
Bài thực hành số 1 – Những khái niệm cơ bản về Android

Chú ý chung với toàn bộ các buổi lab:

- Toàn bộ bài tập trong các buổi lab, bắt buộc sinh viên phải viết mã bằng tay, sử dụng một trong các chương trình soạn thảo mã sau:
 - o Eclipse

Mục tiêu

Buổi thực hành số 1 tiếp tục giúp sinh viên làm quen với IDE lập trình Android là Eclipse và xây dựng một số chương trình cơ bản

Bài 1 (làm ở nhà) – Cài đặt được Java JDK, Android SDK, ADT Plugin for Eclipse, update Android SDK

Tải và cài đặt Java JDK: <http://java.sun.com/javase/downloads/index.jsp> (cài đặt cái này trước, nên chọn bản mới nhất)

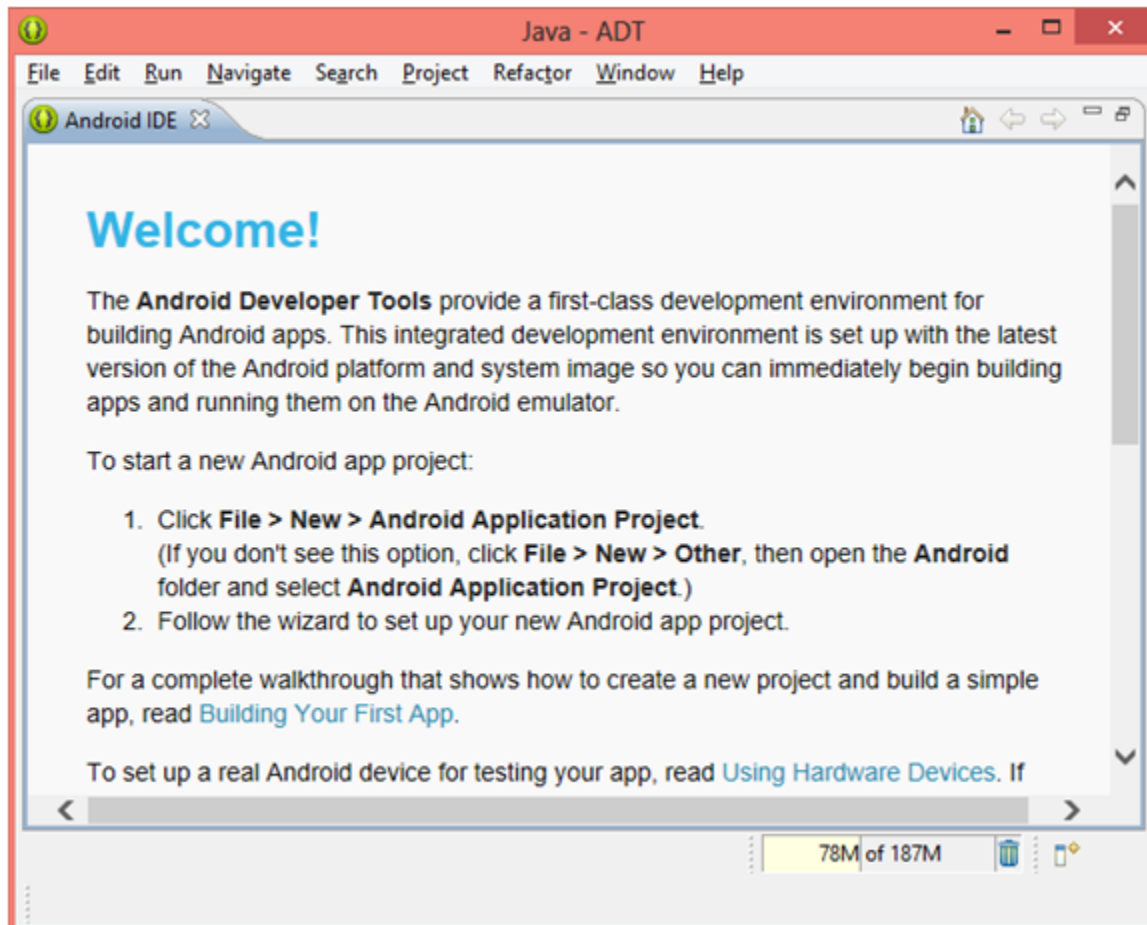
- Tải và cài đặt Android SDK: <http://developer.android.com/sdk/index.html> , tải gói **ADT Bundle for Windows**, gói này sẽ chứa các thành phần:

- ☐ Eclipse + ADT plugin
- ☐ Android SDK Tools
- ☐ Android Platform tools
- ☐ Android API mới nhất

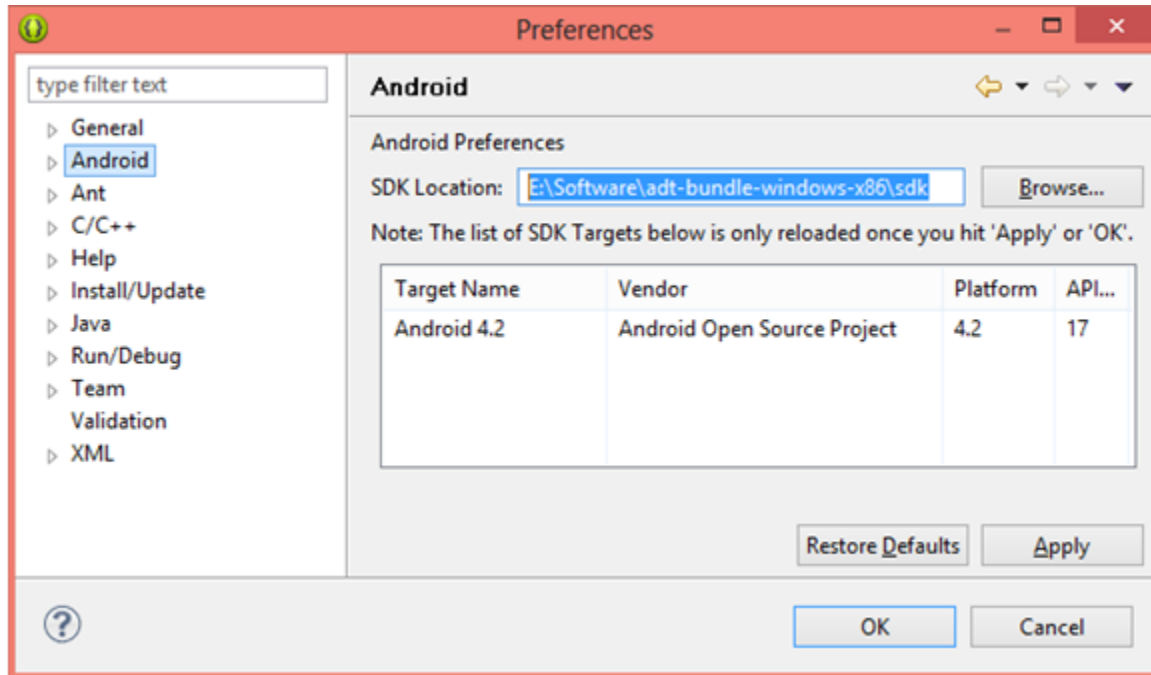
Android system image mới nhất cho Emulator

- Cập nhật ADT plugin mới nhất cho Eclipse:
 1. Vào menu Help / chọn Install New Software
 2. Nhập đường dẫn: <https://dl-ssl.google.com/android/eclipse/> và bấm nút Add ở góc bên phải
 3. Chọn các thông số thích hợp để tiến hành cập nhật.
- Tiến hành Cập nhật Android SDK (nếu cần thiết)
- Tham chiếu Android SDK (bắt buộc, nếu cài gói ADT Bundle for Windows trên trang developer.android.com thì nó đã tự tham chiếu cho chúng ta rồi).

Sau khi cài đặt xong đầy đủ (JDK, Android SDK ...), ta khởi động Elipse trong gói **ADT Bundle for windows**, ta có giao diện như hình bên dưới:



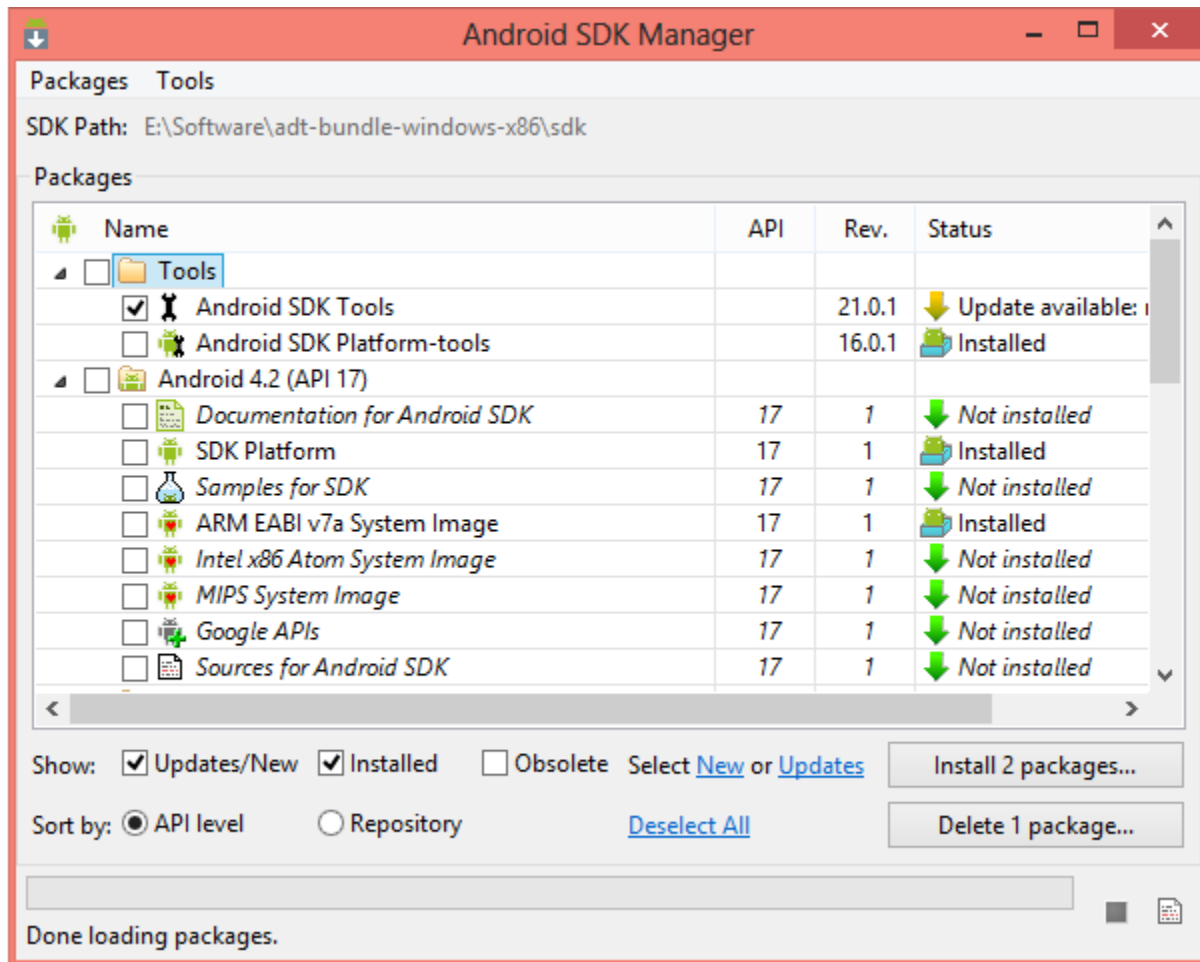
- Đóng màn hình Welcome bằng cách click vào dấu chéo kế bên chữ **Android IDE**
- Để tham chiếu tới Android SDK, ta vào menu **Windows/ Preferences/ chọn Android** và Browse tới SDK location mà bạn lưu trữ:



Các bạn nhìn vào hình trên, thấy có Android 4.2, API 17 hiển thị ra. Đó chính là API mới nhất của Android, tùy vào danh sách API được tải về máy nó sẽ hiển thị ra đây. Hiện tại trong máy của Tôi chỉ có 1 API 17, nếu máy bạn có nhiều API khác thì nó sẽ liệt kê ra đây hết.

- Nếu như máy tính của bạn có kết nối internet, thì bạn có thể cập nhật API theo cách sau:

Vào menu **Windows/ Android SDK Manager** (hoặc bấm vào biểu tượng trên thanh toolbar), màn hình Android SDK Manager sẽ hiện thị ra như bên dưới:



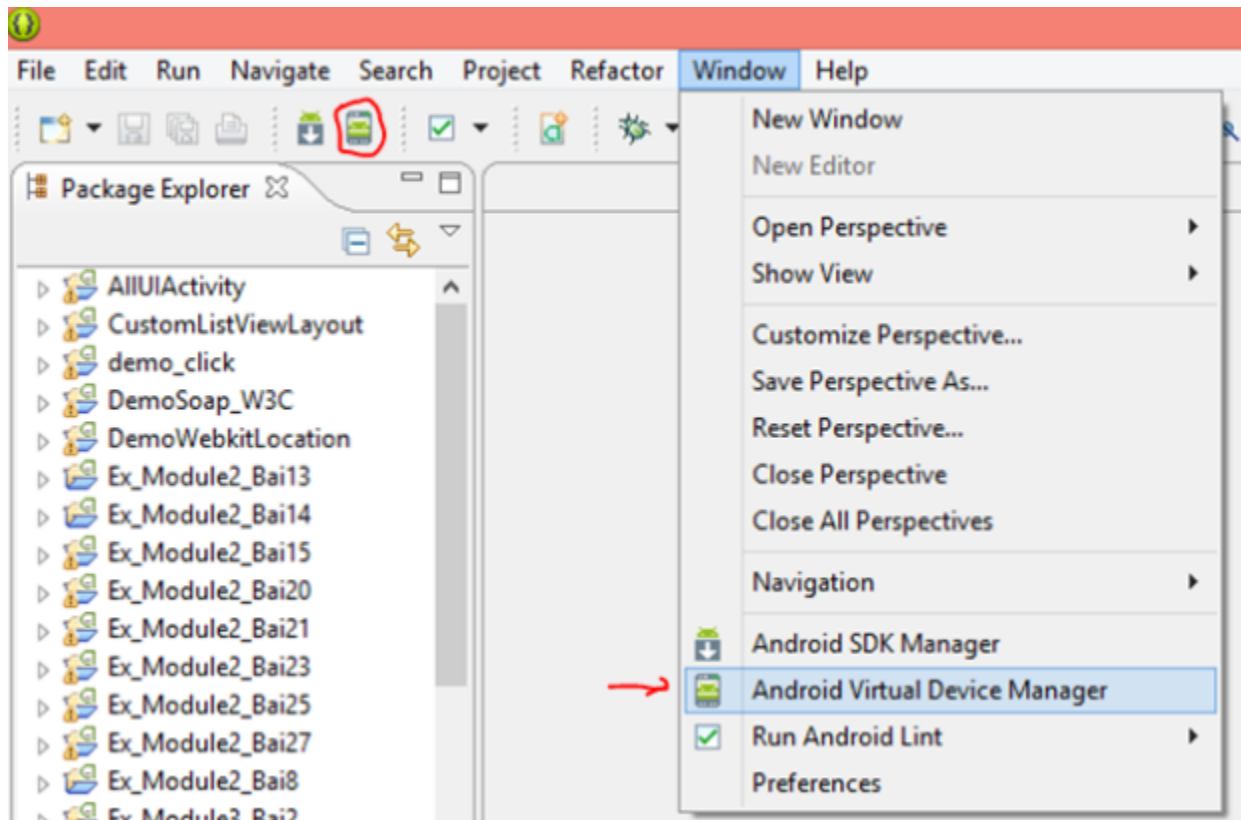
Bạn quan sát trong màn hình bên trên, Những gói nào chưa được cài đặt thì có dòng chữ “Not installed”, bạn checked vào nó và nhấn nút Install packages.. ở góc phải dưới cùng.

Tới đây là bạn đã biết cài đặt JDK, cài đặt Android SDK, cách chạy Eclipse và cập nhật API.

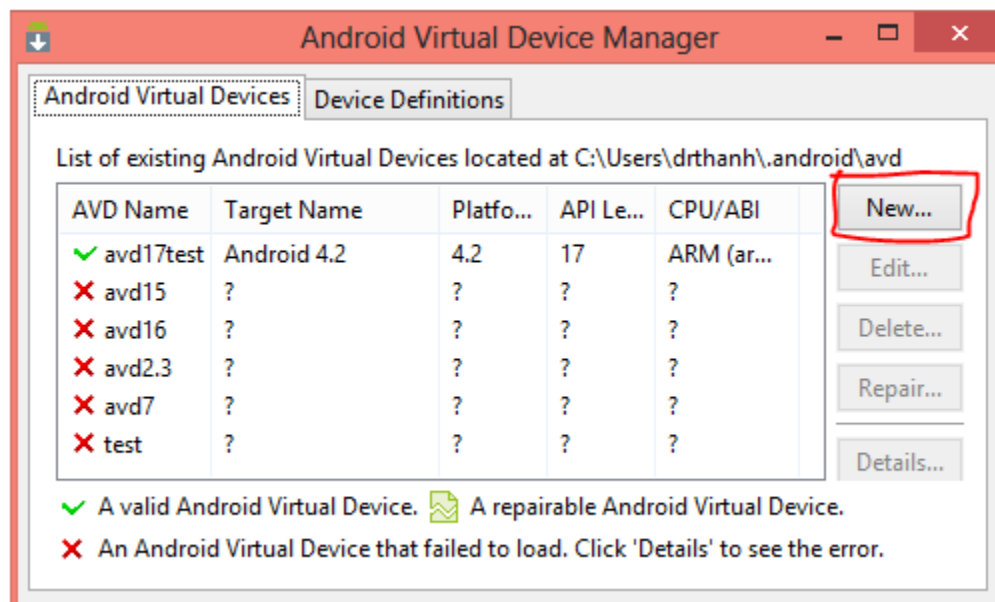
Bài 2 (thời gian thực hiện: 30 phút) – Tạo Emulator

1) Cách tạo Android emulator:

- Vào menu Windows/ chọn Android Virtual Device Manager hoặc click vào biểu tượng máy ảo trên thanh Toolbar:

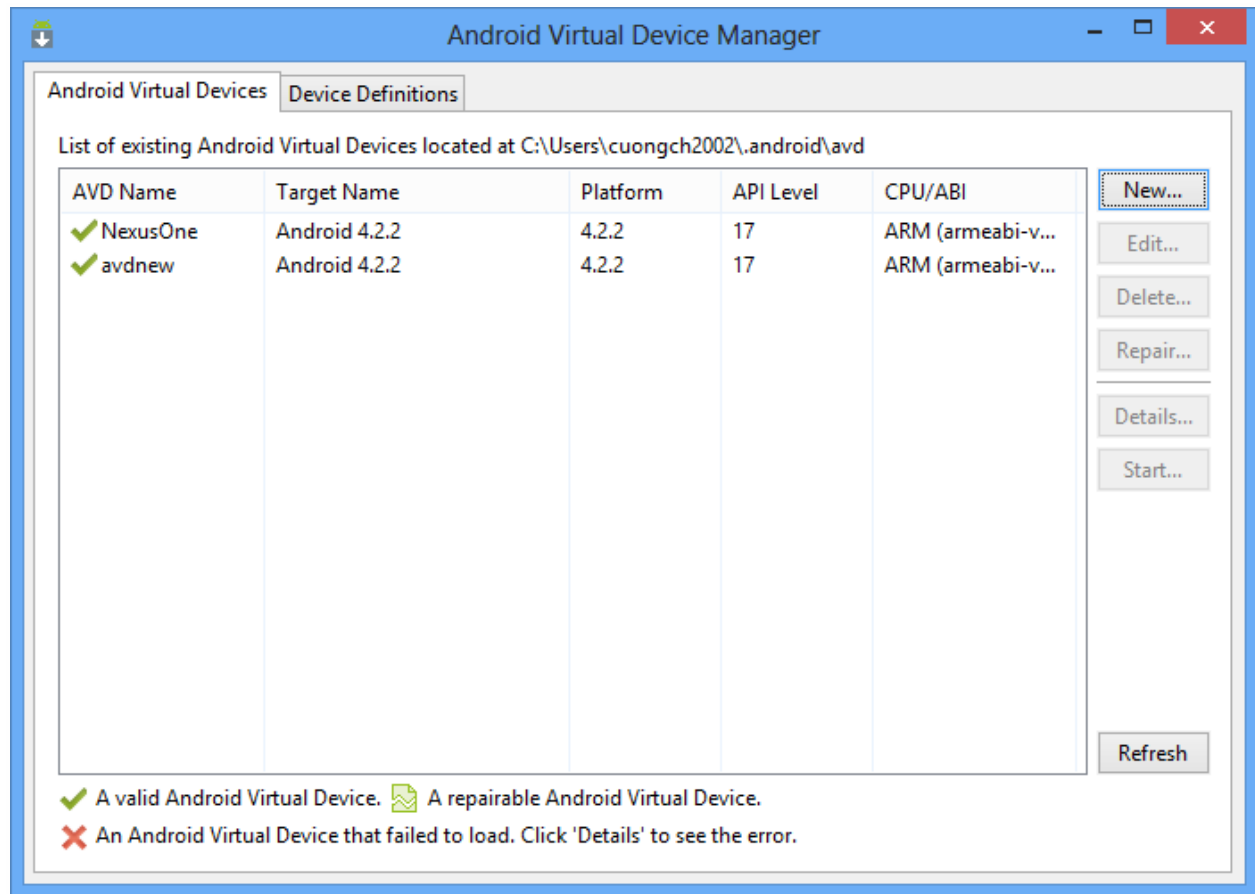


- Cửa sổ **Android Virtual Device Manager** sẽ hiển thị ra như bên dưới:

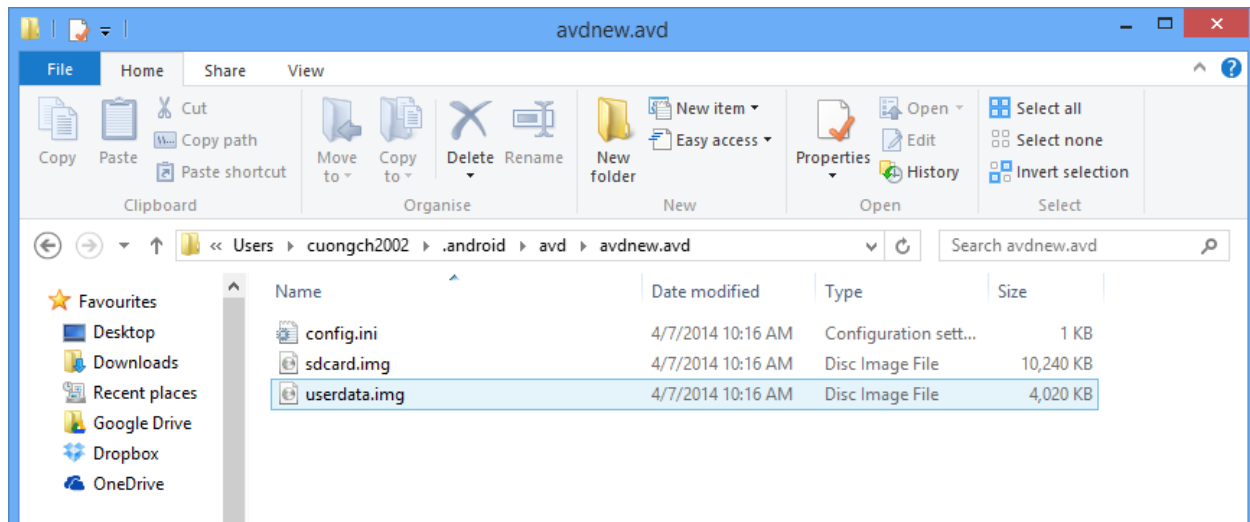


- Để tạo một Android Emulator : click nút New (khoanh vùng màu đỏ):

- Ở màn hình bên trên, ta thiết lập các thông số: Tên máy ảo, độ phân giải, API, bộ nhớ trong, bộ nhớ ngoài (SD Card) rồi nhấn nút OK. Muốn lần tiếp theo khởi động máy ảo nhanh thì ta checked vào mục **Snapshot**



- Như bạn thấy máy ảo **avdnew** đã được tạo ra, bạn chú ý là khi một máy ảo được tạo ra thì nó sẽ được lưu trữ vào user của máy tính đang hoạt động, ở hình trên thì lưu trong “**c:\Users\cuongch2002\.android\avd**“. Máy của bạn cài đặt với user nào thì bạn vào đúng chỗ mà xem cấu trúc bên trong:

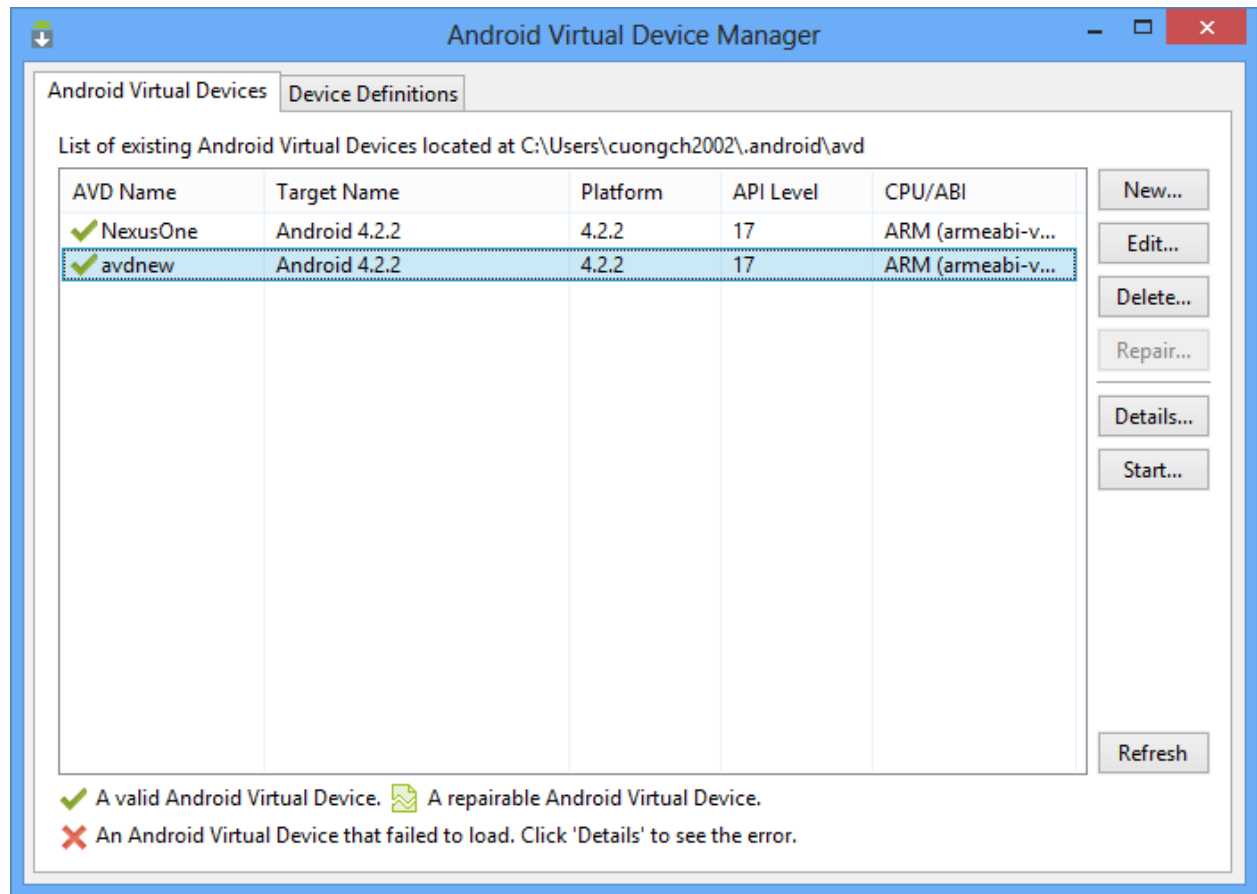


- Ở đây bạn chú ý là nếu bạn cấp dung lượng cho SD Card tại màn hình tạo máy ảo bao nhiêu thì tập tin **sdcard.img** sẽ có dung lượng bấy nhiêu trong này (do đó bạn phải cẩn thận khi tạo SD Card đối với máy tính có dung lượng ổ cứng khiêm tốn)

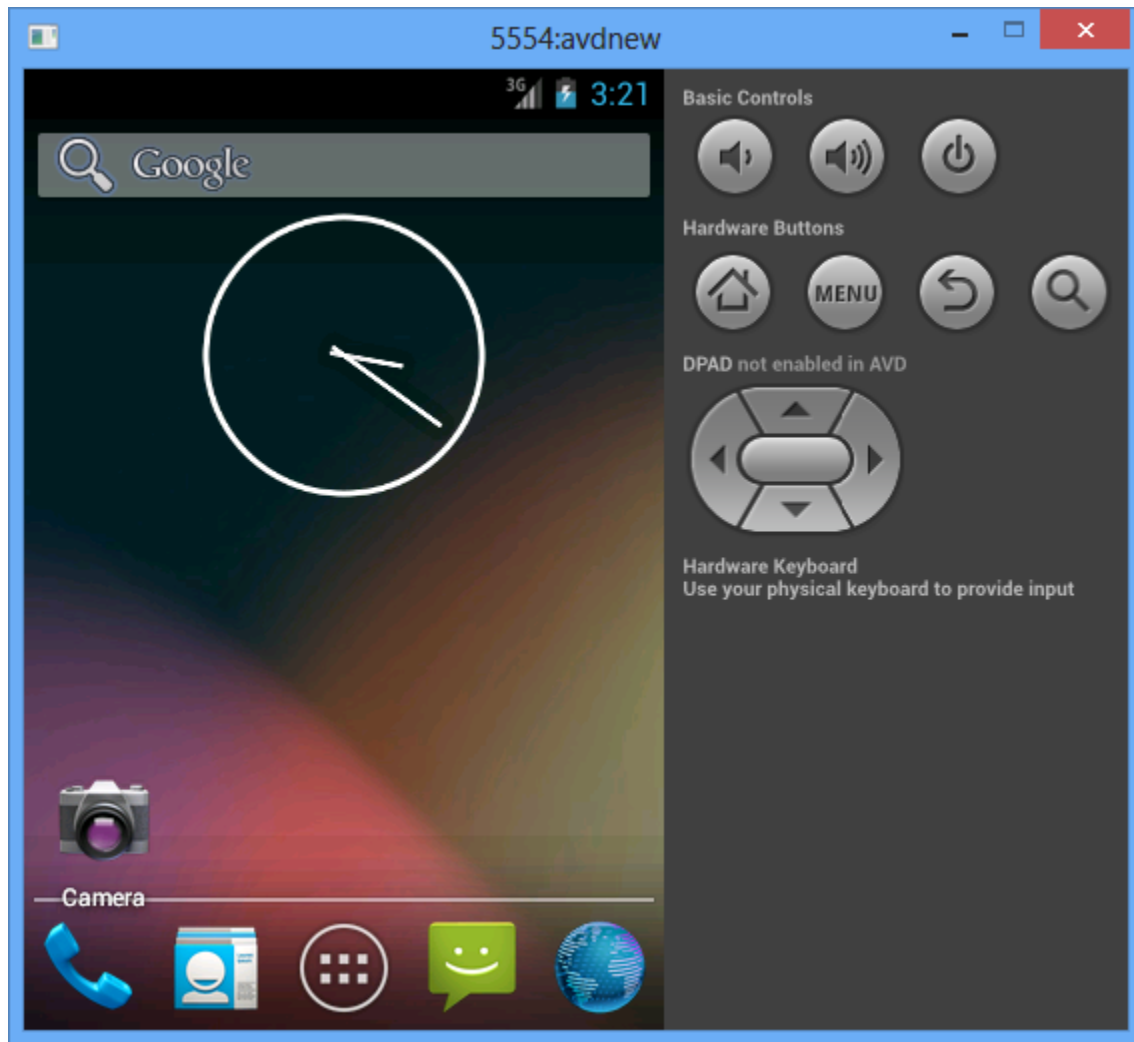
- Tương tự bạn nhìn thấy tập tin **snapshots.img**, hiện giờ bạn chỉ thấy có 250kb, nhưng nếu như bạn khởi động lần đầu tiên và vào đây quan sát thì dung lượng của nó sẽ tăng lên rất lớn, mục đích để lưu trữ lại toàn bộ thông số để lần khởi chạy thứ 2 cho nhanh

2) Chạy Android emulator:

- Từ màn hình Android Virtual Device Manager, ta chọn tên máy ảo cần chạy rồi nhấn nút Start:



- Bạn chờ khoảng một thời gian vài phút (hoặc lâu hơn tùy vào khả năng của máy). Tùy vào cấu hình mà bạn chọn (API, độ phân giải...), Android Emulator sẽ xuất hiện như bên dưới:

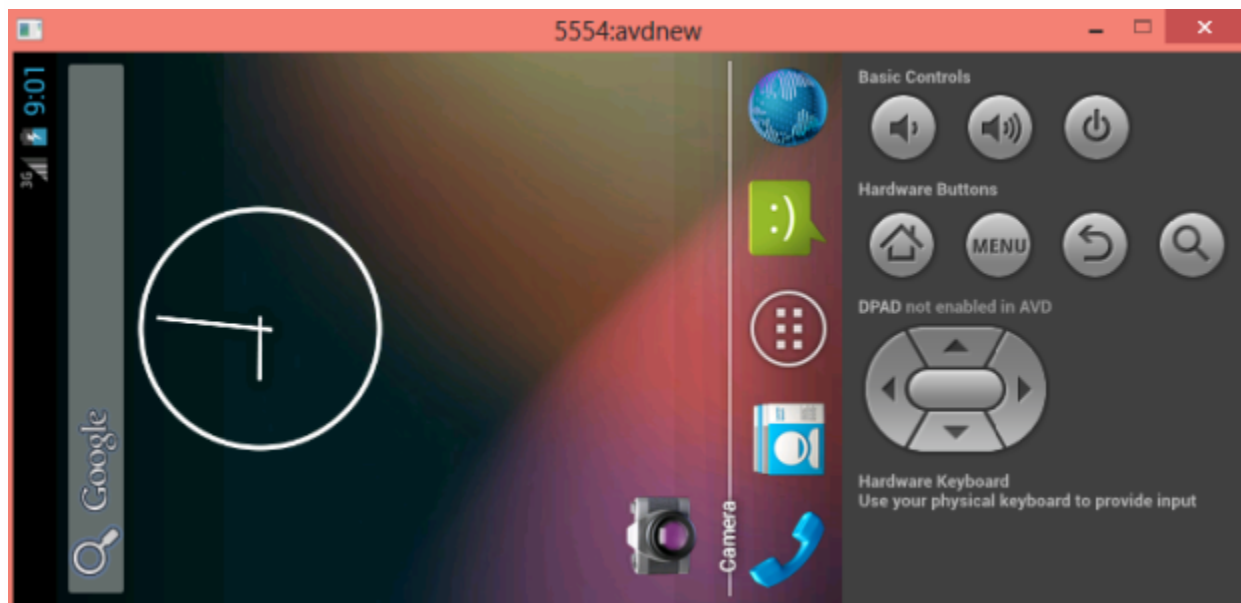


- Khi máy ảo ở trên khởi động xong, bạn vào thư mục trong ổ C: nơi mà lưu trữ máy ảo, bạn quan sát xem có những thư mục và tập tin mới nào được tạo ra. Sau đó bạn tắt máy ảo này đi, tiếp tục quan sát bạn sẽ thấy tập tin **Snapshots.img** đề cập ở trên sẽ có dung lượng rất lớn:

(C:) > Users > drthanh > .android > avd > avdnew.avd				
Name	Date modified	Type	Size	
cache.img	22/03/2013 3:54 PM	Disc Image File	9 KB	
config.ini	22/03/2013 3:36 PM	Configuration sett...	1 KB	
emulator-user.ini	22/03/2013 3:54 PM	Configuration sett...	1 KB	
hardware-qemu.ini	22/03/2013 3:50 PM	Configuration sett...	2 KB	
sdcard.img	22/03/2013 3:54 PM	Disc Image File	10,240 KB	
snapshots.img	22/03/2013 3:54 PM	Disc Image File	426,561 KB	
snapshots.img.default-boot.ini	22/03/2013 3:54 PM	Configuration sett...	2 KB	
userdata.img	22/03/2013 3:36 PM	Disc Image File	4,018 KB	
userdata-qemu.img	22/03/2013 3:54 PM	Disc Image File	14,360 KB	

3) Cách sử dụng Android Emulator:

- Bạn để ý là mỗi một máy ảo được khởi động nó sẽ đi với một port nào đó, ví dụ trong trường hợp ở trên là port **5554** (xem ở tiêu đề của cửa sổ). Bạn chú ý là port này chính là đại diện cho số điện thoại của máy ảo đó. Ta sẽ dựa vào port này để thao tác : Gửi tin nhắn, gọi điện thoại.... trong của sổ DDMS.

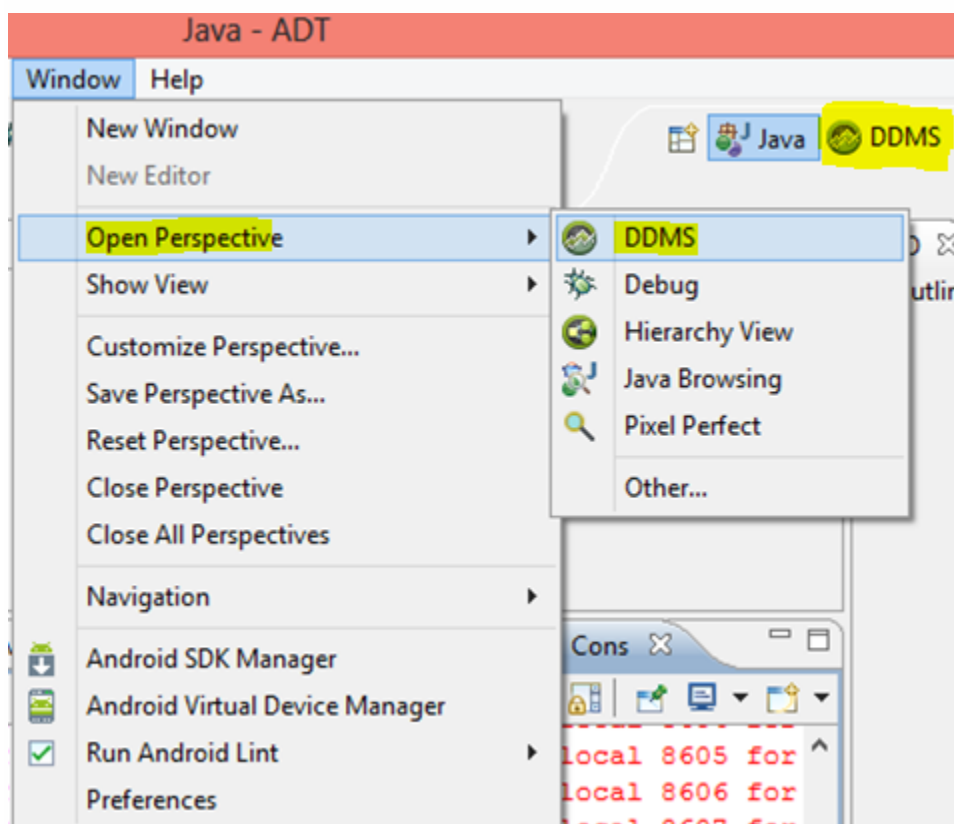


- Cách gửi tin nhắn – gọi điện thoại trong máy ảo:

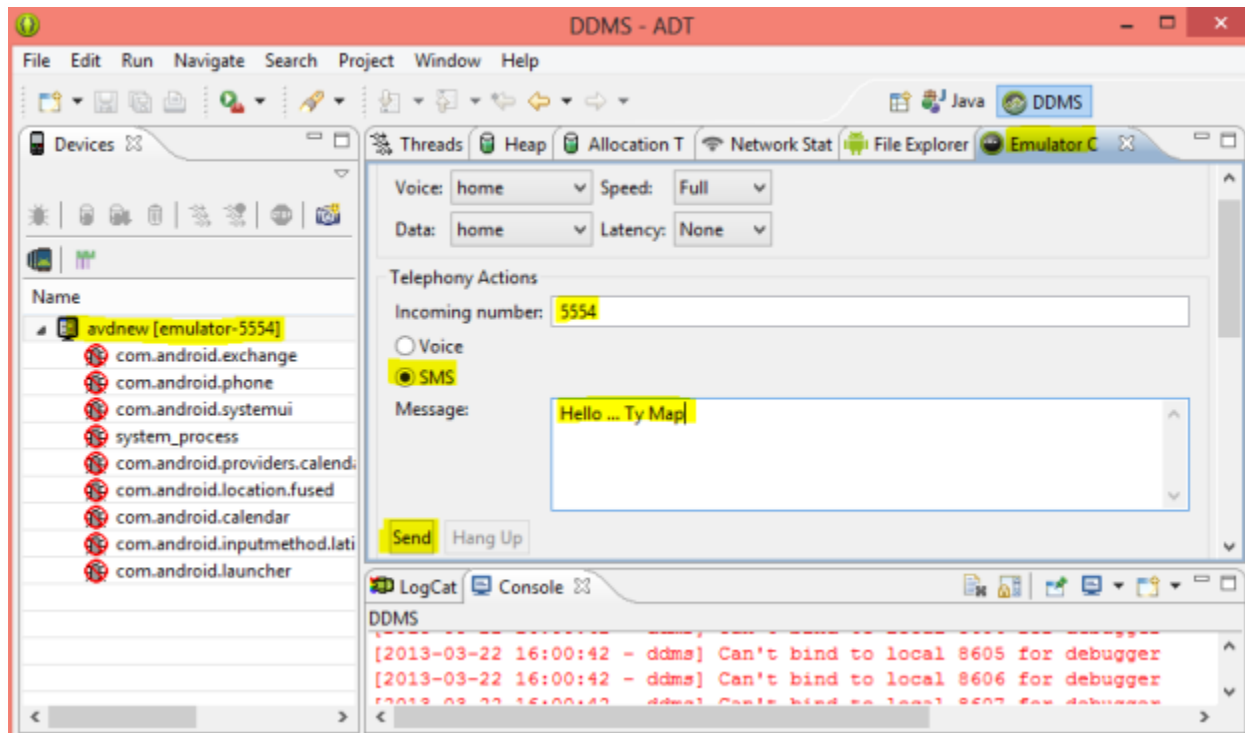
+ **Cách 1:** Bạn tạo thêm một máy ảo Android khác, khi một máy thứ 2 được khởi chạy thì thông thường port của nó sẽ là **5556**, Như bên trên Tôi đã nói Port này là đại diện cho số điện thoại. Máy tính của bạn hiện tại sẽ có 2 máy ảo Android (Bạn tưởng tượng đó là 2 cái điện thoại thật mà bạn đang cầm trên tay đi). Do đó cách mà bạn thao tác với điện thoại thật như thế nào thì với Emulator nó sẽ y xì như vậy.

+ **Cách 2:** Trên thanh công cụ bạn quan sát thấy nút **DDMS**, nhấn vào nó (chú ý là bạn phải ít nhất đang chạy 1 máy ảo android)

Thông thường thì trên thanh công cụ sẽ có DDMS, nhưng nếu như không thấy thì bạn vào menu **Windows/ chọn Open Perspective/ DDMS** , xem hình dưới:



- Sau khi chọn DDMS, bạn vào tab Emulator Control như hình bên dưới:



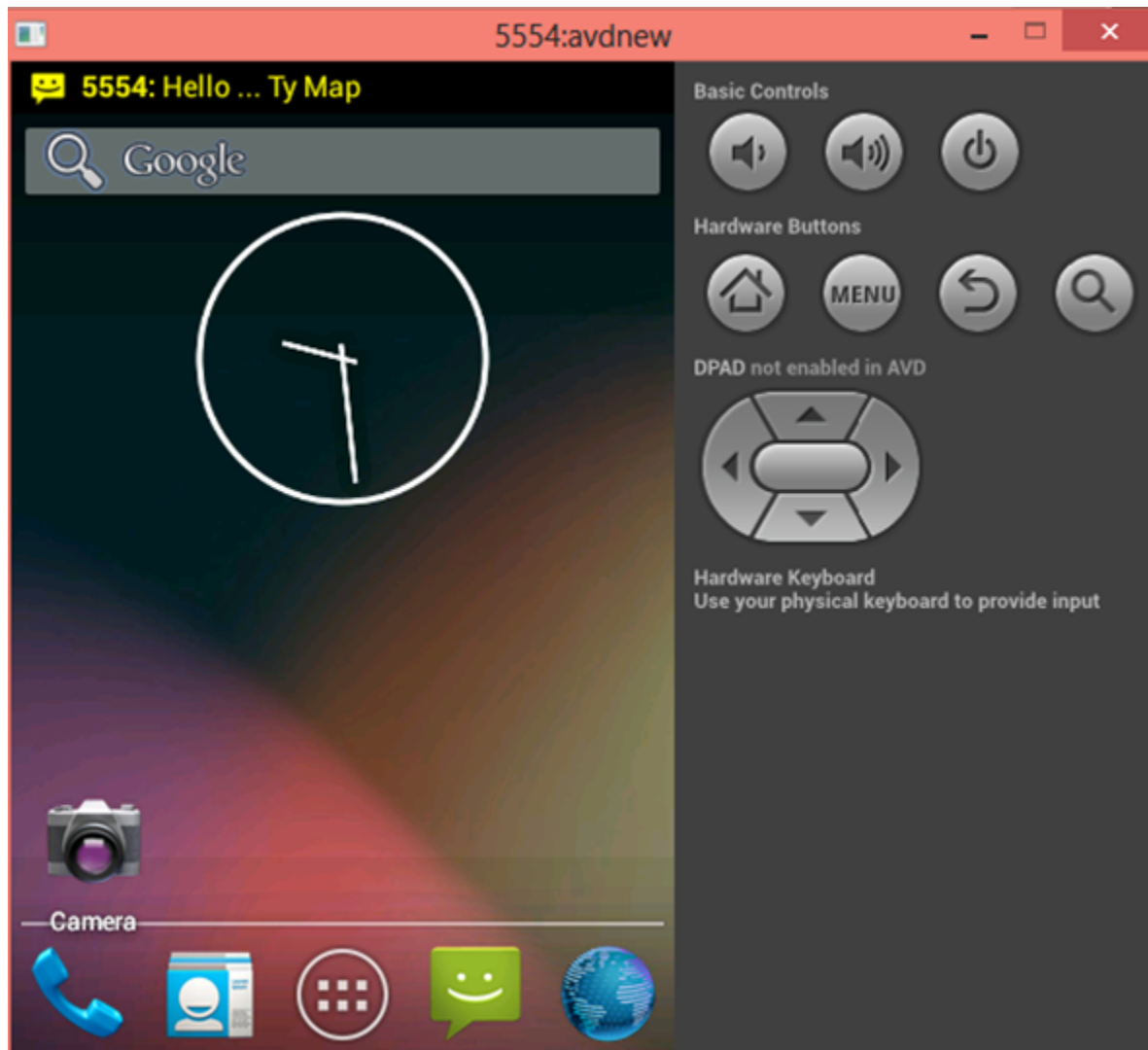
- Ở màn hình trên:

+ Mục Incoming Number Tôi nhập là 5554 (port này chính là số điện thoại của máy ảo)

+ Tôi chọn SMS và nhập vào “Hello ... Ty Map”

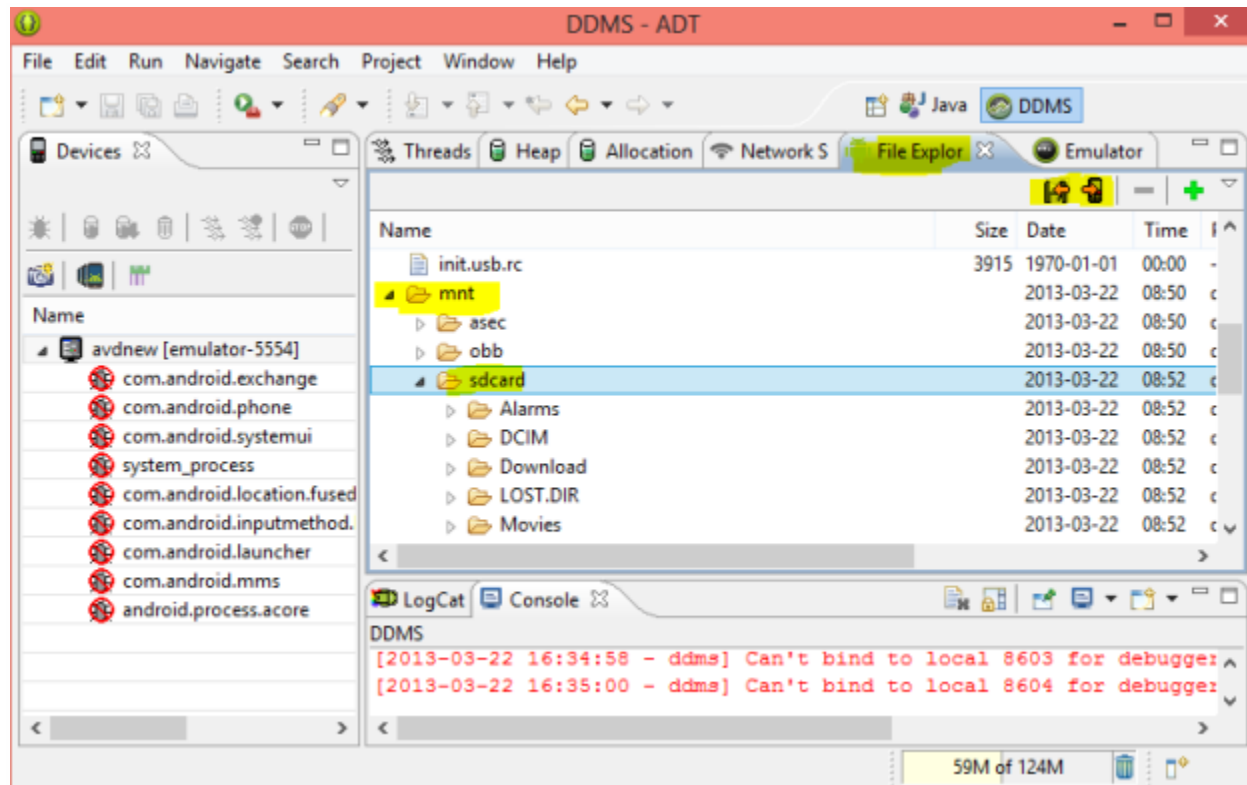
+ Nhấn nút Send

→Bạn quan sát máy Ảo có port 5554 sẽ nhận được tin nhắn này:



- Cách sử dụng Profile Explorer:

+ Trong màn hình DDMS, bạn chọn tab File Explorer: Trong bài tập này bạn chỉ cần tập cách thả 1 tập tin từ máy tính vào SD Card, và kéo 1 tập tin từ SD Card ra máy tính (Chúng ta còn phải làm việc với nó rất nhiều lần ở các bài tập kế tiếp):



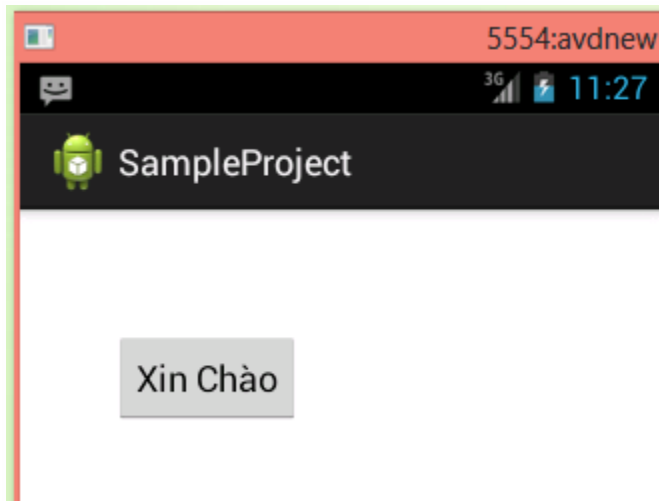
- Bạn quan sát thư mục **mnt/ sdcard** . Chú ý là có một vài nơi cùng tên sdcard, nhưng bạn phải vào **mnt**.

+ Để lấy 1 tập tin từ SD Card ra thì bạn chọn tập tin đó rồi nhấn vào biểu tượng ở đĩa mềm ở bên trên (Tô tô màu vàng)

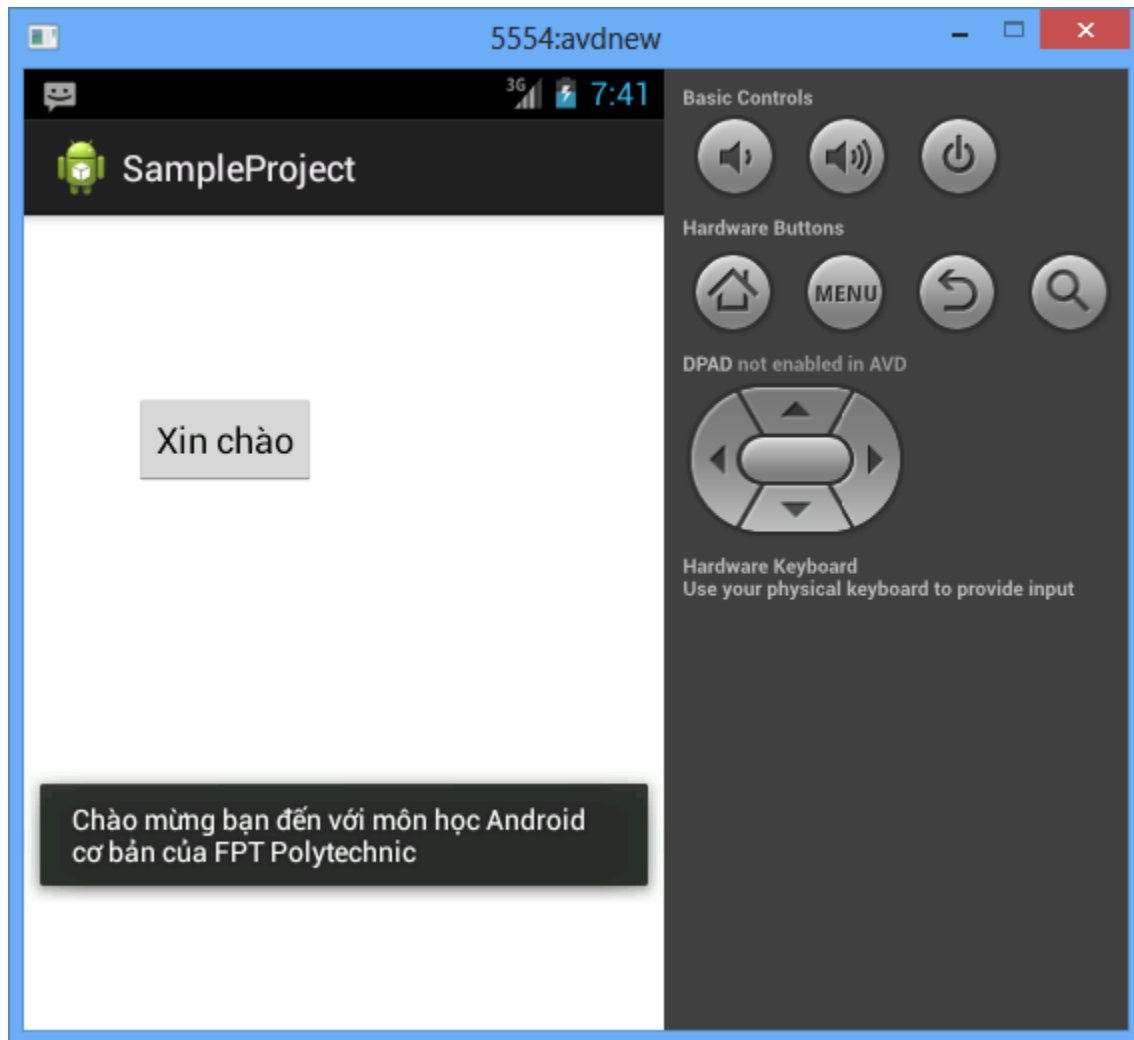
+ Để đưa một tập tin từ Máy tính vào SD card bạn nhấn vào biểu tượng điện thoại (kế bên biểu tượng đĩa mềm), chọn tập tin cần đưa, Hoặc bạn kéo thả trực tiếp một tập tin nào đó vào màn hình này luôn. cả 2 cách đều như nhau.

Bài 3 (thời gian thực hiện: 1h30) – Tạo mới Android Project

Bạn xây dựng chương trình có giao diện như sau:



Khi click vào nút Xin chào, chương trình sẽ hiển thị câu thông báo **“Chào mừng bạn đến với môn học Android cơ bản của FPT Polytechnic”**



- Trong bài tập này các bạn sẽ thực hành cách tạo Android Project và tìm hiểu các thành phần bên trong của nó:

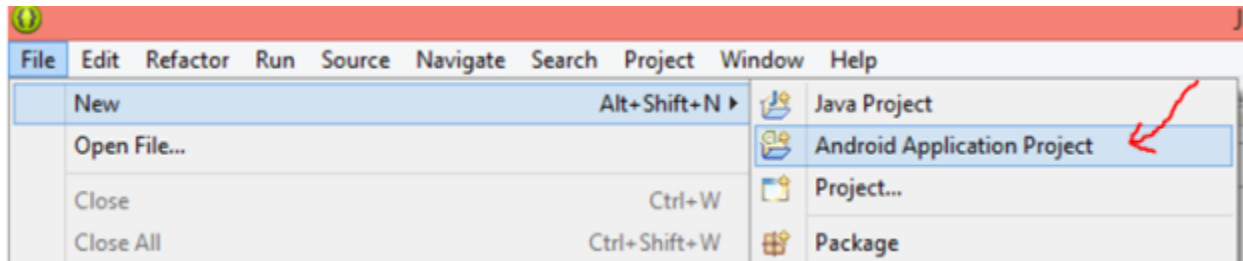
- ☐ Cách tạo Android Project
- ☐ Activity, Intent, View
- ☐ Auto gen
- ☐ Android libs
- ☐ Resource
- ☐ Layout, menu, values
- ☐ Manifest XML

- Cũng như cách chạy chương trình Android, hiểu được cơ chế vận hành của nó.

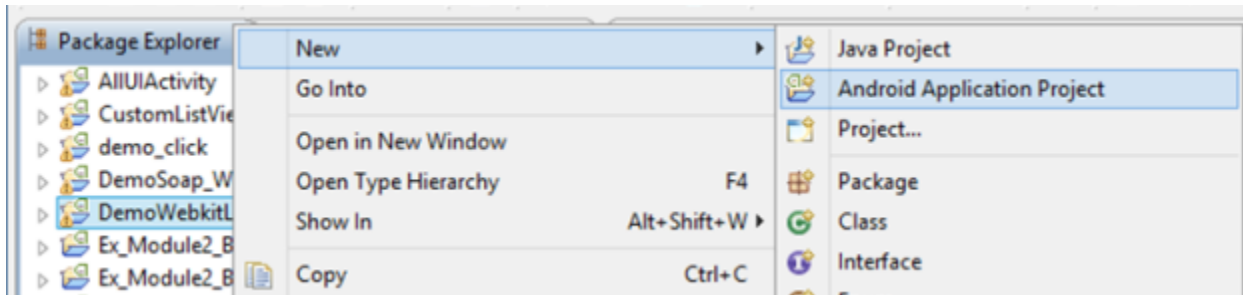
1) Cách tạo một Android Project:

Ở đây có 2 cách tạo Android Project:

Cách 1: Bấm vào Menu File/ Chọn New/ chọn Android Project (xem hình bên dưới):



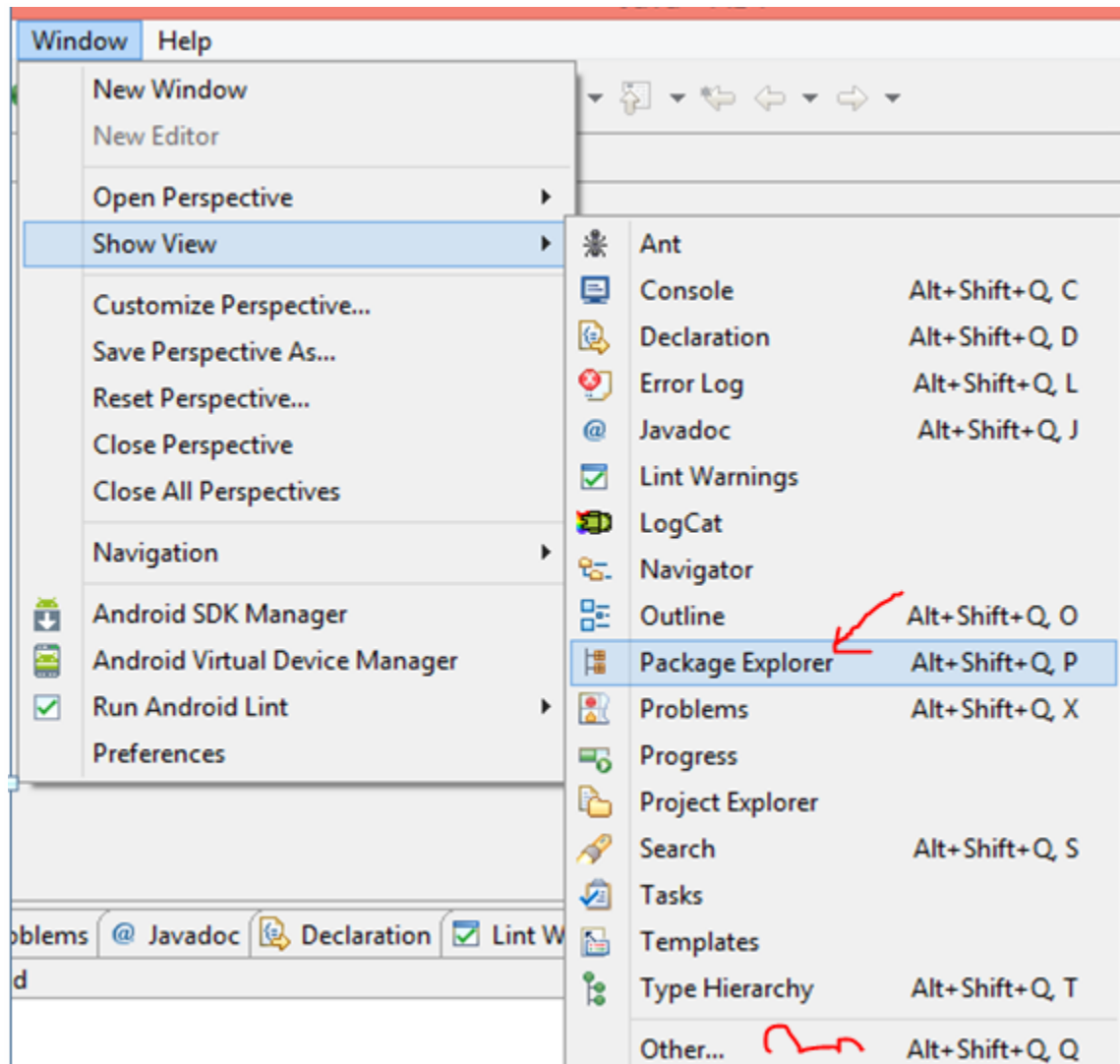
Cách 2: Bấm chuột phải vào vị trí bất kỳ trong Package Explorer / Chọn New / Chọn Android Project (xem hình bên dưới):



Các bạn chú ý là khi một Android Project được tạo ra thì nó sẽ được lưu trữ trong Package Explorer. Một số trường hợp bạn không thể thấy được Package Explorer (do bạn lỡ tay đóng nó đi, hoặc một nguyên nhân nào đó), nếu như chưa biết cách lôi nó ra thì các bạn làm như sau:

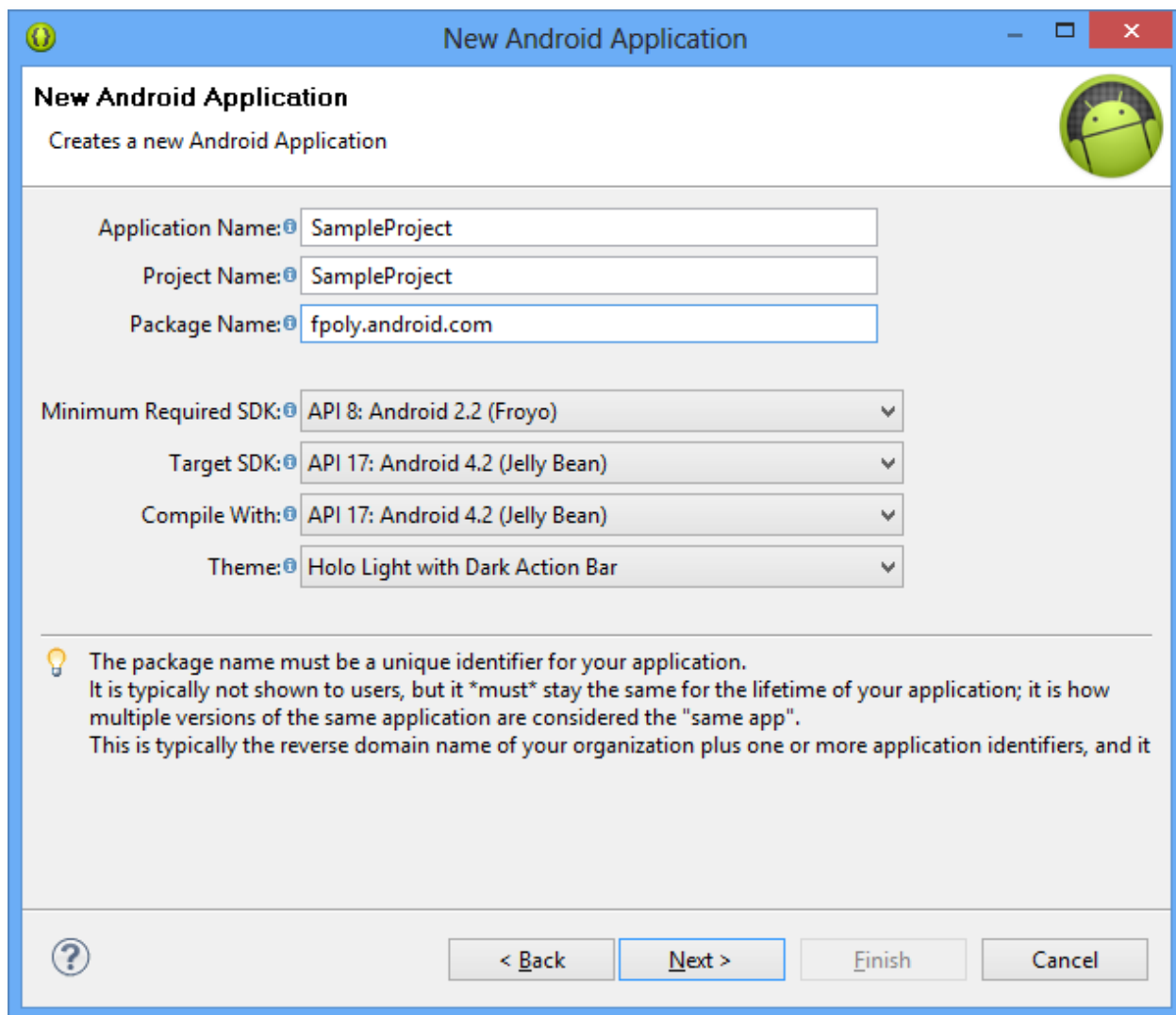
- Vào menu Windows/ chọn Show View/ click chọn **Package Explorer**

- Nếu như trong Show View mà không thấy Package Explorer thì bạn nhìn mục dưới cùng có nhãn “**Other...**“, click vào nó thì chắc chắn bên trong sẽ có Package Explorer:



2) Nhập thông số cho một Android Project mới:

Khi bạn chọn New Android Project thì một màn hình sẽ hiển thị ra như bên dưới:



New Android Application
Creates a new Android Application

Application Name:

Project Name:


Package Name:


Minimum Required SDK:

Target SDK:

Compile With:

Theme:

 The package name must be a unique identifier for your application. It is typically not shown to users, but it **must** stay the same for the lifetime of your application; it is how multiple versions of the same application are considered the "same app". This is typically the reverse domain name of your organization plus one or more application identifiers, and it



Mục **Application Name**: bạn đặt tên cho ứng dụng mà bạn mong muốn, trong ví dụ này đặt là “SampleProject”

Mục **Project Name** : thông thường khi bạn đặt tên cho Application Name thì mục Project Name sẽ tự động cập nhật giống như vậy

Mục **Package Name**: bạn nên viết thường hết và phải ít nhất có 1 dấu chấm ngăn cách, ví dụ bạn có thể đặt “fpoly.android.com” nhưng không thể đặt “fpoly”.

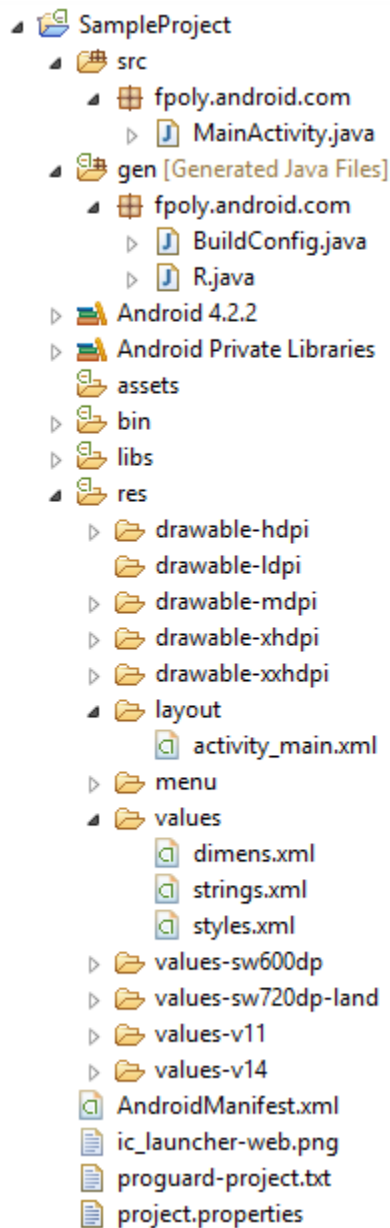
Mục **Minimum Required SDK** : Chọn giới hạn API thấp nhất mà ứng dụng có thể cài đặt (Điện thoại đó có API phải \geq Minimum Require)

Mục **Target SDK** : chọn API mà ứng dụng mong muốn tốt nhất có thể thực thi trên API này

Bạn cần phân biệt giữa Minimum và Target:

- Sau khi chọn các thông số xong, bạn nhấn **Next** cho tới khi nút **Finish** xuất hiện
- Khi bấm Finish thì bạn quan sát Package Explorer để xem cấu trúc bên trong của ứng dụng Android:

3) Tìm hiểu cấu trúc bên trong của Ứng dụng Android:



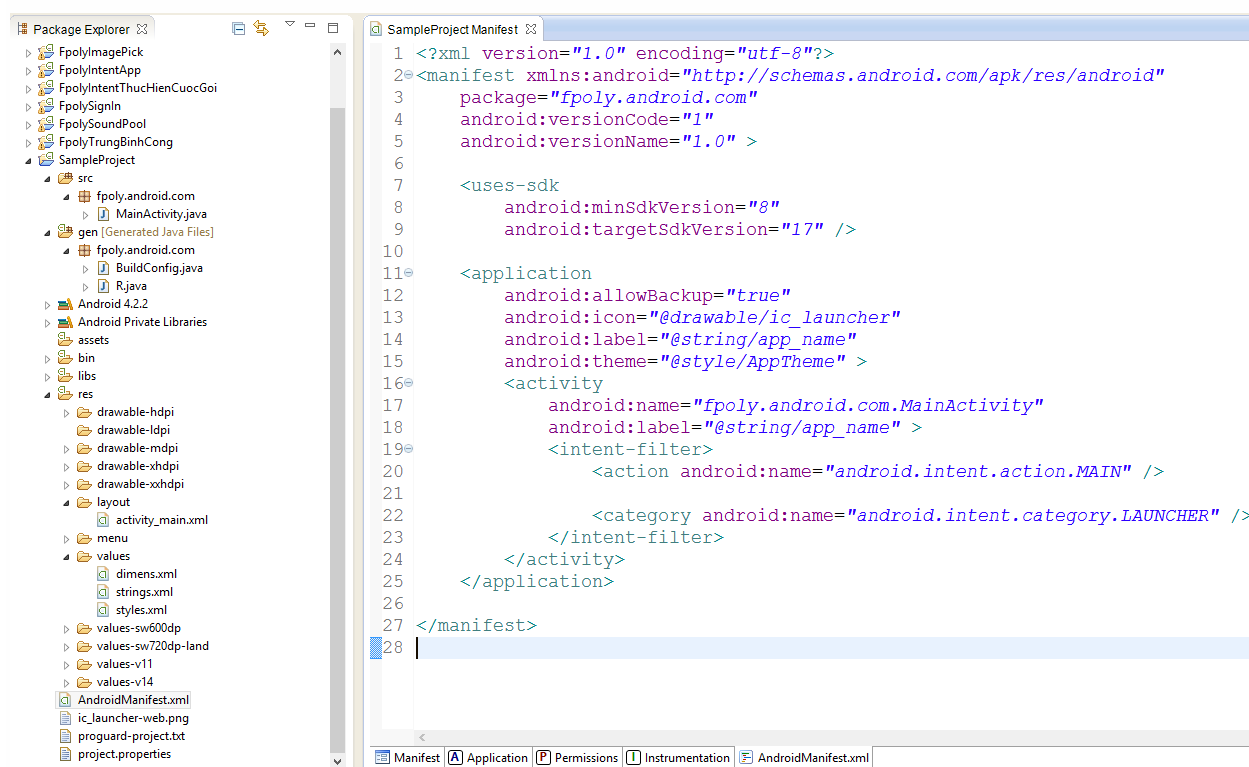
- Bạn cần hiểu được cấu trúc cây trong Ứng Dụng Android ở trên:

Hãy quan sát **MainActivity.java** và **activity_main.xml**. Khi một Ứng dụng được tạo ra thì thông thường sẽ có một Activity để khởi chạy ứng dụng. Ở đây bạn hiểu rằng MainActivity.java chính là class chứa toàn bộ source code, còn activity_main.xml chính là phần giao diện. Đối với Android khi một Activity tạo ra thì thường nó đi kèm với một Layout giao diện nào đó (Tức là nó luôn được tách thành 2 phần: phần source code riêng và phần giao diện riêng). Bạn hiểu

Activity giống như là các màn hình (cửa sổ) tương tự như là C#, mỗi Activity là một màn hình tương tác cụ thể nào đó.

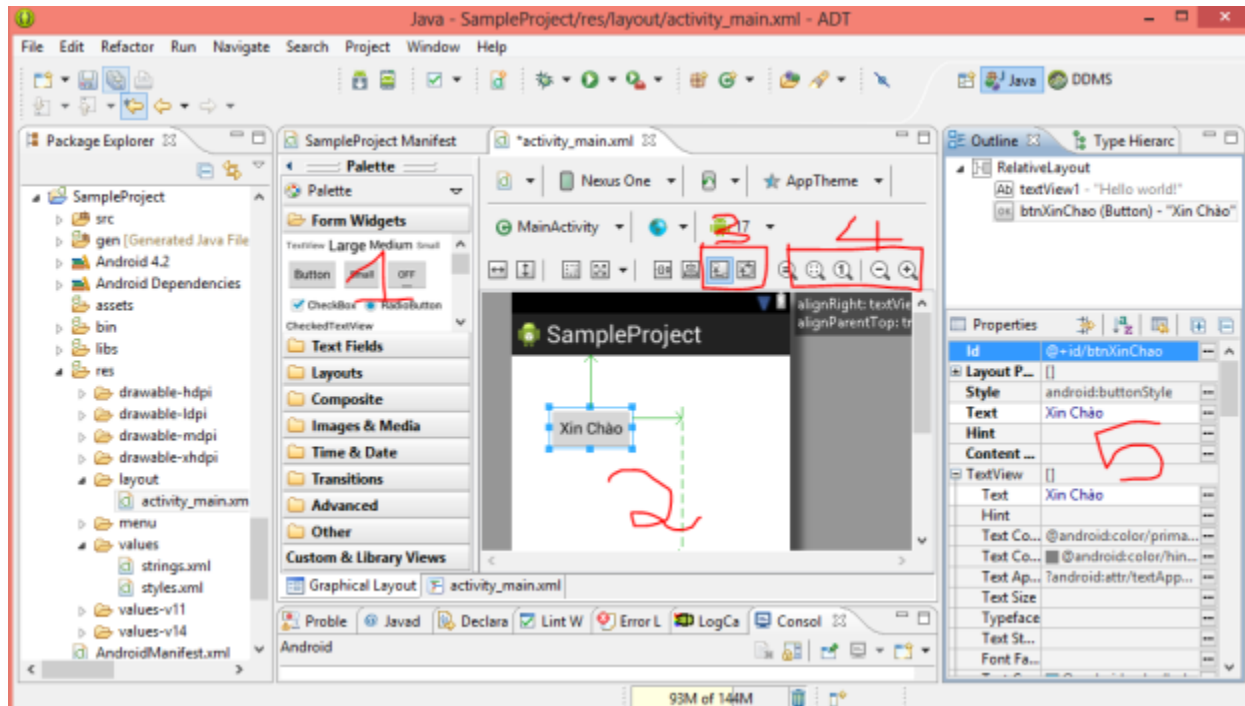
- Bạn cũng cần nhớ rằng bất kỳ một Activity nào muốn được triệu gọi thành công trong Android Project thì bắt buộc nó phải được khai báo trong tập tin **AndroidManifest.xml**. Nếu như bạn gọi một Activity mà Activity này không được khai báo trong Manifest thì chắc chắn chương trình sẽ bị lỗi và tắt luôn.

- Hãy xem cấu trúc bên trong của **AndroidManifest.xml** (double click vào nó và chọn tab như hình bên dưới):



- Như hình bên trên bạn thấy đó, MainActivity muốn được triệu gọi thì nó phải được khai báo trong này. Và đặc biệt nếu như muốn nó là màn hình đầu tiên được thực thi khi chạy ứng dụng thì bạn phải khai báo giống như tag **<intent-filter>** ở trên. Như vậy nếu bạn muốn một Activity bất kỳ nào đó được gọi đầu tiên khi chạy ứng dụng thì bạn chỉ khai báo y chang như vậy, còn các Activity khác bạn không cần khai báo **<intent-filter>** như trên (tức là hoàn toàn không có tag này)

- Tiếp theo bạn double – click vào **activity_main.xml** :



- Hãy quan sát màn hình ở trên (nhìn kỹ có 2 phần **Graphical Layout** và **activity_main.xml**. Bạn nên làm quen trong phần Graphical Layout trước):

+ **Vùng số 1** : chính là nơi chứa các control, layout, component... bạn muốn sử dụng cái nào thì kéo thả nó vào Vùng số 2. Như bạn thấy thì Tôi vừa kéo một Button vào và đặt Id nó là “**btnXinChao**”,

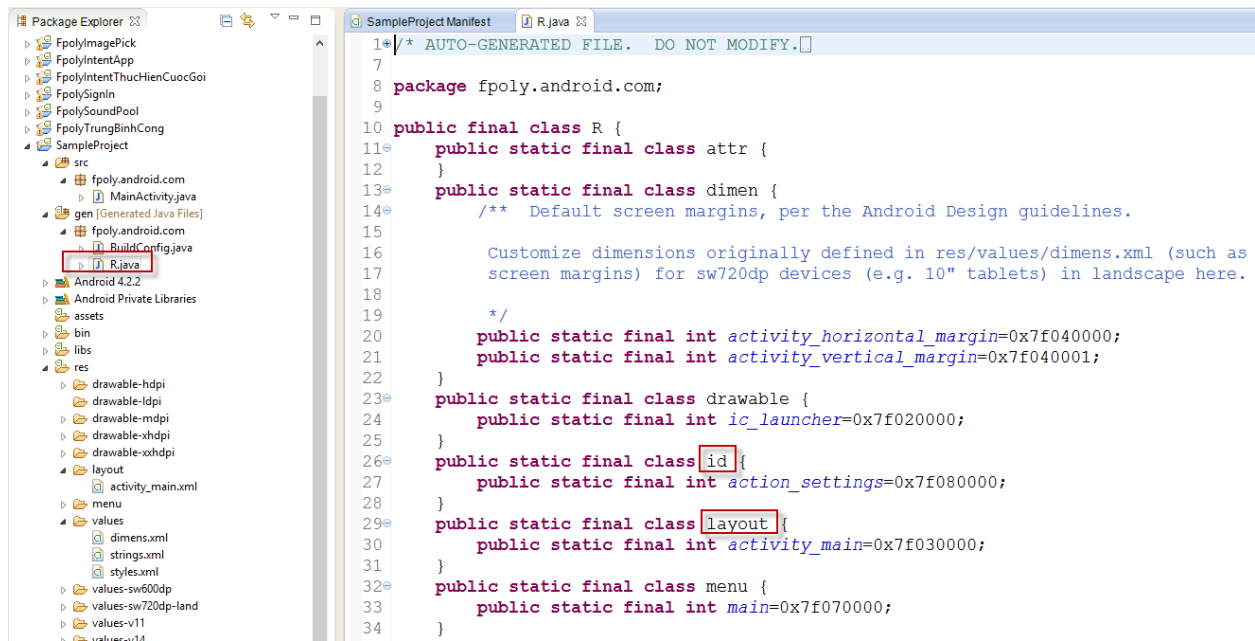
+ **Vùng số 2**: là giao diện, nơi mà bạn cần thiết kế

+ **Vùng số 3**: cho phép thiết kế theo chiều đứng hay chiều ngang

+ **Vùng số 4**: chức năng Zoom in – zoom out để dễ thiết kế

+ **Vùng số 5**: chính là nơi thiết lập các thuộc tính cho các control được kéo thả vào giao diện

- **Bây giờ bạn vào thư mục gen trong Package Explorer và mở tập tin R.java lên:**



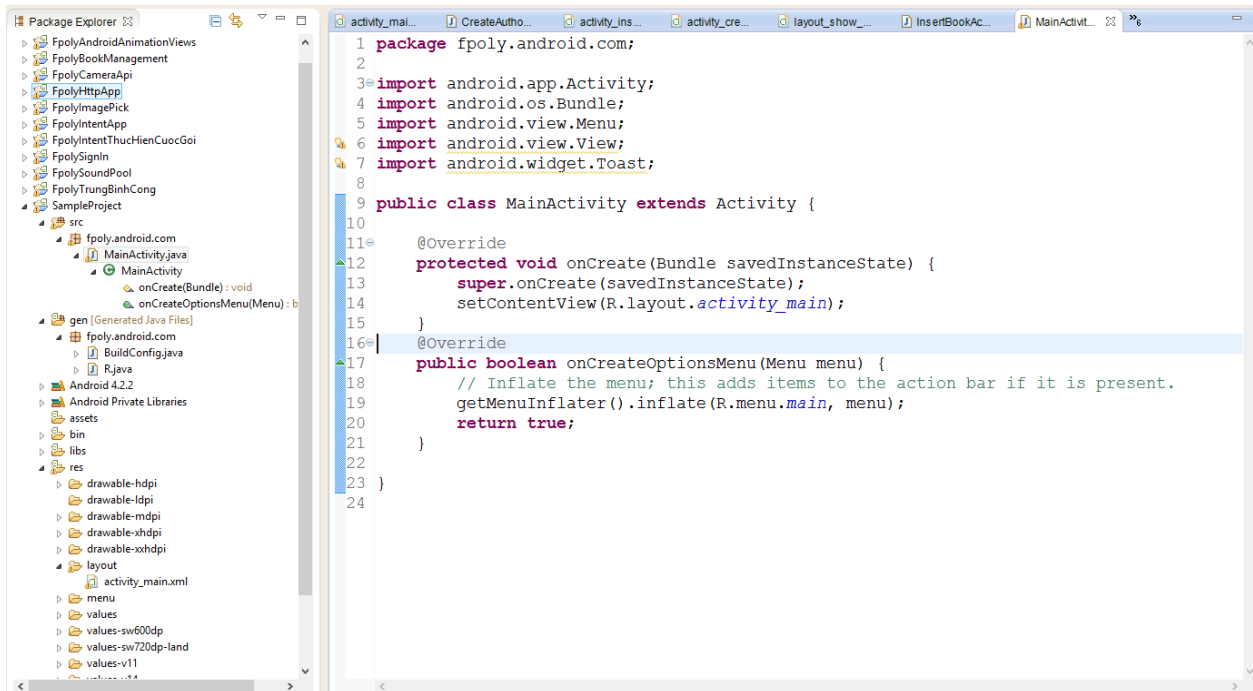
Thư mục **gen**, là thư mục cho Android tự động tạo ra, cho dù bạn có xóa nó thì nó cũng lại tự tạo ra. Nội dung bên trong bạn đừng có chỉnh sửa nó. Tất cả những gì bạn kéo thả vào giao diện, hay thiết lập string.xml, menu ... tất tần tật liên quan tới resource thì nó sẽ được sinh ra bên trong R.java. Dựa vào đây để ta có thể truy xuất các đối tượng trong coding.

Ví dụ:

- Ở trên bạn thấy class Id có chứa Id của Button mà lúc này kéo vào giao diện đặt tên là **btnXinChao**. Ta dựa vào Id này để tương tác với control.

- Hay layout **activity_main** cũng sẽ tự động lưu trữ trong này

- Tiếp tục double – click vào MainActivity.java:



- Trong màn hình trên, bạn thấy bên trong hàm onCreate có lệnh:

+ setContentView(R.layout.activity_main) ==> thiết lập giao diện cho Activity.

Với activity_main lấy từ R.java

+ findViewById(R.id.btnXinChao) ==> truy xuất control là Button trên giao diện.

với btnXinChao lấy từ R.java

* Bạn sẽ hiểu hơn về các hàm này vào các phần sau. Ở đây mục đích là Tôi đang giải thích chức năng của từng phần trong Android

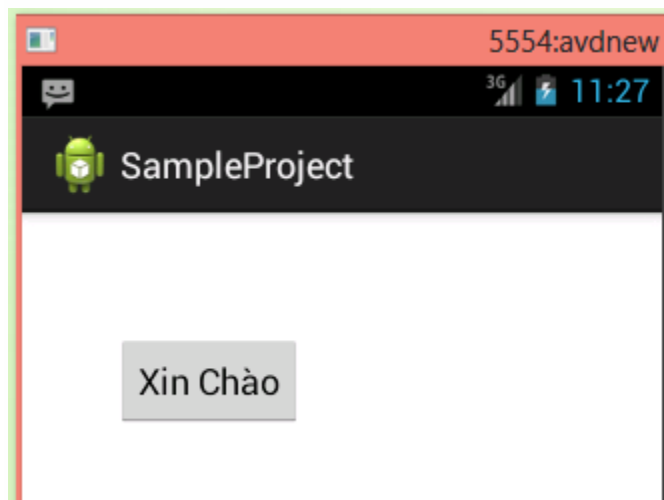
- Tiếp tục bạn quan sát các thư

mục: **Drawable-hdpi, Drawable-ldpi, Drawable-mdpi, Drawable-xdpi:**

Bạn có thể tự tạo thêm một thư mục cùng cấp tên là **Drawable**, các tập tin bạn kéo thả trực tiếp vào trong này (tạo Resource). Khi chương trình load các Resource sẽ tự động vào đây lấy. Còn -hdpi, -ldpi, -xdpi là tùy thuộc vào độ phân giải màn hình mà chương trình tự động vào lấy đúng dữ liệu ở bên trong.

- Tiến hành thực thi chương trình:

Bấm chuột phải vào ứng dụng / chọn Run As/ chọn Android Application , xem kết quả:



- Tôi gom lại thành 5 bước thực hiện của một ứng dụng Android như sau:

bước 1: Android Project sẽ được tự động biên dịch và chuyển qua Android Executables (.dex)

bước 2: Đóng gói thành tập tin .apk

bước 3: Upload .apk vào thiết bị android

bước 4: Tiến hành cài đặt .apk đó

bước 5: Khi cài đặt thành công, chương trình sẽ được thực hiện Activity được thiết lập
: android.intent.action.MAIN

Sửa code file activity_main.xml

```
<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" >

    <Button
        android:id="@+id/btnXinChao"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:layout_marginLeft="38dp"
        android:layout_marginTop="72dp"
        android:text="Xin chào"
        android:onClick="SayHelloClick" />

</RelativeLayout>
```

Sửa code file MainActivity.java

Bổ sung thêm hàm Toast để hiển thị message.

```
package fpoly.android.com;

import android.app.Activity;
import android.os.Bundle;
import android.view.Menu;
import android.view.View;
import android.widget.Toast;

public class MainActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    public void SayHelloClick(View v)
    {
        Toast.makeText(this, "Chào mừng bạn đến với môn học Android cơ bản" +
            "của FPT Polytechnic", Toast.LENGTH_LONG)
            .show();
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
    }
}
```