Google Maps Helloworld

www.mobilepro.vn

Professional Programming

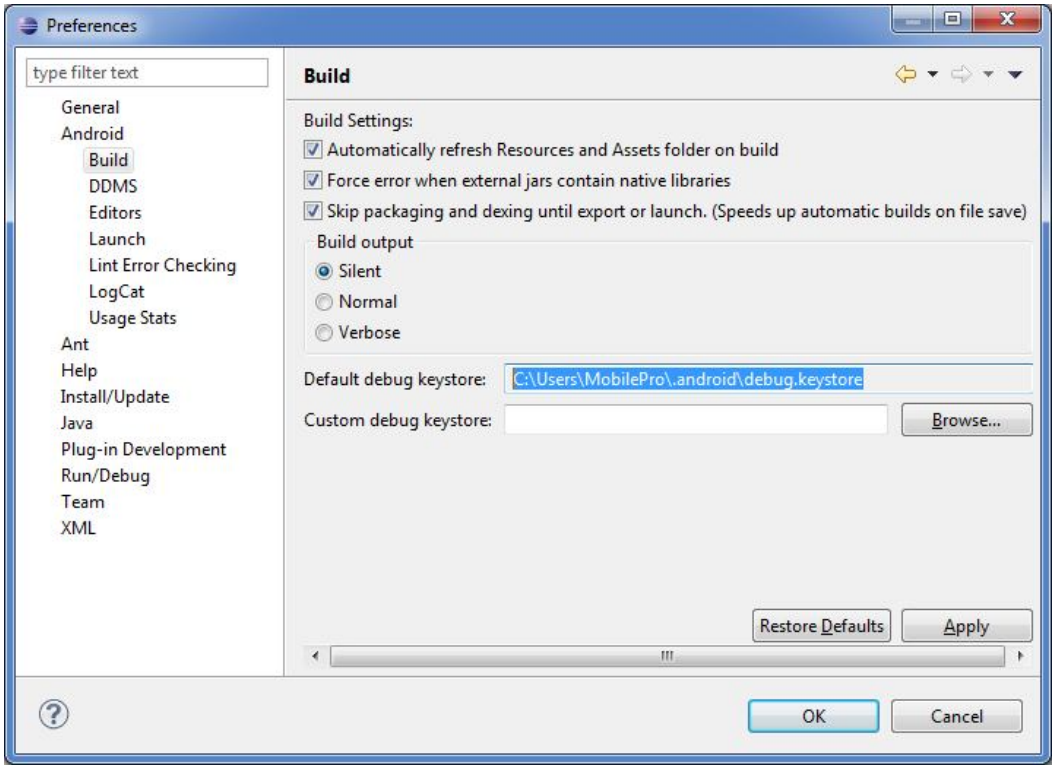
Google Maps API

- Để sử đụng dược Google Maps API trong ứng dụng của mình thì bạn phải đăng ký 1 key miễn phí từ Google Maps API. Để đăng ký được bạn cần làm theo các bước sau :
  - *Bước 1 : Xác định đường dẫn tới file debug.keystore*
  - *Bước 2 : Lấy MD5 fingerprint code*
  - *Bước 3 : Lấy key Google Maps API*

Bước 1 : Xác định file debug.keystore

- Bạn phải xác định được file **debug.keystore** , đường dẫn trên các HDH khác nhau sẽ khác nhau:
  - Windows Vista & Window7: *C:\Users\<user>\.android\debug.keystore*
  - Windows XP: *C:\Documents and Settings\<user>\.android\debug.keystore*
  - OS X and Linux: ~/.android/debug.keystore

Cũng có thể lấy đường dẫn từ Eclipse :
Từ ***Eclipse -> Window -> Preferences -> Android -> Build***
như hình vẽ.



Bước 2 : Lấy MD5 fingerprint code

- Sau khi lấy được đường dẫn của debug.keystore. Bạn bật cmd lên và chạy command sau để lấy MD5 fingerprint code:

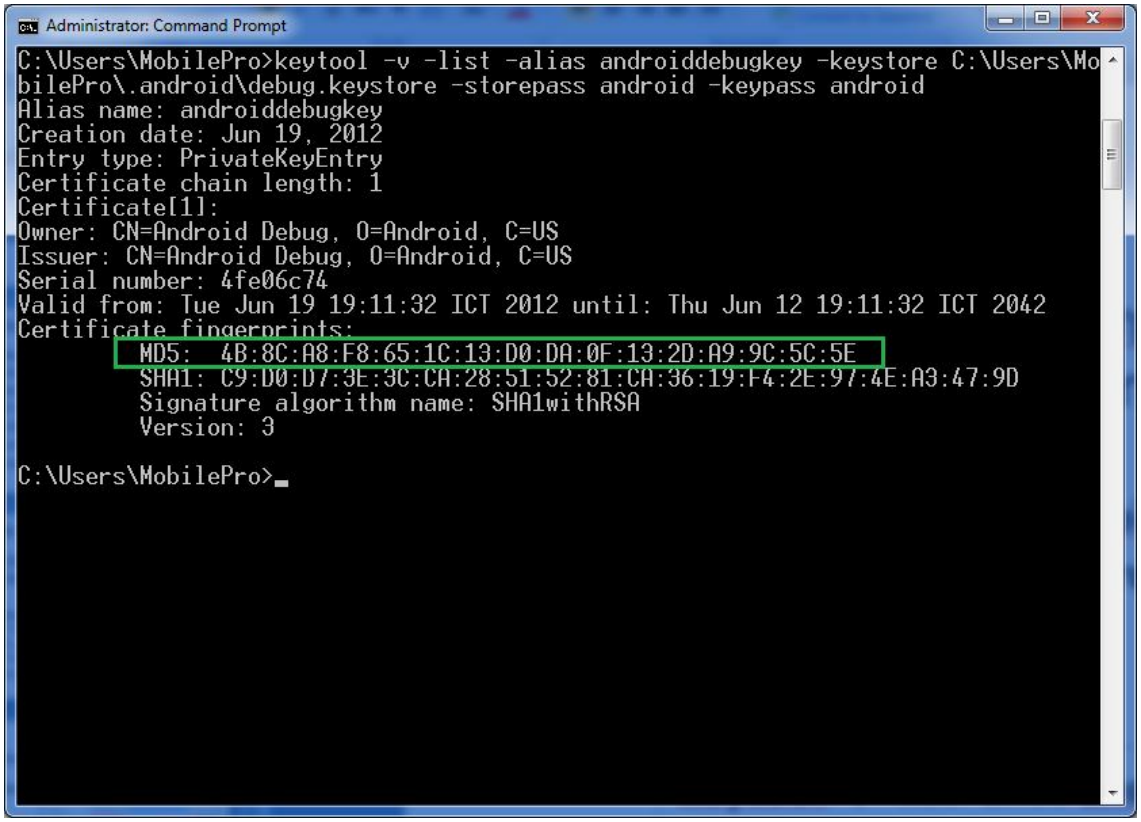
keytool -v -list -alias androiddebugkey -keystore <đường-dẫn-tới-file-keystore> -storepass android -keypass android

- Cụ thể của mình là :

keytool -v -list -alias androiddebugkey -keystore C:\Users\MobilePro\.android\debug.keystore -storepass android -keypass android

Kết quả

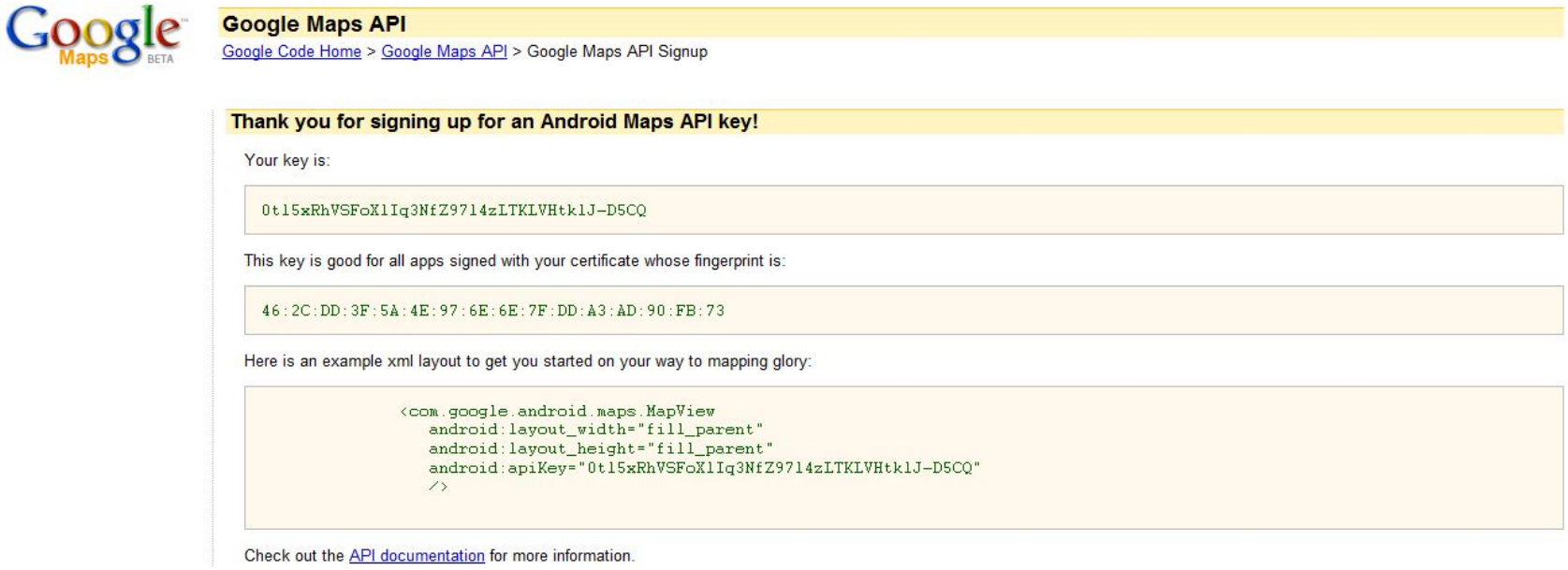
Lưu lại chuỗi MD5 như hình vẽ



Bước 3 : Lấy key Google Maps API

- Bạn vào link sau : http://code.google.com/android/maps-api-signup.html
- Paste MD5 code vào và click Generate API key , site sẽ tự sinh cho bạn 1 key.

Kết quả



Google Maps API

Google Code Home > Google Maps API > Google Maps API Signup

Thank you for signing up for an Android Maps API key!

Your key is:

```
0tl5xRhVSFoXlIq3NfZ9714zLTKLVHtklJ-D5CQ
```

This key is good for all apps signed with your certificate whose fingerprint is:

```
46:2C:DD:3F:5A:4E:97:6E:6E:7F:DD:A3:AD:90:FB:73
```

Here is an example xml layout to get you started on your way to mapping glory:

```
<com.google.android.maps.MapView
   android:layout_width="fill_parent"
   android:layout_height="fill_parent"
   android:apiKey="0tl5xRhVSFoXlIq3NfZ9714zLTKLVHtklJ-D5CQ"
   />
```

Check out the API documentation for more information.

Finished

- Như vậy bạn đã hoàn thành việc đăng ký API Key sử dụng Google Maps.
- Bước tiếp theo bạn sử dụng Key này trong code theo hướng dẫn bên dưới.

Hiển thị Google Maps

- Để sử dụng Google Maps API bạn phải khai báo trong file *AndroidManifest.xml*
- Bằng cách add thêm <*uses-library*> cùng với INTERNET Permission.
- Trong *GoogleMapsActivity.java* bạn phải Extend MapsActivity và overrides phương thức *isRouteDisplayed()*

```
package com.vietandroid.tut.map;

import android.os.Bundle;

import com.google.android.maps.MapActivity;

public class GoogleMapsActivity extends MapActivity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }

      @Override
      protected boolean isRouteDisplayed() {
            // TODO Auto-generated method stub
            return false;
      }
}
```

Map ZoomControl

- Bước 1 : Thêm layout chứa zoom control vào trong cùng layout chứa MapView .

```
<LinearLayout android:id ="@+id/zoom"
    android:layout_width ="wrap_content"
    android:layout_height = "wrap_content"
    android:layout_alignParentBottom = "true"
    android:layout_centerHorizontal = "true"
    />
```

- Bước 2 : Thêm đoạn code để xử lý Zoom View

```
public class GoogleMapsActivity extends MapActivity {

    MapView mapView;
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        //Add zoom view to Google Maps
        mapView = (MapView)findViewById(R.id.myMap);
        LinearLayout zoomLayout = (LinearLayout)findViewById(R.id.zoom);
        View zoomView = mapView.getZoomControls();
        zoomLayout.addView(zoomView, new LinearLayout.LayoutParams(LayoutParams.WRAP_CONTENT, LayoutParams.WRAP_CONTENT));
        mapView.displayZoomControls(true);
    }

    @Override
    protected boolean isRouteDisplayed() {
        // TODO Auto-generated method stub
        return false;
    }
}
```

Zoom Control Shortcuts

- Nếu muốn bổ sung thêm phím tắt để dùng chức năng ZoomIn và ZoomOut , các bạn có thể Override phương thức **onKeyDown()** để bắt sự kiện khi bấm phím:

```
@Override
public boolean onKeyDown(int keyCode, KeyEvent event) {
    // TODO Auto-generated method stub

    MapController mc = mapView.getController();
    switch (keyCode) {
    case KeyEvent.KEYCODE_I:
            mc.zoomIn();
            break;
    case KeyEvent.KEYCODE_O:
            mc.zoomOut();
            break;
    }
    return super.onKeyDown(keyCode, event);
}
```

Map Overlay

- Đánh dấu các địa điểm trên MapView
- Thông thường Overlay sử dụng file ảnh Drawable và có thể tùy biến (inflate 1 layout bất kỳ làm Overlay)

ItemizedOverlay

```
public class HelloItemizedOverlay extends ItemizedOverlay<OverlayItem> {
    private ArrayList<OverlayItem> mOverlays = new ArrayList<OverlayItem>();
    public HelloItemizedOverlay(Drawable defaultMarker) {
        super(boundCenterBottom(defaultMarker));
    }

    public void addOverlay(OverlayItem overlay) {
        mOverlays.add(overlay);
        populate();
    }
    @Override
    protected OverlayItem createItem(int i) {
        return mOverlays.get(i);
    }

    @Override
    public int size() {
        // TODO Auto-generated method stub
        return mOverlays.size();
    }
}
```

Thêm OverlayItem vào MapView

```
List<Overlay> mapOverlays;
Drawable drawable;
HelloItemizedOverlay itemizedOverlay;
```

```
GeoPoint p = new GeoPoint((int)(21.036074 * 1E6), (int)(105.833636 * 1E6));

OverlayItem overlayItem  = new OverlayItem(p, "Title", "");
itemizedOverlay.addOverlay(overlayItem);
mapOverlays.add(itemizedOverlay);
```

Get Current Location

```
private LocationManager lm;
private LocationListener locaListener;
```

```
lm = (LocationManager)getSystemService(Context.LOCATION_SERVICE);
locaListener = new MyLocationListener();
lm.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0, locaListener);
```

```
private class MyLocationListener implements LocationListener{
    @Override
    public void onLocationChanged(android.location.Location location) {
        if(location!=null){
            Toast.makeText(getBaseContext(),"Location is: Lat= "+location.getLatitude()
                                        +"Long= "+location.getLongitude(), Toast.LENGTH_LONG).show();
        }
    }
    @Override
    public void onProviderDisabled(String provider) {
    }

    @Override
    public void onProviderEnabled(String provider) {
    }

    @Override
    public void onStatusChanged(String provider, int status, Bundle extras) {
    }
}
```

The End

- Còn rất nhiều bài giảng & chỉ có tại các lớp học
  www.mobilepro.vn