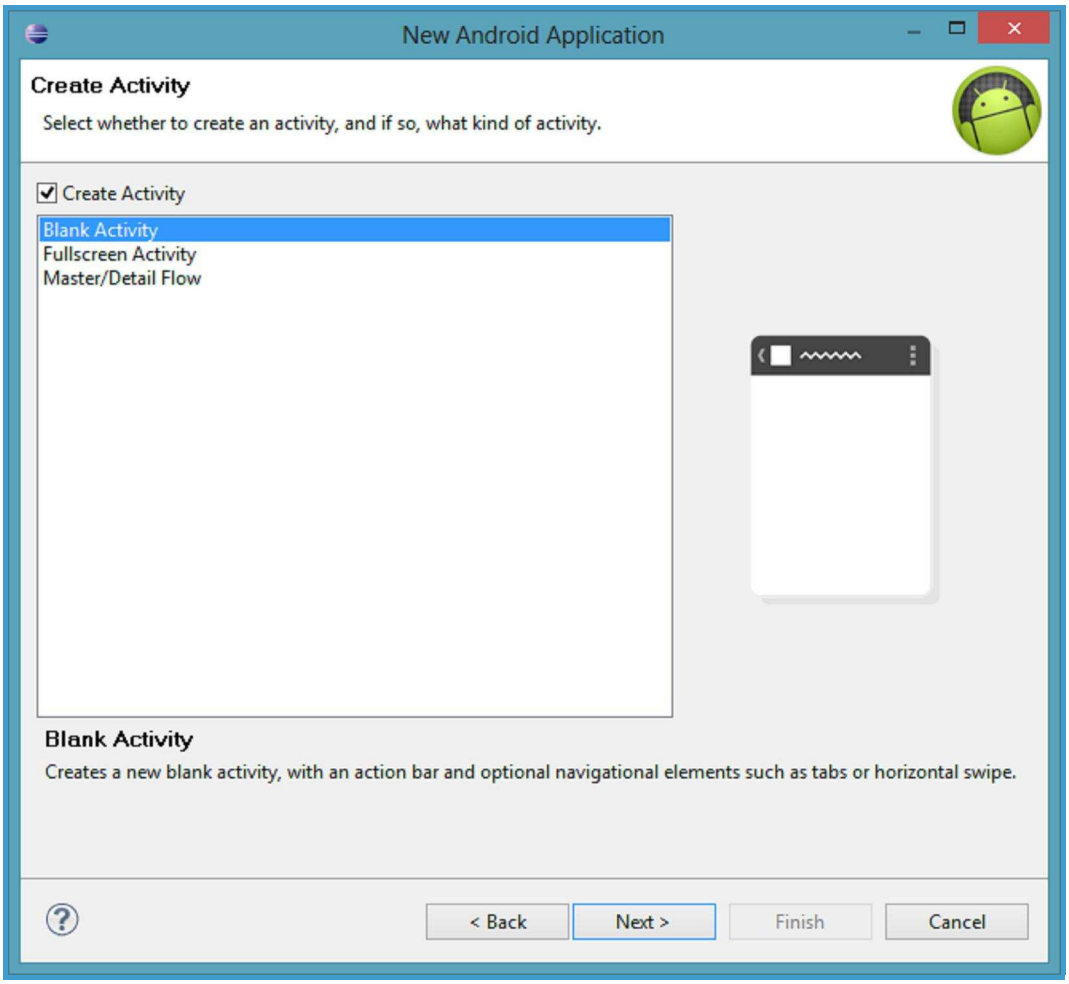
ستلاحظون أن هناك حقولا لابد من ملئها وهي :

Application Name وهو اسم التطبيق.
Project Name وهو اسم المشروع.
Package Name : مشاريع الجافا كلها تكون على هذا الشكل مكونة من حزمات (مجلدات) وهذه خاصية من خصائص الجافا وليست بمستجد في الأندرويد.
Minimum Required SDK هي النسخة الأدنى من الأندرويد التي يمكنها تشغيل تطبيقك هذا.
Target SDK وهي النسخة المستهدفة .
Compile With هنا سنختار النسخة التي سنبرمج بها (ترجمة الكود و تشغيله) التطبيق.

بعد ذلك ننقر على الزر Next مرتين وسنحصل على النافذة التالية:



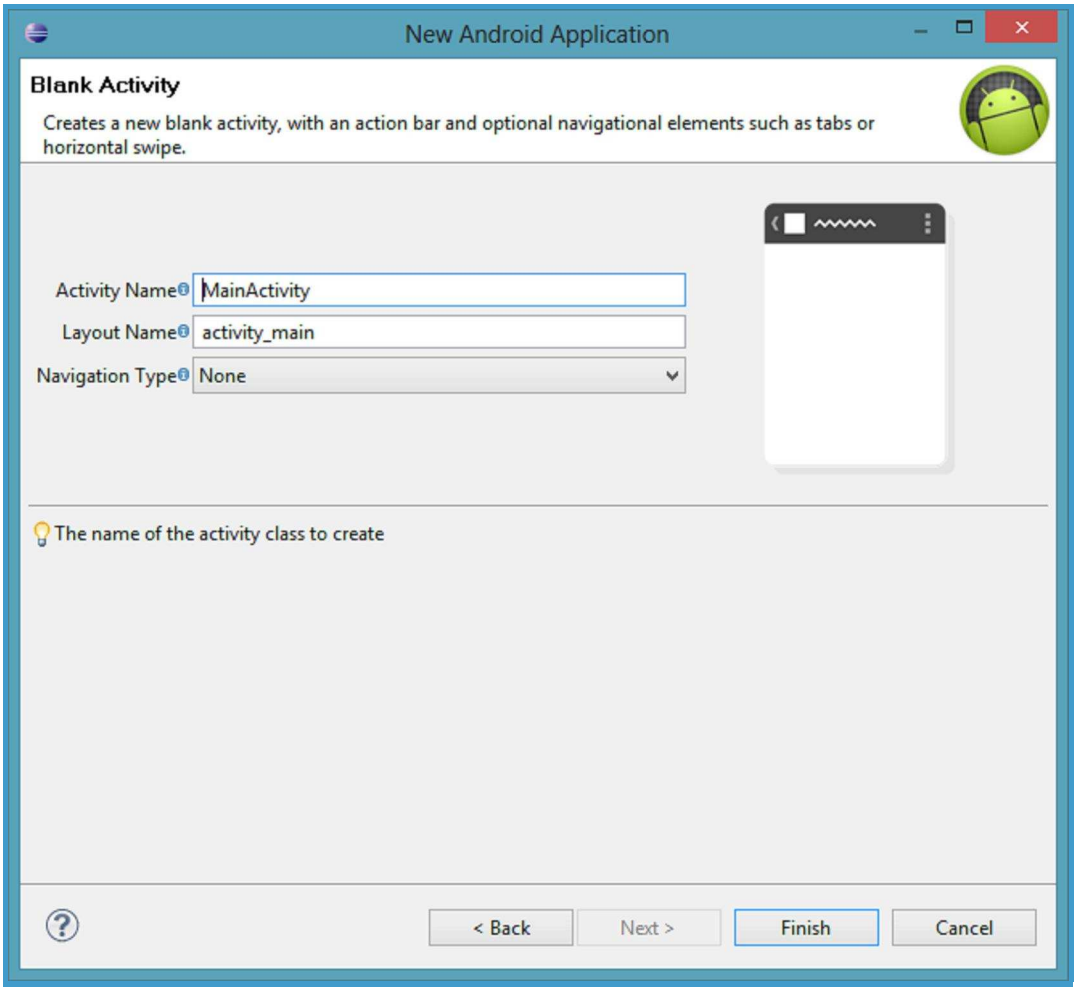
هذه النافذه تسهل عليكم إنشاء أيقونة لتطبيقكم....Next



هنا تجدون اختيارات لل Activity الرئيسية لتطبيقكم :

الإختيار الأول : Activity فارغة .
الإختيار الثاني:Activity تأخذ حيز الشاشة كليا.
الإختيار الثالث: Activity متقدمة.

ننقر على الزر Next ثم نذهب إلى النافذة التالية :



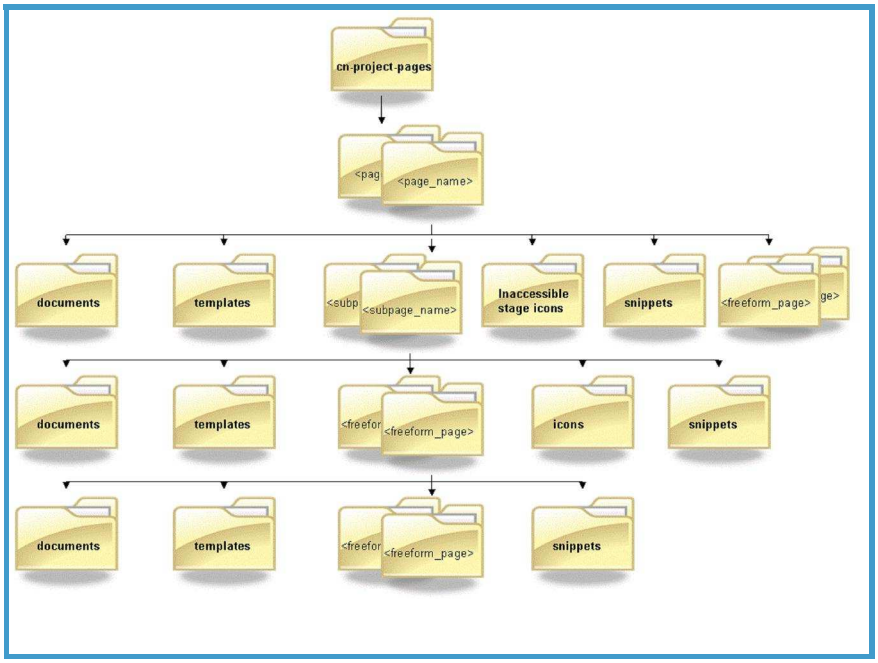
هنا نحدد اسم الActivity الرئيسية و اسم ملف التصميم (Layout) هذا الملف هو الذي سيحدد تصميم الActivity .سنتطرق لهذا الملف لاحقا.

ننقر على الزر Finish و هنا مبروك عليكم إنشاء أول مشروع لكم.

الملفات المكونة لمشروع أندرويد

التفاصيل كتب بواسطة: Ahmed Amansag الزيارات: 1166

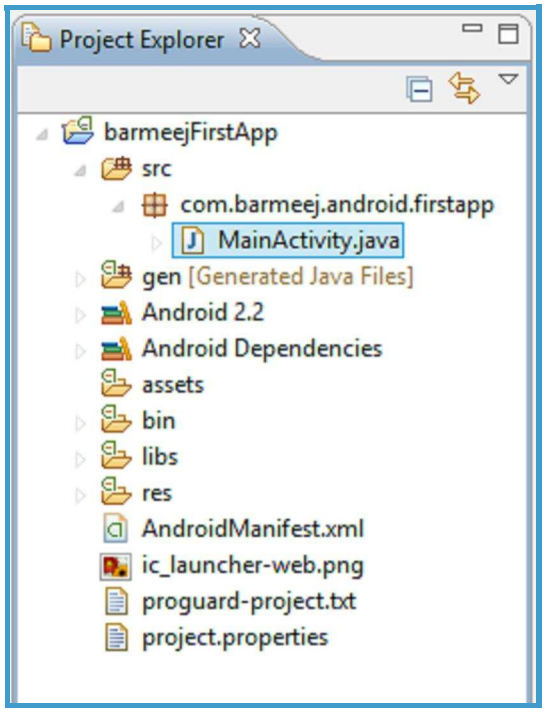
السلام عليكم و رحمة الله .



سنرى إخواني في هذا الدرس الملفات التي يتكون منها مشروعنا بالتفصيل و سنشرح كل ملف عى حدى . على بركة الله

شجرة المشروع:

بعد إنشاء أول مشروع لنا ، لابد أننا لاحظنا أن المشروع يتكون من العديد من الملفات و المجلدات ، وذلك يظهر جليا من متصفح المشروع (Project Explorer) :

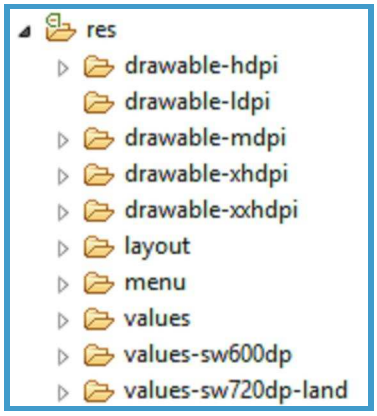


سنحاول شرح شجرة المشروع :

المجلد src :يضم ملفات الأكواد و الحزمات(Packages) .

المجلد gen : يضم ملفات Java يتم توليدها أوتوماتيكيا .

المجلد res : هذا المجلد مهم جدا و يجب فهم محتوياته، فهو يضم الموارد التي سيستخدمه التطبيق،فالننظر إليه عن كثب :



المجلدات drawable خاصة بتصميم مكونات التطبيق و الصور

المجلد layout يضم ملفات XML تحدد تصميم الActivity أو تصاميم نوافذ Dialog (إنذارات،إعلامات ...) أو تصاميم أخرى.

المجلدات values تضم ملفات XML تحدد قيم الألوان أو السلاسل النصية أو الأبعاد التي سيستخدمها التطبيق كلون الخلفية مثلا أو النص الذي سيكتب مثلا على زر ما .....

بالنسبة للمجلدات الأخرى التي لم نتطرق إليها،سيأتي موعدها عند الحاجة إلى ذلك .

عادة ، التطبيقات الكبرى لا تستغني عن طريقة تنظيم الموارد هذه، فهذه المجلدات تنظم تطبيقاتنا بحيث تمككنا من جعلها عالمية (بلغات متعددة مثلا أو مناسبة لأجهزة متعددة) وتمكننا كذلك من عدم خلط عملية تصميم الواجهات ببرمجة نواة التطبيق،فهي تنظم و تسهل عملية البرمجة،وربما ذهب الكثير إلى قول أن هذه الطريقة تستعمل نموذج MVC .

الفئة الرئيسية:

الفئة(Class) الرئيسية توجد في المجلد src داخل الحزمة التي أنشأناها عند إنشاء المشروع com.barmmej.android.firstapp وقد أسميتها MainActivity .سنعرض كود هذه الفئة وسنشرح ما يتطلب الشرح:

```
1  package com.barmeej.android.firstapp;
2
3  import android.app.Activity;
4  import android.os.Bundle;
5
6  public class MainActivity extends Activity {
7
8      @Override
9      protected void onCreate(Bundle savedInstanceState) {
10         super.onCreate(savedInstanceState);
11         setContentView(R.layout.activity_main);
12     }
13
14 }
```

الفئة الرئيسية تضم دالة واحدة (هذا لا يعني أنه لا يوجد غيرها بل سنكتفي بهذه في هذه المرحلة) هي الدالة onCreate ويتم استدعاء هذه الدالة عند إنشاء

ال Activity .

الدالة setContentView تمكننا من تحديد تصميم (Layout) للActivity و هو R.layout.activity_main وسنشرحه :

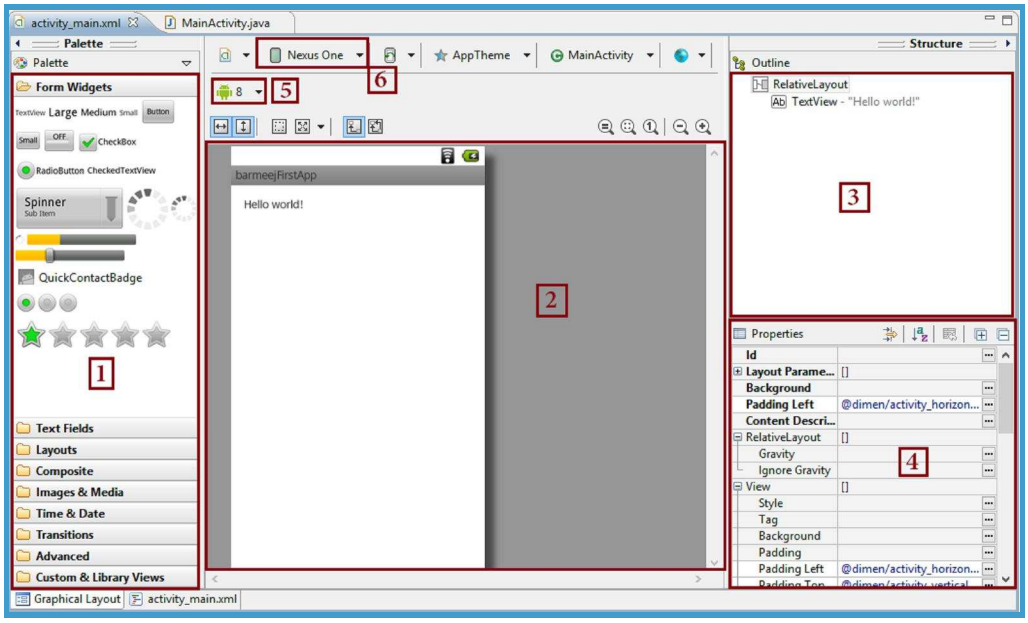
- R هو اسم فئة(Class) يتم توليدها وتحديثها تلقائيا كلما قمنا بإضافة أو تغيير تصميم (Layout) أو إضافة أو تغيير قيم(ألوان،أبعاد،سلاسل نصية) ، عموما هذه الفئة تضم معرفات (identificateur,identifier) لهذه العناصر التي ذكرناها.
- layout هنا نحدد نوع العنصر الذي نريد تحديده فربما يكون string عند الرغبة في تحديد سلسلة نصية، color عند الرغبة في تحديد لون ....
- activity_main هو اسم ملف التصميم (Layout) الذي تم إنشاؤه عند إنشاء المشروع.

سنحاول سرد بعض الدوال المهمة الأخرى التي نجدها في فئة ال Activity :

الدالة	الوصف
onCreate	يتم استدعائها عند إنشاء الActivty لأول مرة.وفيها يتم القيام بكل الإعدادت الأولية من تحديد للتصاميم.وهي متبوعة غالبا بالدالة onStart.
onRestart	يتم استدعاؤها بعد توقيف الActivity . هي متبوعة غالبا بالدالة onStart.
onStart	يتم استدعاؤها عند ظهور ال Activity للمستخدم .وتتبعها الدالة onResume إذاكانت ظاهرة أو onStop إذاكانت مخفية.
onResume	يتم استدعاؤها عندما تكون Activity في تفاعل مع المستخدم.دائما متبوعة ب الدالة onStop.
onPause	عند تعليق الActivity .
onStop	يتم استدعاؤها عند اختفاء ال Activity .
onDestroy	يتم استدعاؤها عند تدمير ال Activity .

تصميم الواجهة الرئيسية :

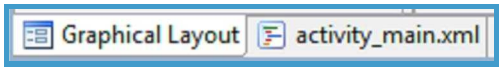
الملف الذي يمكننا من تصميم الواجهة الرئيسية للActivity هو الملف الذي أسميته activity_main.xml ونجده دائما في الرابط res/layout .عند فتح الملف سنجد ما يلي:



يمكن ملاحظة النافذة و هي مقسمة إلى مناطق متعددة:

المنطقة 1 : هذا الجزء يضم المكونات التي يتم إضافتها من أزرار و حقول كتابة و تنسيقات (Layout) و عناصر أخرى ....
المنطقة 2 : هذا الجزء يعطينا لمحة عن كيف سيكون التصميم على الهاتف و إليه يتم إضافة المكونات السابق الذكر .
المنطقة 3 : هنا يوجد تشجير للعناصر التي تمت إضافتها لتصميم الActivity .
المنطقة 4 : عند اختيار مكون ما من المنطقة 2 ، يمكننا تغيير خصاءصه في هذا الجزء(لون ، أبعاد ، ...).
المنطقة 5 : هنا يمكن تحديد نسخة الأندرويد التي يستهدفها التطبيق،ذلك لأن كل نسخة تأتي بإضافات عن سابقاتها فيما يخص المكونات و غيرها....
المنطقة 6 : هنا يمكننا اختيار بعض أنواع الهواتف او الألواح ، بالأحرى شاشاتها و أبعاد شاشاتها و ذلك لكي يتمكن المبرمج من معاينة تطبيقه على أكبر عدد من الأجهزة.

عند إضافة أو تغيير مكون ما في التصميم،يتم تلقائيا تحديث ملف activity_main.xml،كما يمكن تحديثه يدويا و ذلك بعرضه مباشرة وذلك بالإنتقال إلى اللسان (Ongelt) في أسفل النافذة كما يلي:



وستجدون الملف على الشكل التالي :

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/hello_world" />

</RelativeLayout>
```

إذن يمكن تعديل الملف من هنا ، لكن يشترط معرفة الحقائب التي نستخدمها و خصائص كل حقيبة، وذلك سهل بإذن الله و يأتي مع الممارسة.

عموما نستخدم ال GUI (الغرافيك) لأنه أسهل من ال XML .

هكذا نكون قد عرجنا على جميع الملفات المهمة في مشروع الأندرويد.سنمر الآن إلى أول تطبيق أندرويد .

أول تطبيق لك مع الأندرويد

التفاصيل كتب بواسطة: AMANSAG الزيارات: 415

السلام عليكم ورحمة الله و بركاته.

في هذه الفقرة ، سنتطرق إلى التطبيق الأول و سنلقي نظرة على بعض المكونات التي تكون ال Activity ، كما سنقوم بتشغيل تطبيقنا على المحاكي.

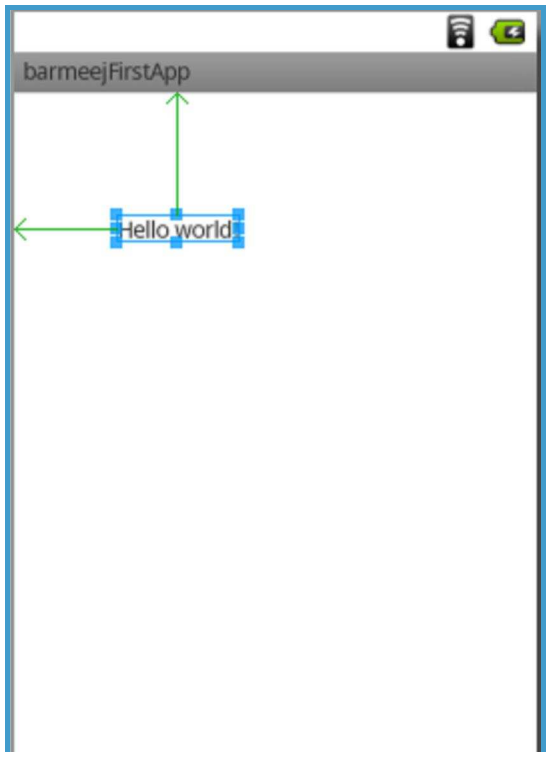
سنفتح المشروع السابق الذي قمنا بإنشائه مسبقا، أو بإمكانكم إنشاء مشروع جديد باتباع الخطوات السابقة.

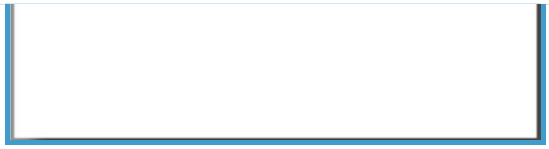
لنتأمل قليلا ملف التصميم (Layout) الذي يعطينا شكل الActivity و الذي يحدد مكوناتها. لنفتح الملف و تجدونه هنا res/layout :

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/hello_world" />

</RelativeLayout>
```

هناك نقط مهمة لابد من الوقوف عندها :

- كل ملف تصميم لابد من أن يضم نوعا من المكونات و هي التي ترث من الفئة ViewGroup و هي مكونات تحدد كيفية ترصيص (ترتيب) المكونات الأخرى.فهناك عدة طرق لذلك، فإذا استعملنا الRelativeLayout كما هو الحال عندنا، و هو يقوم بترصيص المكونات وفق إحداثياتها ونقصد بها المسافة الفاصلة بين المكون والحد الأعلى و المسافة الفاصلة بينه و بين الحد الأيسر





الRelativeLayout يعطينا حرية أكبر في ترصيص المكونات،لكنه ليس عمليا إلى حد ما،لأنه قد لا يكون التصميم ملائما لأنواع أخرى من الهواتف أو الألواح ...

يمكن كذلك أن نستعمل ال LinearLayout و الذي يقوم بترصيص المكونات إما أفقيا أو عموديا ، أو ال GridLayout الذي يرصص المكونات في شبكة و هناك عناصر أخرى سنراها في دروس قادمة.

- كل المكونات الأخرى يتم إضافتها داخل الوسم (Tag أو balise) التالي <RelativeLayout> </RealativeLayout> .

- المكون الذي تلاحظونه داخل الوسم السابق هو ال TextView وهو الذي يمكننا من أظهار نصوص و سلاسل نصية . بالنسبة للخصائص الخاصة بكل وسم مثل android:layout_width فهي تحدد عرض المكون وهذه الخاصية يمكن أن تأخذ القيم "wrap-content" أي أن يكون عرض المكون عرض محتواه،أو "match-parent" أي أن يكون عرض المكون هو عرض العنصر الذي يحتويه أو عرض أبيه (لأن هناك علاقة تربط بين العناصر في الأندرويد و هي علاقة الأب و الإبن) وهناك قيمة أخرى و هي تحديد القيمة التي نريدها مثلا "100dp" و dp هي وحدة من الوحدات المستعملة كـ sp و mm و px وغيرها .ولكم أن تعاينوا كل وحدة على حدى في الصفحة الخاصة بنظام الأندرويد هنا.

كذلك نلاحظ الخاصية android:text وهي التي تحدد لنا النص الذي ستم إظهاره من طرف هذا المكون،وهذا النص محدد كما يلي : @string/hello_world أي النص الذي تم تحديده في ملف XML تحت اسم (id أو معرف) hello_world ورابطه هو res/values /strings.xml .في هذا الملف يتم تحديد السلاسل النصية التي نريد استخدامها في تطبيقنا،وعندما نريد استخدامها نستدعيها بالطريقة السابقة.ربما يقول البعض لماذا لا نكتب النص مباشرة ؟ نعم يمكن ذلك ، لكن تصور معي لو أنك أردت أن يكون تطبيقك متعدد اللغات، أي بلغات مختلفة.ما العمل ؟ هل ستنشأ تطبيقا لكل لغة .ليس عمليا أليس كذلك؟وهنا تأتي هذه التقنية التي يمكننا فقط إنشاء ملف XML آخر بنفس المعرفات وبلغة أخرى...

سنحاول ،بعد هذه المقدمة ، أن ننجز تطبيقا بسيطا هدفه هو تغيير النص الذي يظهره ال TextView عند اللمس زر. سنبقى مع نفس المشروع سنغير ملف التصميم ،بالضبط سنغير ال Viewgroup أي RelativeLayout إلى LinearLayout لكي نلاحظ الطريقة التي يتم بها ترصيص المكونات داخله.

سيكون ملف التصميم على هذه الشاكلة :

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity" >

    <TextView
        android:id="@+id/textview"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/hello_world" />
    <Button
        android:id="@+id/button"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="change"
        />

</LinearLayout>
```

هناك بعض التغييرات البسيطة و هي كالتالي :

تم استبدال RelativeLayout بالعنصر LinearLayout :بالنسبة للخصائص layout_width و layout_height هي فقط لتحديد عرض و ارتفاع المكون.أما بالنسبة للخاصية orientation فهي لتحديد اتجاه ترصيص المكونات التي سيضمها LinearLayout ، وهذه الخاصية تأخذ القيمتين vertical (ترصيص المكونات عموديا أي مكون فوق مكون) أو horizontal (ترصيص المكونات أفقيا أي مكون بجانب مكون).

تم إضافة خاصية جديدة للمكون TextView و هي android:id والقيمة التي أعطيناها للخاصية هي textview ، يمكنكم وضع ما شئتم،وهذه الخاصية تحدد اسما خاصا بالمكون ومن خلاله يمكن استدعاءه برمجيا.سنرى ذلك في ما سيأتي.

تم إضافة مكون آخر هو المكون Button أي الزر وتم تحديد اسمه(id) والعرض والإرتفاع و النص الذي سيكتب على الزر.

بالنسبة للتصميم سيكون كما يلي :

الآن،بعدما قمنا بإعداد التصميم ، سنمر إلى الجانب البرمجي.لنفتح الملف الذي يحتوي على الفئة الرئيسية (Class) ، و سنعرف متغيرين الأول للزر الذي أضفناه(Button) و الثاني ل (TextView) :

```
1  package com.barmeej.android.firstapp;
2
3  import android.app.Activity;
4  import android.os.Bundle;
5  import android.view.View;
6  import android.view.View.OnClickListener;
7  import android.widget.Button;
8  import android.widget.TextView;
9
10 public class MainActivity extends Activity {
11
12     //التصريح بالمتغيرات
13     private Button myButton;
14     private TextView myTextView;
15
16     @Override
```

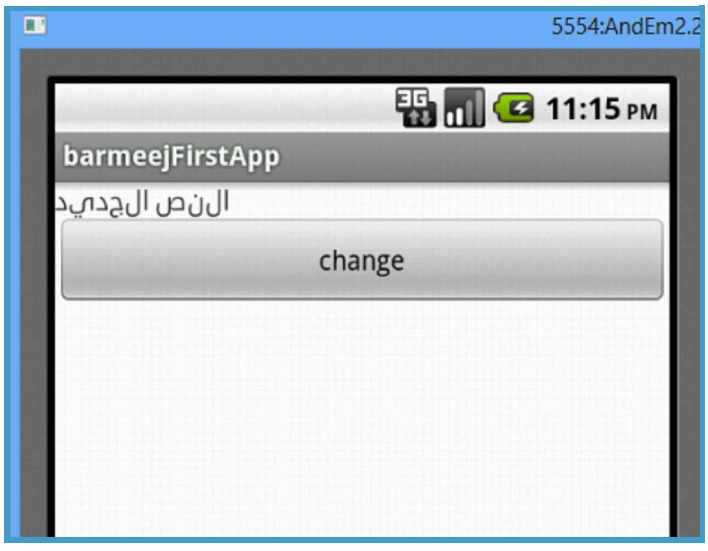
```
17      protected void onCreate(Bundle savedInstanceState) {
18          super.onCreate(savedInstanceState);
19          setContentView(R.layout.activity_main);
20          //تهيئة المتغيرات
21          myButton = (Button)findViewById(R.id.button);
22          myTextView = (TextView)findViewById(R.id.textview);
23          //عند الضغط على الزر يتم تنفيذ ما يلي
24          myButton.setOnClickListener(new OnClickListener() {
25
26              @Override
27              public void onClick(View v) {
28                  //TextView نغير النص في ال
29                  myTextView.setText("النص الجديد");
30              }
31          });
32
33      }
34
35  }
```

الكود بسيط للغاية،قمنا بتعريف المتغيرين myButton و myTextView و قمنا بربطهما بالزر و TextView (الذين قمنا بإضافتهما لتصميم التطبيق) ، وهنا يأتي دور المعرف (id) حيث بواسطته نربط بين المتغير myButton و الزر في التصميم.

عند لمس الزر يتم تنفيذ السطر البرمجي

```
1   myButton.setOnClickListener(new OnClickListener() {
2
3               @Override
4               public void onClick(View v) {
5                   //TextView نغير النص في ال
6                   myTextView.setText("النص الجديد");
7               }
8           });
```

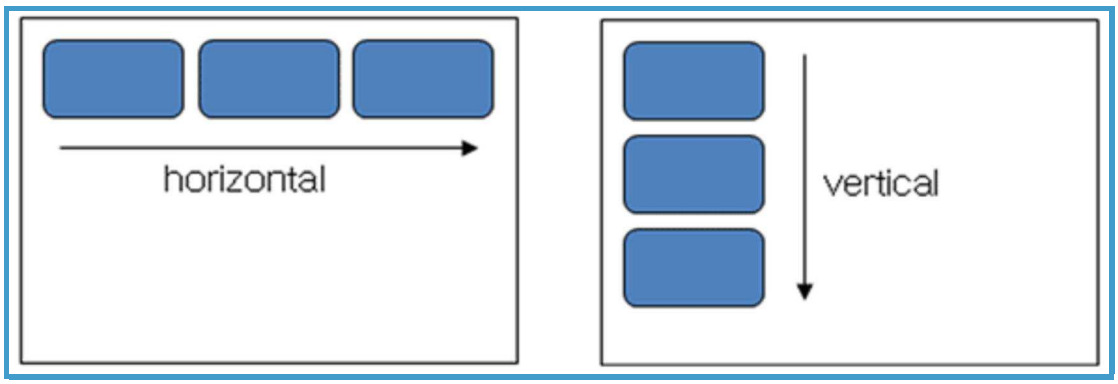
يمكنكم تشغيل التطبيق في المحاكي و عند الضغط على الزر سيتغير النص hello world إلى نص جديد كما في الصورة :



إذن الخلاصة التي يمكن أن نخرج بها من تطبيقنا الأول هي أن أول شيئ يمكن البدء به هو إنشاء تصميم تكون المكونات فيه معرفة بمعرّفات (id) يتم بواسطتها ربط هذه المكونات بالمتغيرات البرمجية التي يمكننا من خلالها التحكم في هذه المكونات ...

LinearLayout المكون

التفاصيل كتب بواسطة: Ahmed Amansag الزيارات: 182



بسم الله الرحمــن الرحيم : السلام عليكم ورحمة الله و بركاته.

في هذا الدرس سنرى ،إن شاء الله، التصميم LinearLayout و كيفية استخدامه و بعض خصائصه.

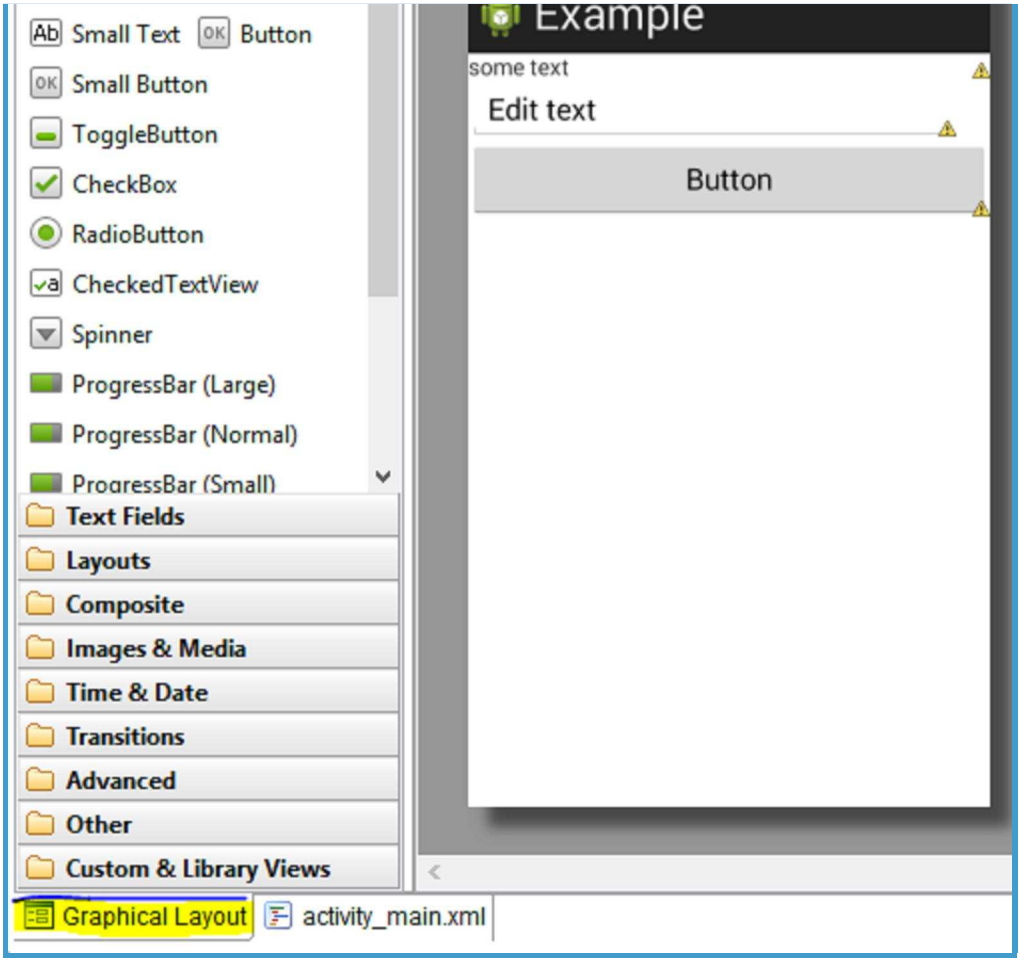
التصميم LinearLayout هو مكون يقوم بترصيص المكونات الأخرى داخله بطريقة خطية، أي إما أفقيا أو عموديا.والخاصية "orientation" هي التي تحدد بأي كيفية سيتم الترصيص.وهذه الخاصية تأخذ القيمتين "vertical" أي عموديا أو "horizontal" أي أفقيا.

سنرى في هذا الدرس مثالا ندرج فيه الحالتين:ننشئ مشروعا جديدا نسميه LinearLayoutExample ، وما يهمنا هو ملف تصميم التطبيق و سنجعله كما يلي:

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:orientation="vertical" >

    <TextView
        android:text="some text"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <EditText
        android:layout_width="300dp"
        android:layout_height="wrap_content"
        android:text="Edit text"/>
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Button"/>

</LinearLayout>
```

الLinearLayout يضم هنا ثلاث عناصر نسميها "أبناء" أو children و يقوم بترصيصها حسب قيمة الخاصية orientation، هنا سيتم ذالك عموديا:أي سيضع المكون الأول في الأعلى ثم التالي تحته ثم الموالي تحته....

عند معاينة ملف ال xml عن طريق Graphical Layout ستشاهدون ما يلي :



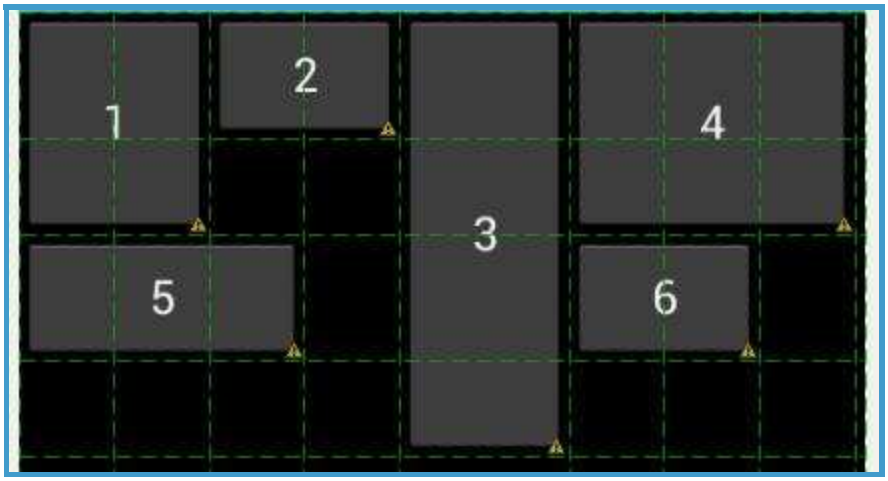
تلاحظون أن المكونات تم ترصيصها بطريقة عمودية ، ولكي يتم ترصيصها بطريقة عمودية يكفي أن نغير الخاصية orientation في LinearLayout و نحصل على ترصيص أفقي لكل العناصر.

هكذا نكون قد أعطينا فكرة عن كيفية استخدام LinearLayout وهي بسيطة جدا.

GridLayout

التفاصيل كتب بواسطة: Ahmed Amansag الزيارات: 131

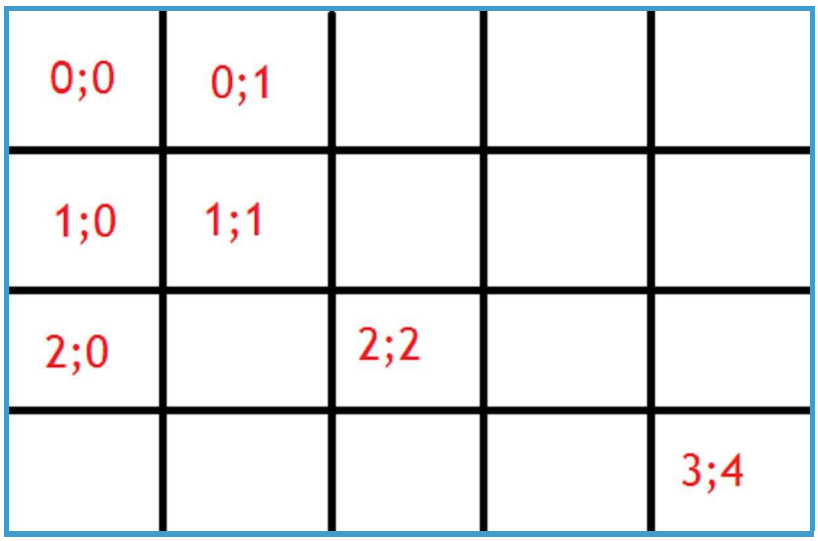
بسم الله الرحمـــن الرحيم :



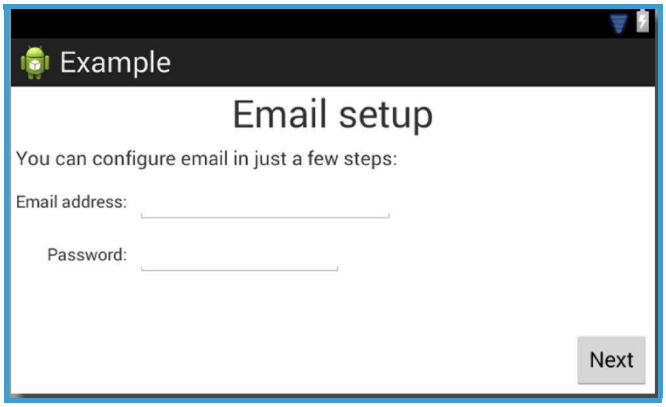
مرحبا إخواني،سنرى في درسنا اليوم تعريفا للـ GridLayout وطريقة استعماله ثم سنرى مثالا بسيطا نطبق فيه ما تعلمناه.

GridLayout هو تصميم يضع أبنائه في شبكة مستطيلة .
الأبناءيمكن أن يشغلوا واحدة أو أكثر من الخلايا المتجاورة ، وذلك باستعمال الخاصية rowSpan و columnSpan ،حيث الأول يمثل عدد الأسطر التي نريد أن يحتلها المكون و الثاني يمثل عدد الأعمدة التي نريد للإبن أن يحتلها.

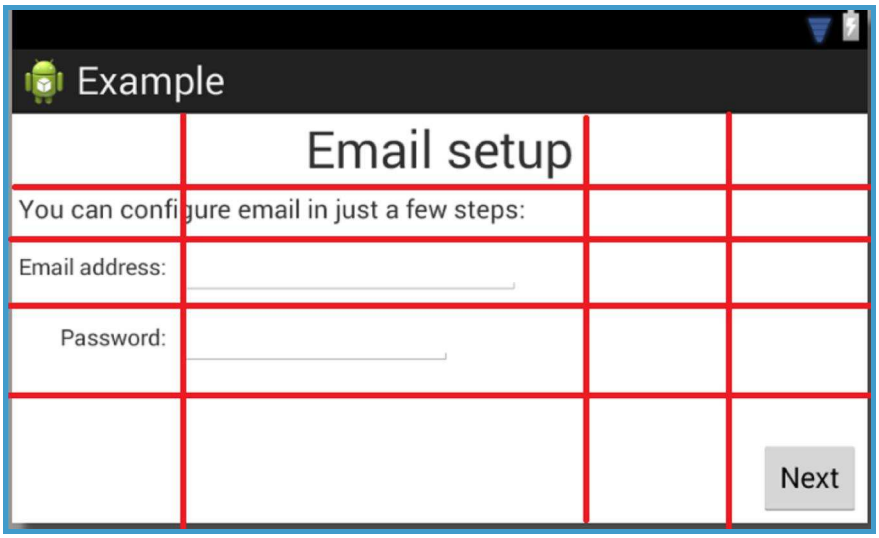
الطريقة التي يتم ترصيص بها المكونات هي أن ال GridLayout يقوم بترصيصها أفقيا حتى ينتهي من السطر ثم ينتقل إلى السطر الموالي،أو يضع المكون في الخانة التي حددناها له باستعمال الخاصيتين layout_row و layout_column و هما يمثلان السطر والعمود الذان تتواجد فيهما الخانة،على أساس أن رقم العمود و السطر يبدأ من الصفر كما في الصورة الموالية :



وهنا يمكن جليا أن نلاحظ أهمية هذا المكون في تسهيل عملية إنشاء تصاميم معقدة كالتصاميم التالية:



سنحاول إذن أن نشئ المثال الأول و إليكم الصورة الحقيقية و الخانات الخفية التي لا نلاحظها و التي تساعدنا في صنع التصميم المراد انطلاقا مما قلناه سابقا:



سنجعل المكون الأول (Email setup) يحتل الخانات الأربع كلها وذلك بوضع columnSpan = 4 ،نفس الشيئ بالنسبة للمكون الموالي،وأخيرا الزر سنجعله في الخانة الأخيرة وذلك باستعمال الخاصيتين layout_row و layout_column . وإليكم ملف الـ xml :

```
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:alignmentMode="alignBounds"
    android:columnCount="4"
    android:columnOrderPreserved="false"
    android:useDefaultMargins="true" >

    <Button
        android:layout_columnSpan="4"
        android:layout_gravity="center_horizontal"
        android:text="Button"
        android:textSize="32dip" />

    <TextView
        android:layout_columnSpan="4"
        android:layout_gravity="left"
        android:text="You can configure email in just a few steps:"
```

```
        android:textSize="16dip" />

    <TextView
        android:layout_gravity="right"
        android:text="Email address:" />

    <EditText android:ems="10" />

    <TextView
        android:layout_column="0"
        android:layout_gravity="right"
        android:text="Password:" />

    <EditText android:ems="8" />

    <Space
        android:layout_column="0"
        android:layout_columnSpan="3"
        android:layout_gravity="fill"
        android:layout_row="4" />

    <Button
        android:layout_column="3"
        android:layout_row="4"
        android:text="Next" />

</GridLayout>
```

بعض الخاصيات تحتاج إلى شرح :

layout_gravity : ونقصد بها كيف نريد أن يتمركز المكون داخل المكون الأب(الخلية) الذي يضمه،مثلا على اليمين (right) او اليسار (left) او الأسفل (bottom) او الأعلى (top) أو أن يملاء المكون الأب (fill) ....
columnCount وتمثل عدد الأعمدة التي ستكون جدولنا،يمكن كذلك تحديد عدد الأسطر باستخدام الخاصية rowCount.

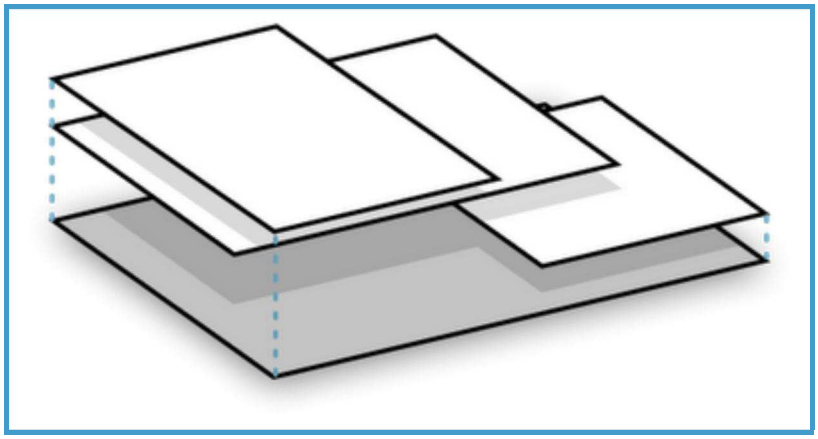
هكذا نكون قد أحطنا بأهم الجوانب فيما يخص هذا المكون ،يمكنكم الإطلاع على مزيد من الشرح في موقع أندرويد.

و السلام عليكم.

FrameLayout

التفاصيل كتب بواسطة: Ahmed Amansag الزيارات: 115

السلام عليكم و رحمة الله:



مرحبا أحبتي،اليوم لنا موعد مع عنصر آخر مميز و هو FrameLayout ،سنرى بعض خصائصه في هذا الدرس إن شاء الله.

هذا المكون لا يشبه كثيرا سابقيه(LinearLayout,GridlLayout,RelativeLayout)فهو مصمم لكي يضم عنصرا واحدا فقط،وهذا لايعني أنه لايمكن أن يضم أكثر من عنصر بل يمكنه ذلك لأنه يرث من كلاسViewGroup أي المكونات التي يمكن أن تضم أكثر من عنصر،ويستعمل أيضا لوضع مكون فوق مكون،كما يمكن وضع به مجموعة من المكونات ثم نتحكم برمجيا في إظهارها وذلك باستدعاء الدالة(setVisibility(int visibility .

سأعرض لكم هنا ملف xml يوضح كيفية استعمال هذا المكون لوضع عناصر فوق بعضها :

```
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="300dp"
    android:layout_height="200dp" >

    <View
        android:background="#113400"
        android:layout_width="100dp"
        android:layout_height="150dp"
        android:layout_gravity="bottom|right" />

    <View
        android:background="#005667"
        android:layout_width="200dp"
        android:layout_height="175dp"
        android:layout_gravity="top|left" />

    <View
        android:background="#113467"
        android:layout_width="100dp"
        android:layout_height="match_parent"
        android:layout_gravity="top|left" />

</FrameLayout>
```

يكفي أن تنقلوا هذا إلى ملفكم و ستلاحظون أن العناصر تتموضع فوق بعضها.

هكذا نكون قد أنهينا هذا الدرس و إلى درس آخر إن شاء الله.

RelativeLayout

التفاصيل كتب بواسطة: Ahmed Amansag الزيارات: 114

بسم اله الرحمـن الرحيم:

مرحبا بكم أحبتي،درسنا اليوم سيكون حول عنصر أخر وهو التصميم (أو الحاوية) RelativeLayout.

الطريقة التي يتم بها ترصيص مكون داخل الـ RelativeLayout تكون بالنسبة للعناصر المحيطة به،أي أنه سيتموضع حسب تلك العناصر.

هذه بعض الخاصيات المستعملة :

تموضع المكون بالنسبة للـRelativeLayout :

android:layout_alignParentTop تأخذ القيم true /false ونقصد بها هل نريد أن يوضع الحد العلوي للمكون و الحد العلوي للمكون الذي يحتويه(الأب) في استقامة واحدة.

android:layout_alignParentBottom تأخذ القيم true /false ونقصد بها هل نريد أن يوضع الحد السفلي للمكون و الحد السفلي للمكون الذي يحتويه(الأب) في استقامة واحدة.

نفس الشيء بالنسبة لـ**android:layout_alignParentLeft** و **android:layout_alignParentRight**

تموضع المكون بالنسبة للـمكونات الأخرى :

قبل تحديد موضع عنصر ما بالنسبة للعناصر الأخرى،لابد أن تكون هذه العناصر معرفة ب id يميز كل واحدة منها،مثلا نريد تحديد موضع مكون ما بالنسبة للعنصر الذي سنعرفه ب android:id=@+id/object :

إذا أردنا أن نضع المكون :

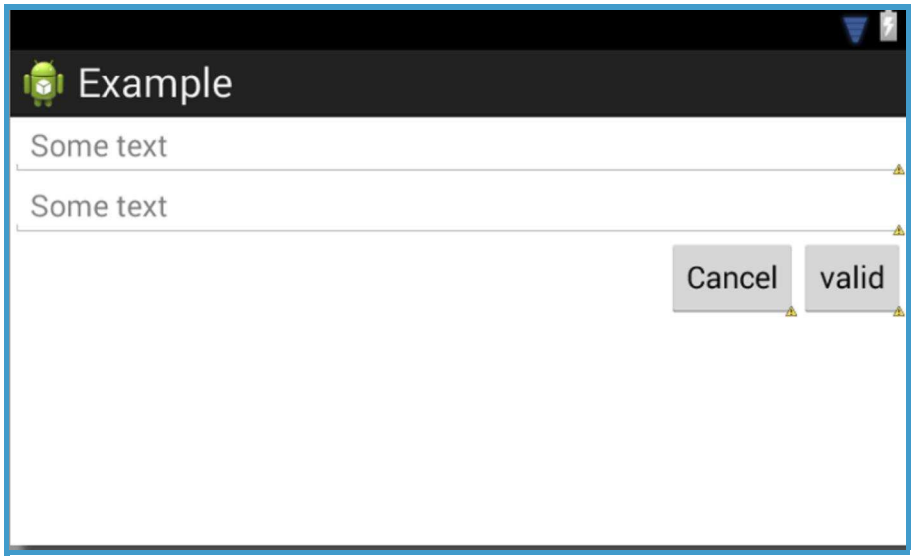
فوق المكون المعرف نضيف الخاصية التالية للمكون الذي نريد ترصيصه : **android:layout_above=@id/object** (بدون "+" لأننا هنا نستعمل الإسم و لا ننشئه)
تحت المكون **android:layout_below=@id/object**
على يسار المكون **android:layout_toLeftOf=@id/object**
على يمين المكون **android:layout_toRightOf=@id/object**
بحيث يكون حده الأعلى في استقامة معى الحد الأعلى للمكون **android:layout_alignTop=@id/object**
نفس الشيء بالنسبة للإستقامة مع الحد السفلي و الأيم و الايسر (left,right,bottom)

الآن وصلنا إلى التطبيق ، سنطبق بعض ما رأيناه في الملف التالي :

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

    <EditText
        android:id="@+id/object"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:layout_alignParentTop="true"
        android:hint="Some text" />

    <EditText
        android:id="@+id/obj"
```

```
17          android:layout_width="fill_parent"
18          android:layout_height="wrap_content"
19          android:layout_alignParentRight="true"
20          android:layout_below="@id/object"
21          android:hint="Some text" />
22
23      <Button
24          android:id="@+id/valid"
25          android:layout_width="wrap_content"
26          android:layout_height="wrap_content"
27          android:layout_alignRight="@id/obj"
28          android:layout_below="@id/obj"
29          android:text="valid" />
30
31      <Button
32          android:id="@+id/cancel"
33          android:layout_width="wrap_content"
34          android:layout_height="wrap_content"
35          android:layout_alignTop="@id/valid"
36          android:layout_toLeftOf="@id/valid"
37          android:text="Cancel" />
38
39  </RelativeLayout>
```

و سنحصل على تصميم يشبه هذا



هكذا يكون هذا الدرس منتهيا وعلى أمل اللقاء بكم،أستودعكم الله

الأزرار

التفاصيل كتب بواسطة: Ahmed Amansag الزيارات: 127

بسم الله الرحمـــن الرحيم:

نصل إخواني إلى المرحلة "التفاعلية" مع تطبيقاتنا،وسنبدأ بالأزرار

أنواع الأزرار:

الزر هو مكون المكونات التي تسمح للمستعمل التفاعل مع التطبيق،فهي عند "لمسها" تستدعي القيام بعملية ما أي ترسل حدثا .سنرى فيما بعد كيف نستجيب لهذه الأحداث.

يمكن لنا أن نضع نصا داخل الزر أو صورة (أيقونة) أو هما معا.

في ملف xml :

زر بنص:

```
Button>
 "android:layout_width="wrap_content
"android:layout_height="wrap_content
  "android:text="@string/button_text
                                </ ...
```

زر بأيقونة:

```
ImageButton>
 "android:layout_width="wrap_content
"android:layout_height="wrap_content
"android:src="/@drawable/button_icon
                                </ ...
```

لاحظوا هنا استعملنا الفئة ImageButton .

زر بنص و أيقونة:

```
Button>
 "android:layout_width="wrap_content
"android:layout_height="wrap_content
  "android:text="@string/button_text
"android:drawableLeft="@drawable/button_icon
                                </ ...
```

الإستجابة لأحداث لمس الزر:

نقصد بالإستجابة لأحداث الزر،هو عندما يتم لمس الزر ماهي العملية التي نريد القيام بها،مثلا إرسال رسالة،فتح مجلد،إطفاء الهاتف.....

ومن أجل فعل ذلك،هناك طريقتان،الأولى تقتضي أن نحدد داخل ملف xml اسم الدالة التي يجب استدعائها عند لمس الزر مثلا سنرسل رسالة عند لمس الزر التالي:

```
"Button xmlns:android="http://schemas.android.com/apk/res/android>
                                        "android:id="@+id/button_send
                                    "android:layout_width="wrap_content
                                  "android:layout_height="wrap_content
                                    "android:text="@string/button_send
                                      </ "android:onClick="sendMessage
```

هنا نستخدم الخاصية android:onClick ثم نحدد اسم الدالة،لكن أين يجب أن تتواجد هذه الدالة؟

يجب أن تكون في كلاس الـActivity المرتبطة بملف الxml الذي يضم هذا الزر،ويجب أن تكون صيغتها على الشكل التالي:

```
                                    } (public void sendMessage(View view
                                Do something in response to button click //
                                                                        {
```

ويجب أن تكون الدالة علنية(public) ثم أن تكون القيمة التي تنتجها هي لاشيء (void) و يكون البارامتر الوحيد الذي تأخذه هو من فئة View.

أما الطريقة الثانية فتقتضي أولا أن يكون للدالة معرِّف (id) والذي من خلاله يمكن معرفة الزر،ثم بعد ذلك في كلاس ال Activity نعرف كائنا من فئة Button كمايلي:

```
                                ;(Button button = (Button) findViewById(R.id.button_send
```

وهذا الزر هو نفسه الزر الذي أنشأناه في ملف الXML .

الدالة findViewById تبحث في عن العناصر باستعمال معرِّفها.

بعد ذلك،نستدعي الدالة setOnClickListener للزر كما يلي

```
                                } ()button.setOnClickListener(new View.OnClickListener
                                                    } (public void onClick(View v
                                    Do something in response to button click //
                                                                            {
                                                                              ;({
```

داخل الدالة onClick نضع الإجراءات التي نريدها عند لمس الزر.

حقل النصوص EditText

التفاصيل كتب بواسطة: Ahmed Amansag الزيارات: 104

بسم الله الرحمـــن الرحيم:

السلام عليكم و رحمة الله و بركاته، نلتقي اليوم أحبتي،مع مكون آخر يمكن من تفاعل المستعمل مع التطبيق و هو حقل النص EditText.

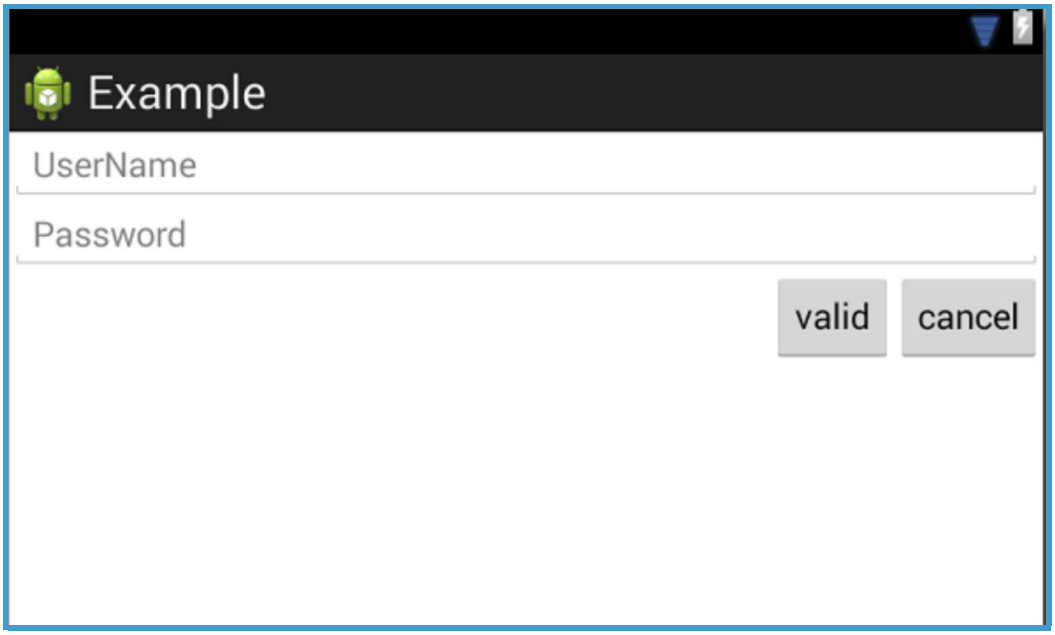
يسمح حقل نص EditText للمستخدم بكتابة نص في التطبيق الخاص بك. يمكنك الحقل من كتابة نص إما أن يتكون من سطر واحد أو متعدد الأسطر.عند لمس الحقل تظهر مباشرة لوحة مفاتيح للكتابة.

تمكننا الحقول EditText من تحديد نوع المدخل الذي نريد من المستعمل أن يقوم بإدخاله،فمثلا يمكننا أن نفرض على المستعمل أن يقوم بإدخال أعداد صحيحة فقط،أو أعداد عشرية،أو أرقام هواتف....وذلك باستعمال الخاصية android:inputType التي تأخذ القيم:

text : وهي القيمة الإفتراضية وتمكن من إدخال جميع أنواع الرموز.وبذلك تكون لوحة المفاتيح التي تظهر شاملة لكل الرموز.
number : وتمكننا من جعل لوحة المفاتيح لا تحتوي إلا على أرقام
textEmailAdress : تجعل لوحة المفاتيح تحتوي على الرمز @ .
textCapWords :تكون الحروف اللتينية كلها كبيرة.

هناك قيم أخرى تحدد بالتفصيل نوع المدخل يمكن لكم معاينتها عند إضافة حقل النص في ملف الxml.

فيما سيأتي سنرى كيف نتعامل مع حقول النص،لننشئ التصميم التالي :



كود ال xml سيكون على هذا الشكل :

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

    <EditText
        android:id="@+id/edittext"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:hint="UserName" />

    <EditText
        android:id="@+id/edittext"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
        android:hint="Password" />

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="right"
        android:orientation="horizontal" >

        <Button
            android:id="@+id/valid"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="valid" />

        <Button
            android:id="@+id/cancel"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="right"
            android:text="cancel" />
    </LinearLayout>

</LinearLayout>
```

كما تلاحظون ،هذا الفورم خاص مثلا بالدخول إلى تطبيق،يحتاج إلى user و password،وكما تعلمون أن الباسورد لايتم إظهاره،لذا قمنا بجعل الخاصية inputType الخاصة بحقل نص كلمة السر ،تأخذ القيمة textPassword .

من خلال كلاس ال Activity ، سنحاول استعادة ما كتب في الحقلين ومقارنتهما مثلا ب user و password و إذا تطابقا نقوم بإظهار نافذة تبين تطابق ما بداخل الحقلين(EditText) و الكلمتين،أما إذا لم يتطابقا ننبهه أن هناك خطأ ما في الحقلين.وهذا هو كود الكلاس الرئيسية:

```
Activity {



 savedInstanceState) {
```

```
ceState);
tivity_main);
ViewById(R.id.userEdit);
findViewById(R.id.passwordEdit);
yId(R.id.valid);
ew View.OnClickListener() {


ew v) {
Edit.getText().toString();
wordEdit.getText().toString();
er") && pass.equals("password")){
(getApplicationContext(), "the user and the pass are valid", Toast.LENGTH_SHORT).show();

(getApplicationContext(), "Wrong user or pass", Toast.LENGTH_SHORT).show();



3yId(R.id.cancel);
new View.OnClickListener() {


ew v) {
```

نقط مهمة وجب التنبه لها:

أولا التصريح بجميع المكونات التي نحتاجها و التي توجد في ملف الxml.
تعريف ما قمنا بتصريحه عبر الid الخاص بكل واحد،مثلا

```
1  userEdit = (EditText) findViewById(R.id.userEdit);
```

عند لمس الزر،لابد من أن نكون قد حددنا ما سيحدث(تعريف الأحداث)،مثلا عند لمس الزر valid ،التطبيق سيستخرج ما كتب في الحقلين (user,pass)ويقارنهما بما كتبناه آنفا.الزر cancel ليس مهما في الوقت الراهن،يمكن مثلا أن نخرج من التطبيق أو ....

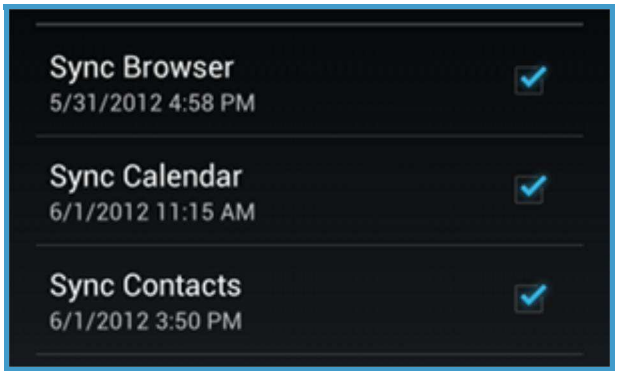
بالنسبة لToast سنتطرق لها فيما سيأتي .

أتنمى أن يكون الدرس مفيدا ،لا تنسوا أن تطرحوا تساؤلاتكم.

دمتم في رعاية الله.

صندوق الإختيار-CheckBox

التفاصيل كتب بواسطة: Ahmed Amansag الزيارات: 95



بسم الله الرحمان الرحيم:

مرحبا أحبابي،درسنا اليوم سيأخذنا إلى مكون من المكونات التي نستعملها كثيرا في تطبيقاتنا و هو صندوق الإختيار كما في الصورة.

كما هو مفهوم من اسم المكون صندوق الإختيار CheckBox،فهذا المكون يستخدم عندما نريد من المستعمل أن يختار مثلا عنصرا ما أو لا يختاره.

عندما يختار المستعمل صندوق الإختيار،هذا الأخير يستدعي الحدث onClick و هنا يمكننا أن نقوم بالإجراءات اللازمة التي نريدها.

سنقوم بإنشاء تطبيق يقوم فيه المستعمل باختيار نوع الأطعمة التي يفضلها

أولا دعونا نمر إلى التصميم و ملف الxml :

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:layout_width="fill_parent"
4      android:layout_height="fill_parent"
5      android:orientation="vertical" >
6      <TextView
7          android:layout_width="wrap_content"
8          android:layout_height="wrap_content"
9          android:text="Which foods do you prefer?"/>
10
11     <CheckBox
12         android:id="@+id/checkbox_meat"
13         android:layout_width="wrap_content"
14         android:layout_height="wrap_content"
15         android:text="meat" />
16
17     <CheckBox
18         android:id="@+id/checkbox_vegetables"
19         android:layout_width="wrap_content"
20         android:layout_height="wrap_content"
21         android:text="vegetables" />
22
23     <Button
24         android:id="@+id/button"
25         android:layout_width="wrap_content"
26         android:layout_height="wrap_content"
27         android:text="OK" />
28
29 </LinearLayout>
```

تصميمنا يضم صندوقي اختيار،كل واحد منهما معرف بمعرِّف id وحيد،ويحمل نصاً يظهر بجانب الصندوق.أضفنا كذلك زرا .

عند اختيار المستعمل لما يريد أن يختاره،يقوم بلمس الزر و هنا سنتمكن من تحديد اختياراته و ضعها مثلا في قاعدة بيانات أو .....

في مثالنا نحن،سنقوم فقط بإظهار نافذة تكون فيها اختياراته،و هذا هو كود الكلاس الرئيسية (Activity):

?

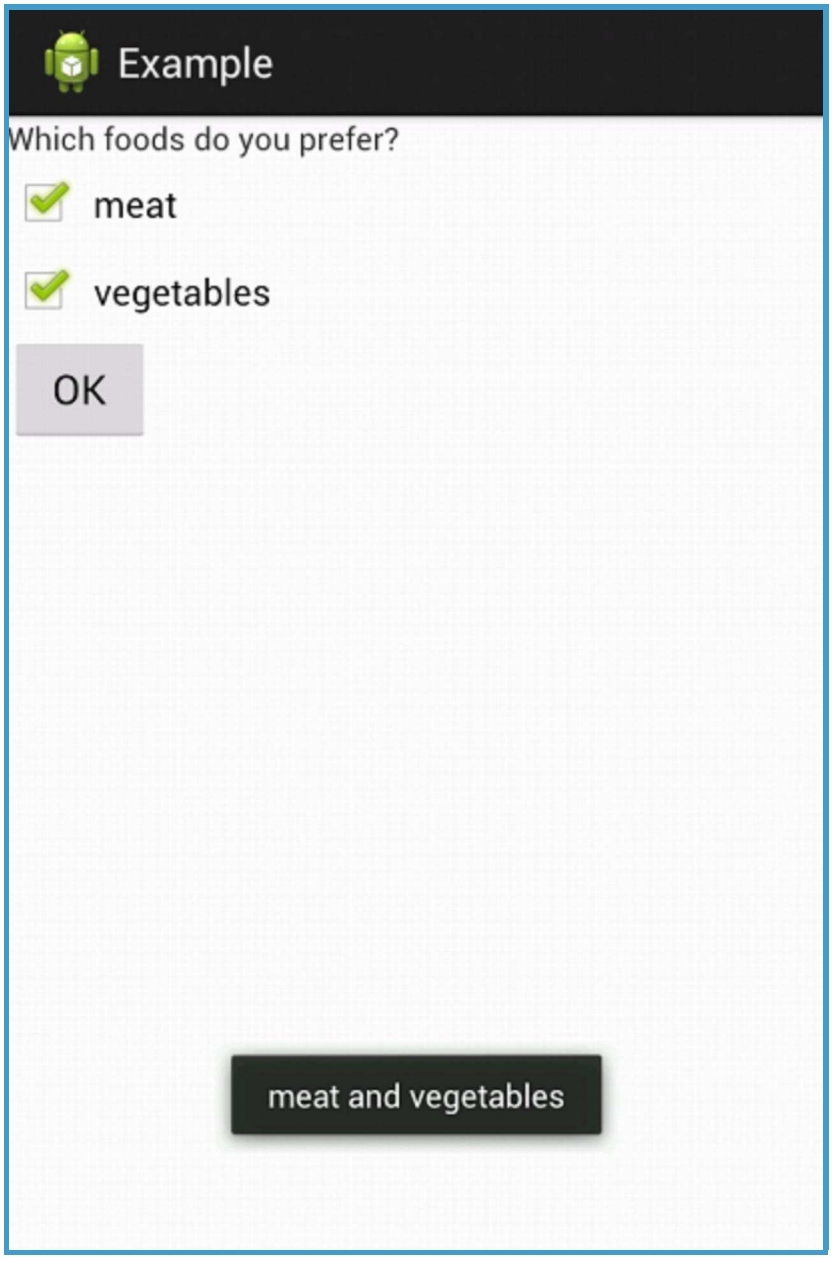
```
;
ɔx;


tends Activity {


les;



ɜundle savedInstanceState) {
InstanceState);
ɔut.activity_main);

‹Box) findViewById(R.id.checkbox_meat);
 (CheckBox) findViewById(R.id.checkbox_vegetables);
findViewById(R.id.button);
istener(new View.OnClickListener() {


ick(View v) {
 = checkboxMeat.isChecked();
tables = checkboxVegetables.isChecked();
ɐgetables)
‹eText(getApplicationContext(), "meat and vegetables", Toast.LENGTH_SHORT).show();
 && !vegetables)
‹eText(getApplicationContext(), "only meat", Toast.LENGTH_SHORT).show();
t && vegetables)
‹eText(getApplicationContext(), "only vegetables", Toast.LENGTH_SHORT).show();

‹eText(getApplicationContext(), "no meat and no vegetables", Toast.LENGTH_SHORT).show();
```

أول ما نقوم به دائما هو التصريح بالمكونات التي نحتاجها و الموجودة في التصميم ، وهي صندوقا الإختيار ثم الزر،وبعد ذلك نعرف كل واحد منها عبر المعرّف id .

عند لمس الزر لابد من الإستجابة للحدث عبر تحديد ما نريده،لكن كيف نعرف ما الذي تم اختياره؟

هناك دالة تجيب عن هذا السؤال،وهي isChecked ، وتأخذ قيمتين إما true أو false ،لذلك قمنا بالتصريح بمتغيرين meat و vegetables من نوع boolean ،كل واحد يأخذ قيمة صحيح أو خطأ (هل تم اختيار الصندوق أو لا)وحسب هذه القيم نظهر النافذة مع المعطيات التي نريد،الكود بسيط،أليس

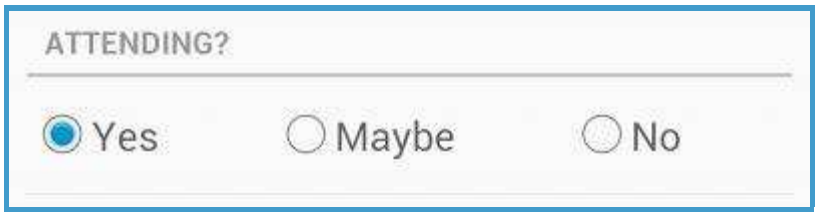
كذلك؟ وهذه هي النتيجة.



هكذا نكون قد وصلنا إلى نهاية الدرس،دمتم في رعاية الله.

زر الإنتقاء-RadioButton

التفاصيل كتب بواسطة: Ahmed Amansag الزيارات: 103



بسم الله الرحمان الرحيم:

السلام عليكم و رحمة الله،نلتقي اليوم أحبتي مع درس جديد مع RadioButton و RadioGroup و تعريف كل واحد منهما.

المكون RadioButton يمكن المستخدم من اختيار **عنصر واحد** من مجموعة معينة(RadioGroup)،

المكون RadioGroup هو المكون الذي يضم أزرار الإختيار RadioButton ،ويمكننا ترصيصها فيه عموديا أو أفقيا.

نعطي هنا مثالا نطلب فيه من المستخدم أن يحدد نوع نظام تشغيل هاتفه ، هذا هو ملف ال xml :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="What is your phone OS?"/>

    <RadioGroup
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/radioGroupOS"
        android:orientation="vertical">
        <RadioButton
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:id="@+id/radioButtonAndroid"
            android:text="Android"/>
        <RadioButton
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:id="@+id/radioButtonIOS"
            android:text="iOS"/>
        <RadioButton
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:id="@+id/radioButtonBada"
            android:text="Bada"/>

    </RadioGroup>



    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="OK" />
```

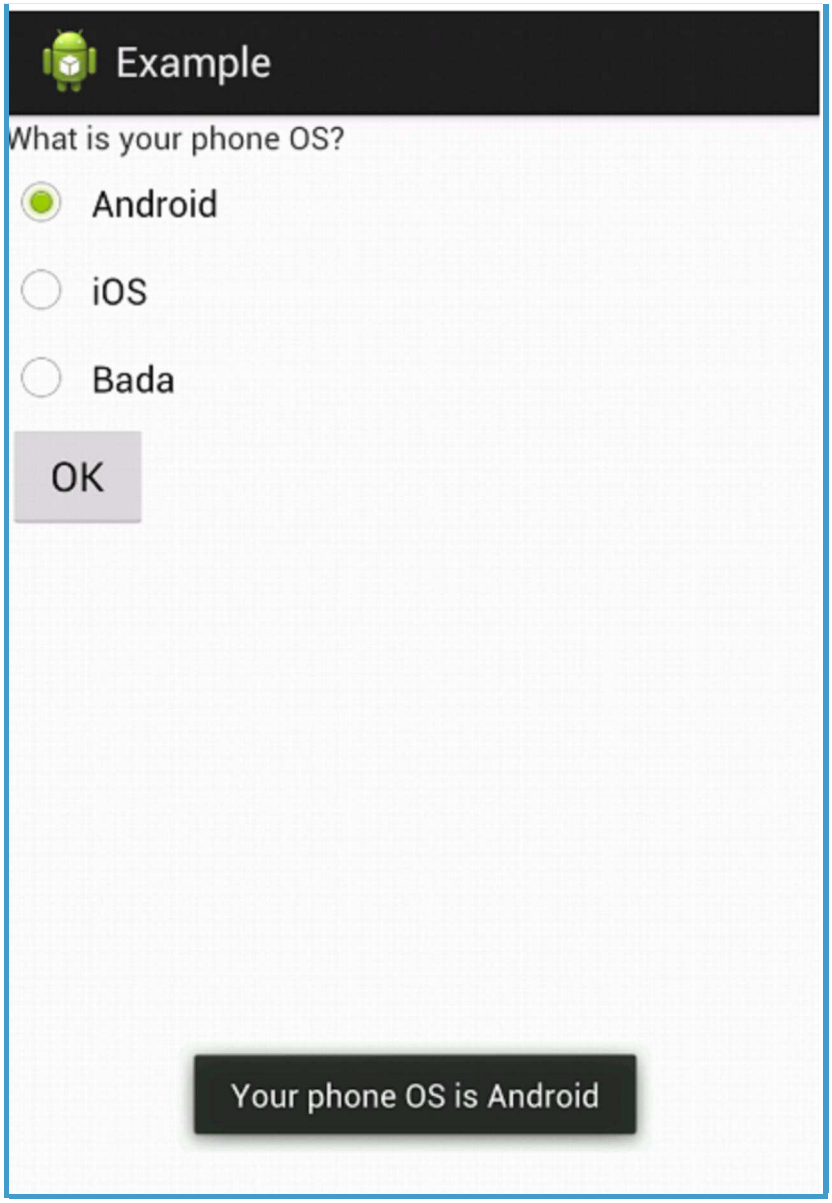
```
41
42  </LinearLayout>
```

المكون RadioGroup يضم أزرار الإختيار التي أنشأناها وعرفناها،وهو كذلك معرف بمعرّف سنستعمله برمجيا.الخاصيات ليس هناك خاصية تستدعي شرحها في هذا الملف.ننتقل إلى كود الكلاس

```
;                                                                    ?

;


ı;
3ox;
3utton;
3roup;
;

xtends Activity {

;



(Bundle savedInstanceState) {
dInstanceState);
yout.activity_main);
ioGroup) findViewById(R.id.radioGroupOS);

 findViewById(R.id.button);
Listener(new View.OnClickListener() {


lick(View v) {
edRadio = radiogroupOS.getCheckedRadioButtonId();
eckedRadio){
adioButtonAndroid:
akeText(getApplicationContext(), "Your phone OS is Android", Toast.LENGTH_SHORT).show();

adioButtonIOS:
akeText(getApplicationContext(), "Your phone OS is iOS", Toast.LENGTH_SHORT).show();

adioButtonBada:
akeText(getApplicationContext(), "Your phone OS is Bada", Toast.LENGTH_SHORT).show();
```

قمنا بالتصريح بالمتغير radiogroupOS و قمنا بتعريفه أيضا من خلال الidالموجود لدينا في التصميم،هذا المكون يوفر لنا دالة من خلالها يمكن معرفة id الزر الذي تم اختياره و هي getCheckedRadioButtonId و مُخرج هذه الدالة عدد من نوع int ،يكفي مقارنة هذا العدد بidالأزرار الأخرى،إذن يكفي أن نعرف متغير من نوع int و نعطيه القيمة radiogroupOS.getCheckedRadioButtonId() ونقارنها ب id الأزرار الثلاثة

وها هي النتيجة:



دمتم في رعاية الله