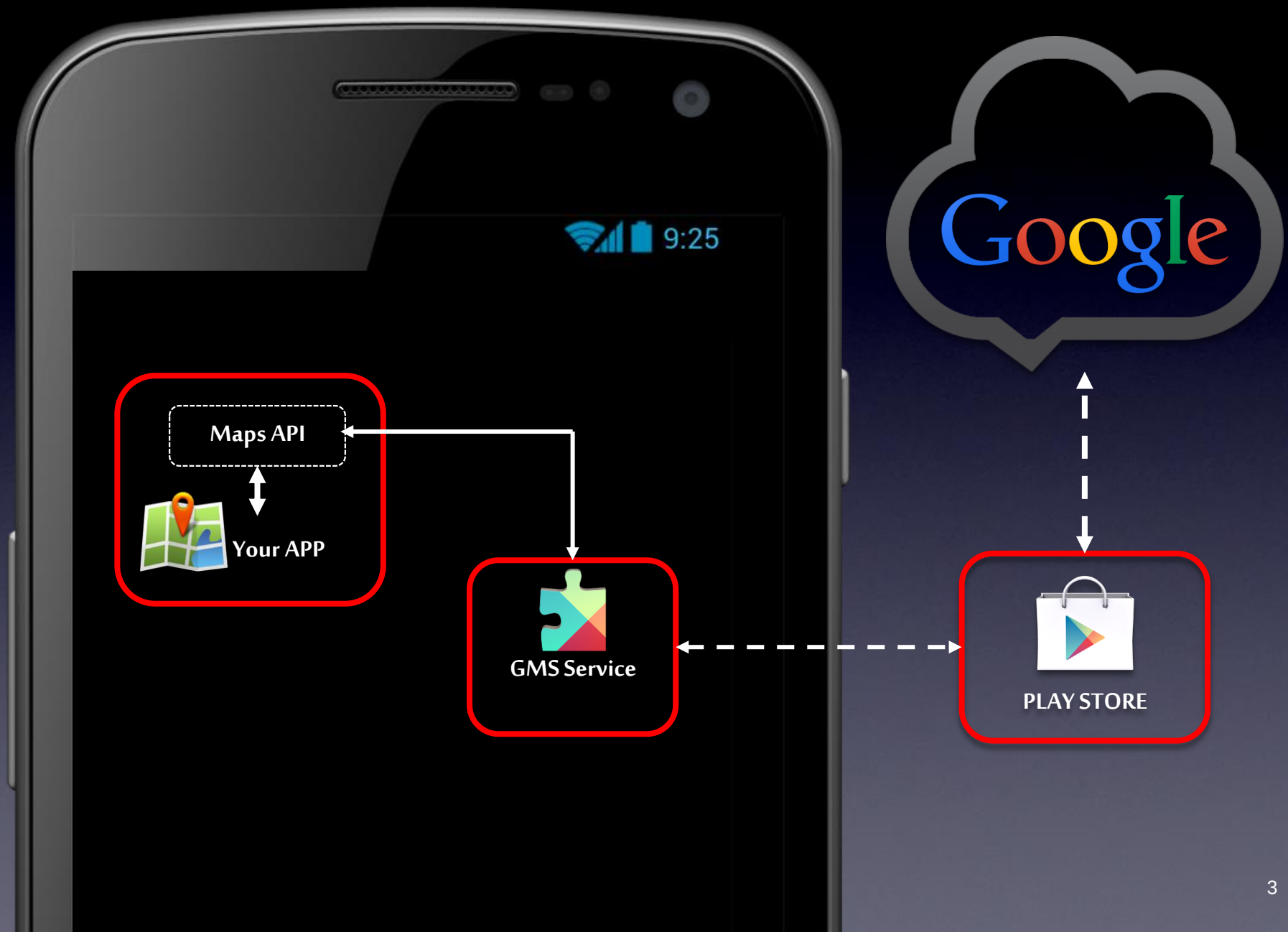
第十三章

GOOGLE MAP

運作原理


9:25
Google
Maps API
Your APP
GMS Service
PLAY STORE

申請方式

步驟

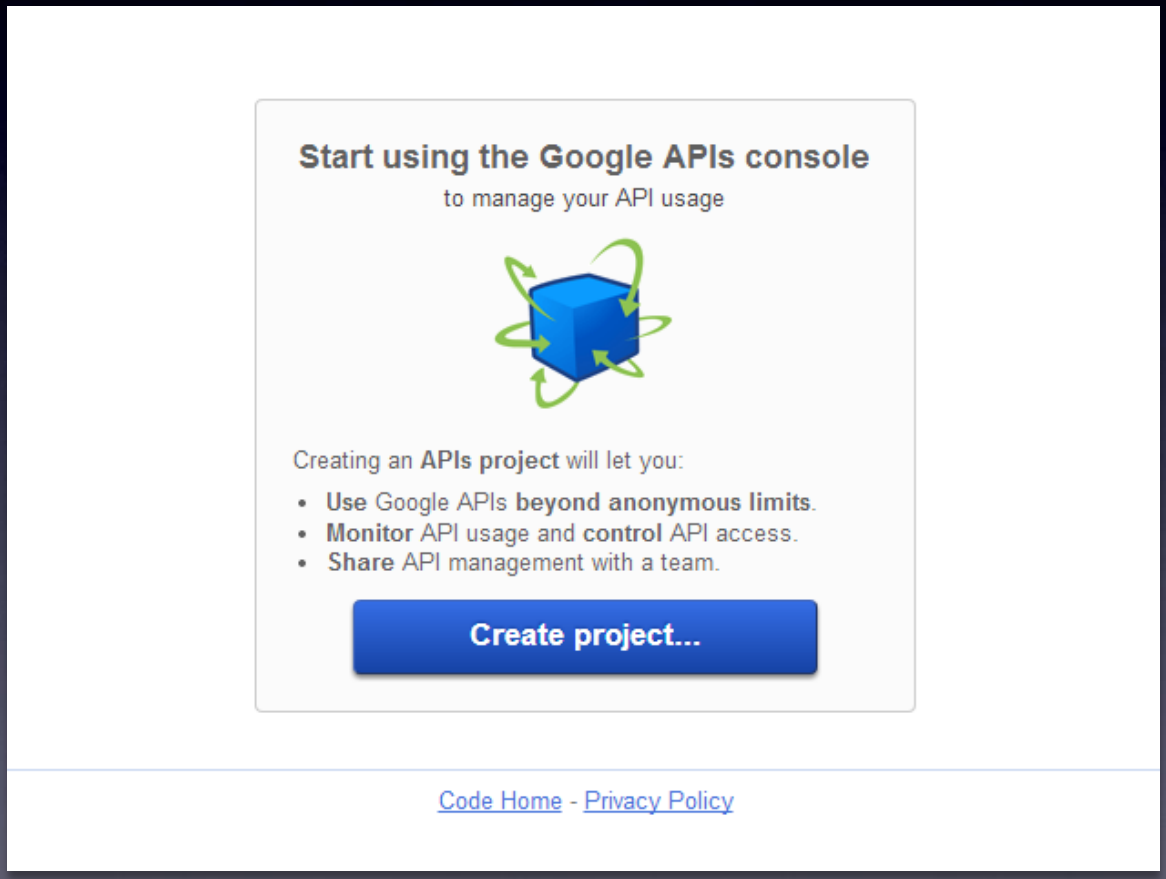
1. 準備好Package Name
2. 登入API Console並開新專案
3. 啟動Google MAP Android API v2.0
4. 產生本機端的SHA1 KEY
5. 使用SHA1 Key + Package Name取得API Key

1. 準備好PACKAGE NAME

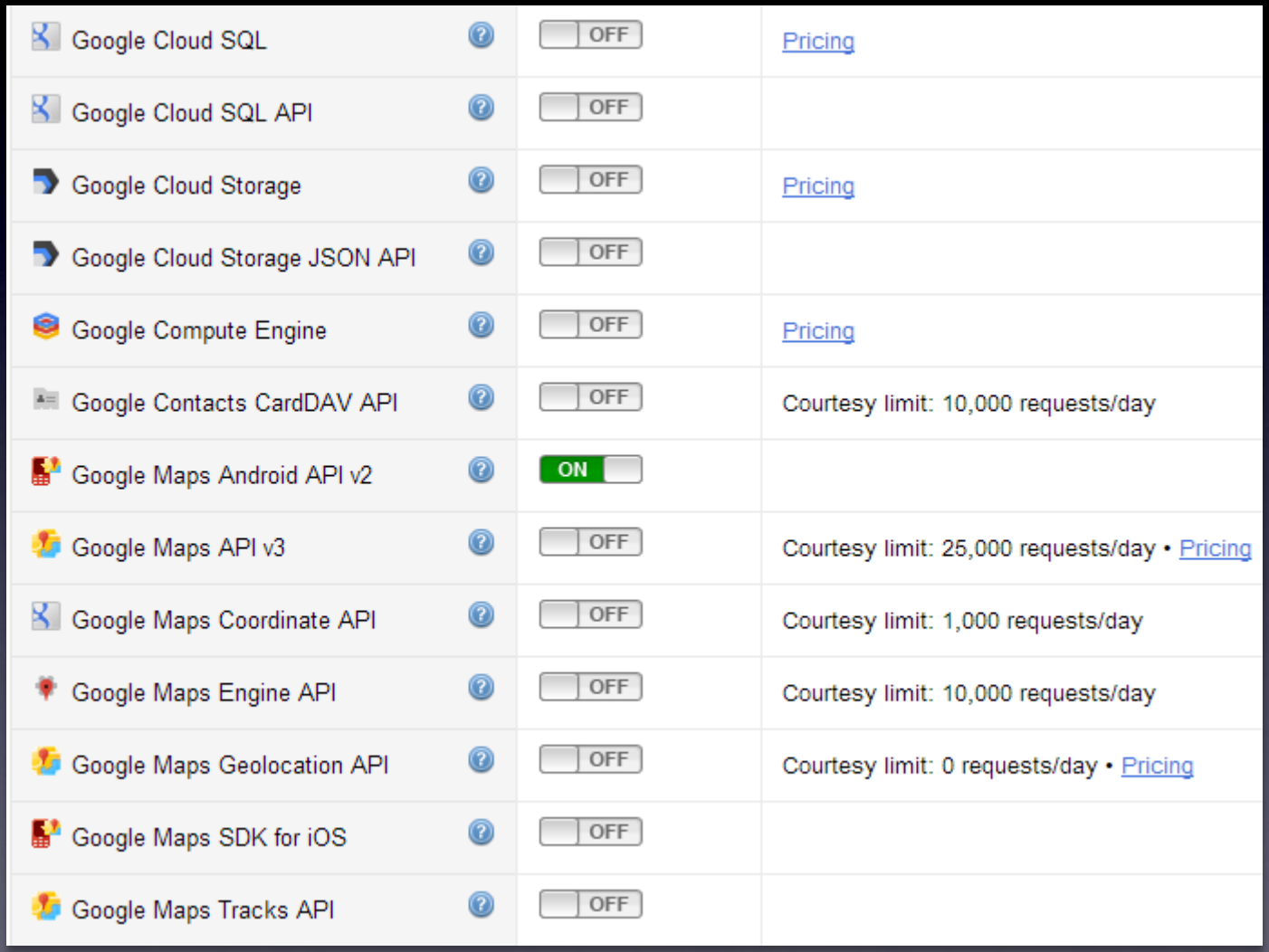
- 建議使用：com.公司名稱.專案名稱
- 申請後要變動很麻煩

2.登入API CONSOLE並開新專案

- https://code.google.com/apis/console



3.啟動GOOGLE MAP ANDROID API V2



4.產生本機端的SHA1 KEY

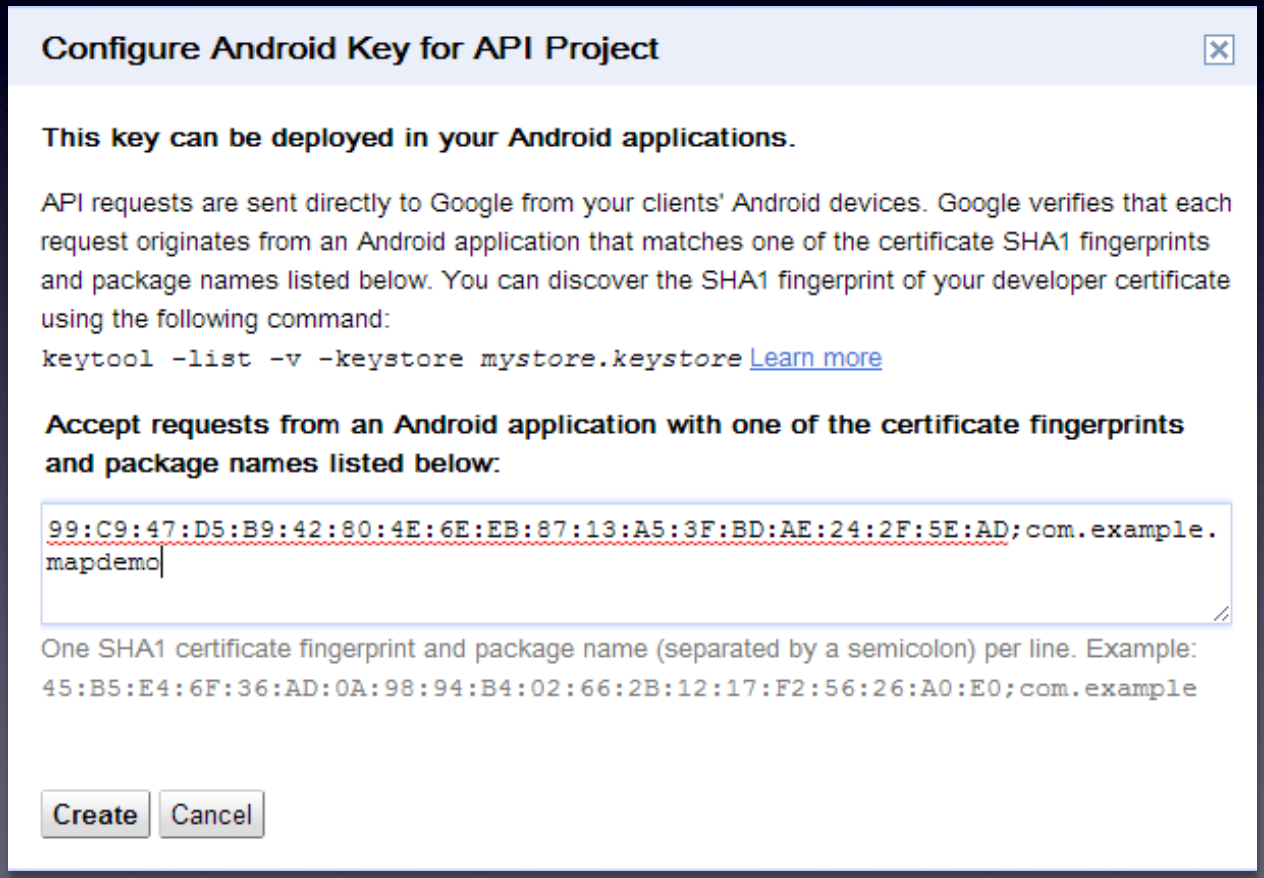
- Windows下打開命令提示字元，輸入以下指令
- keytool -list -v -keystore "C:\Users\用戶名稱\.android\debug.keystore" -alias androiddebugkey -storepass android -keypass android

5. 取得API KEY

- 選擇Create new Android Key

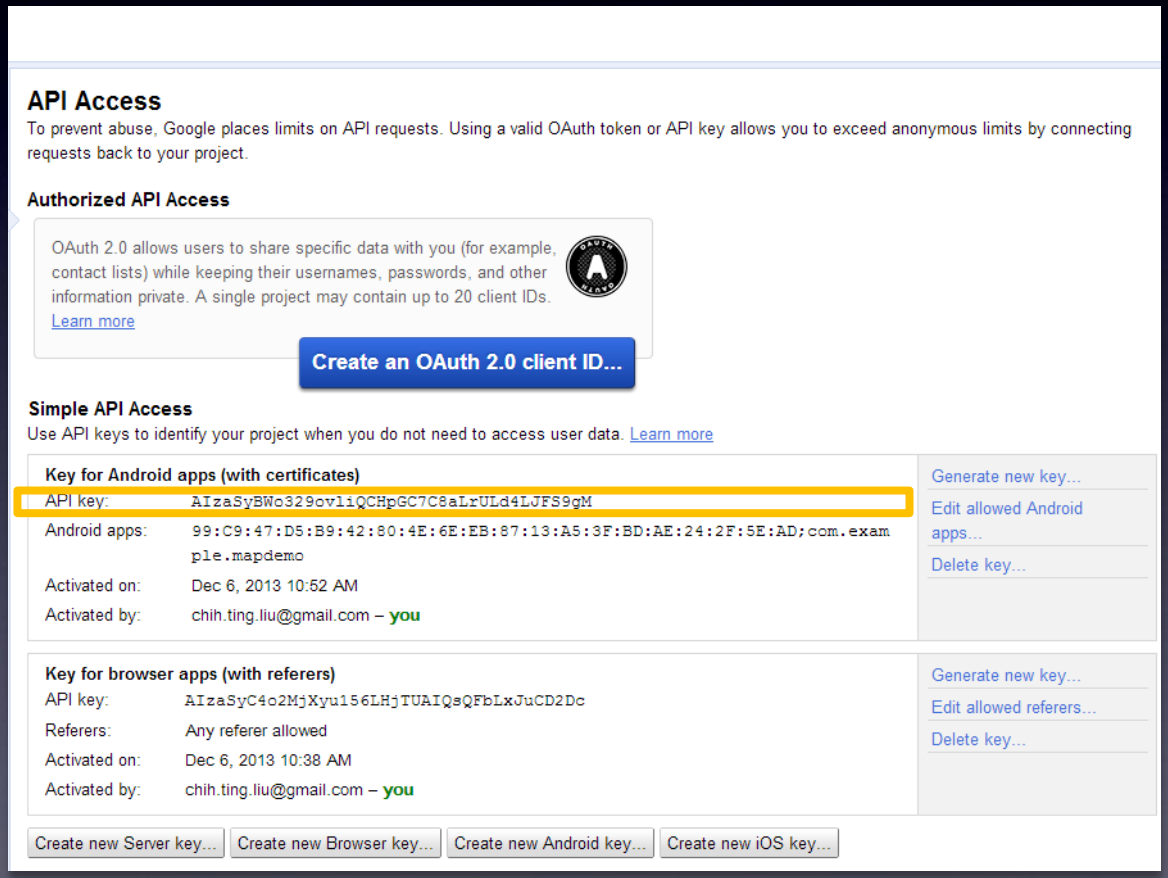
5. 取得API KEY

- 輸入命令提示字元下取得的SHA1 Key並以逗號連接package name



5. 取得API KEY

- 取得API KEY!!

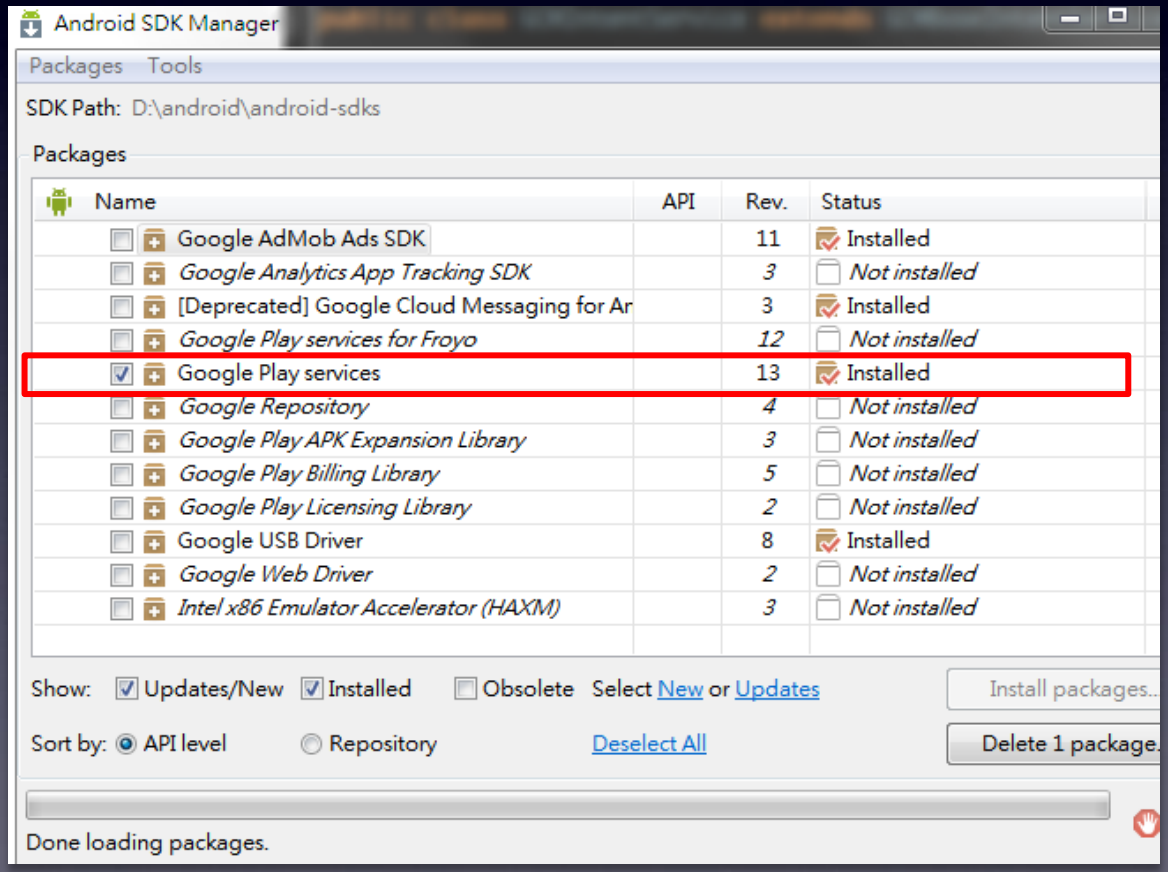


GoogleMap/BasicMap

建立GOOGLE MAP專案

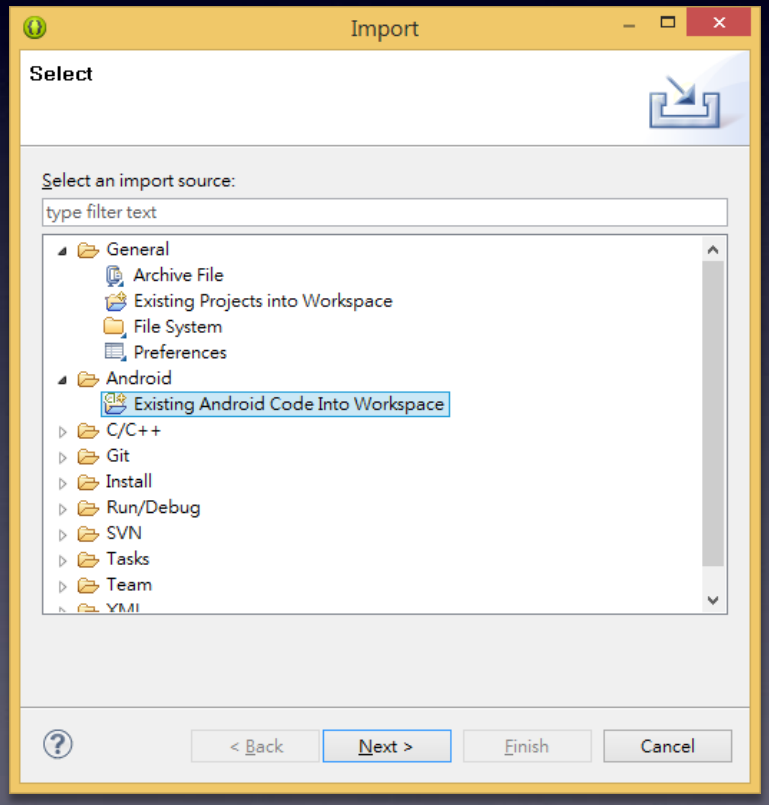
前置作業

- 打開Android SDK Manager
- 確認有下載最新版的Google Play Service



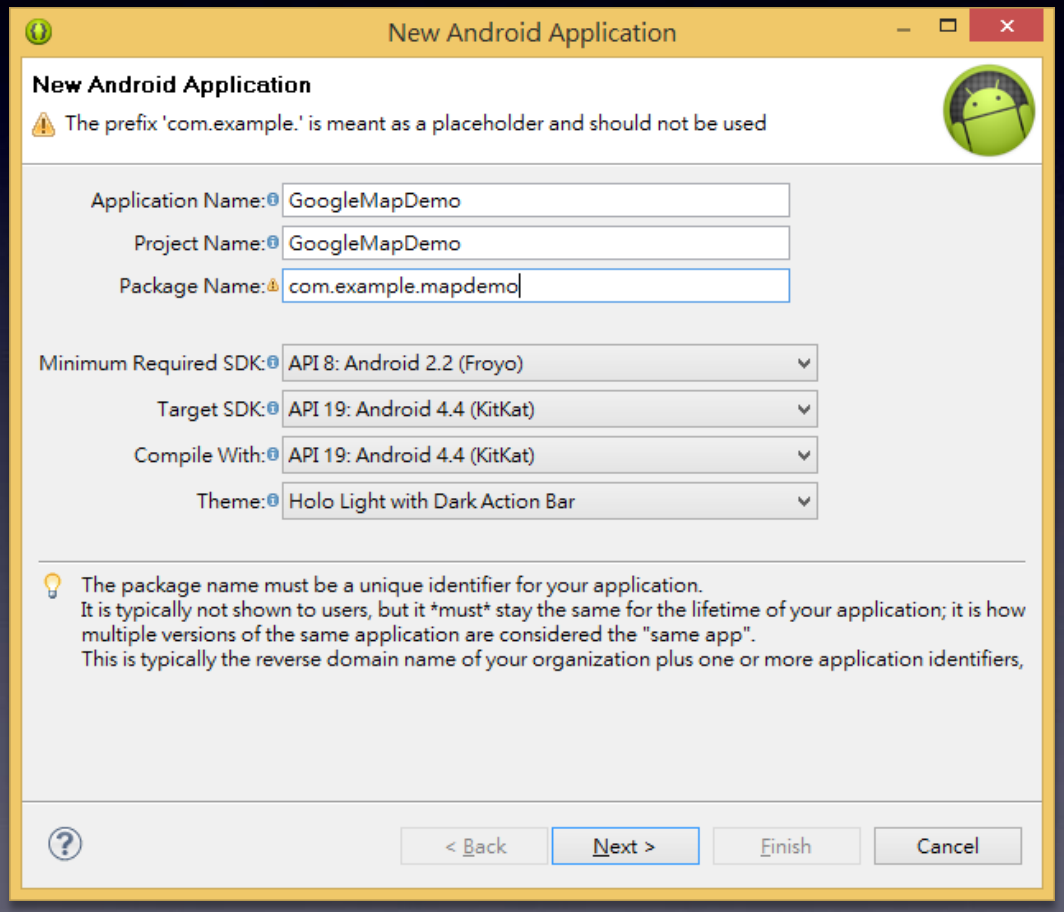
匯入GOOGLE PLAY SERVICE

- File →Import→Android→
  Existing Android Code Into Workspace



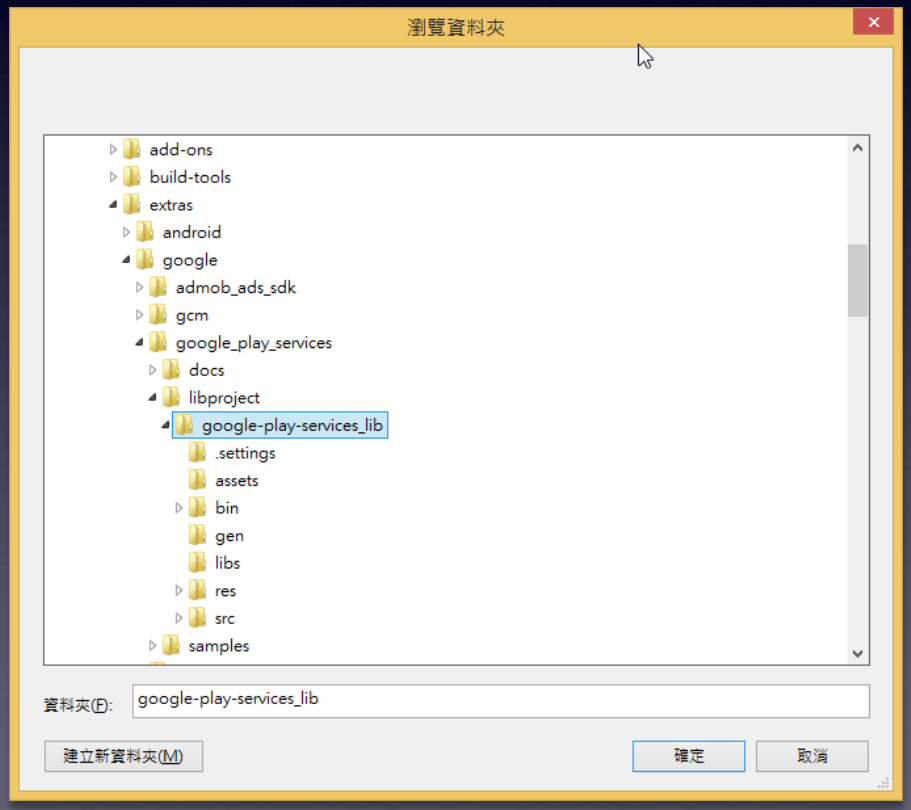
建立專案

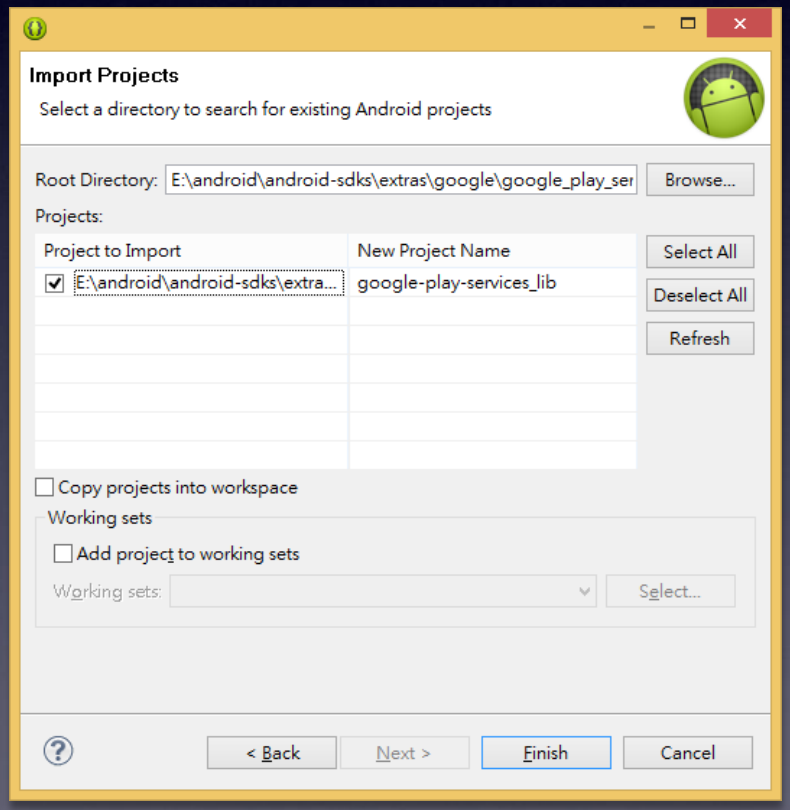
- 建立專案的Package Name要與申請API Key的相同



建立專案

- 匯入路徑: <sdk>\extras\google\google_play_services\libproject\google-play-services_lib



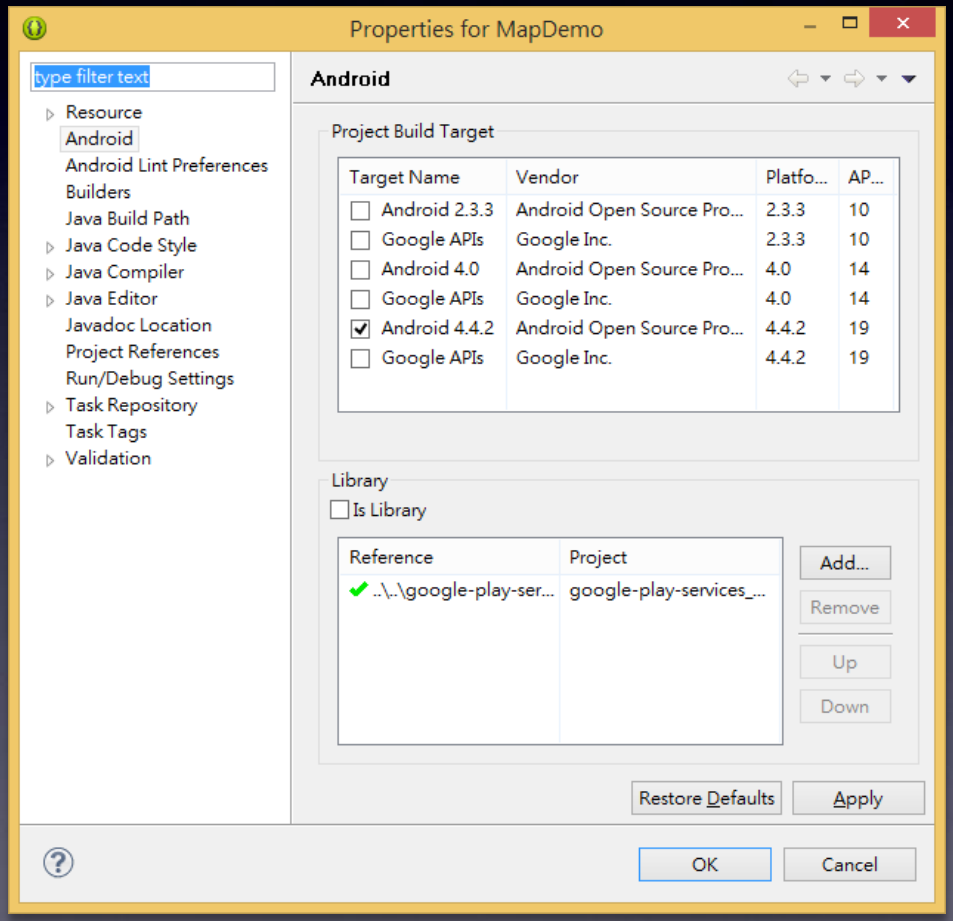


設定專案與PLAY SERVICE關聯

- 對專案按滑鼠右鍵
  Properties→Android→Library→Add

設定專案與PLAY SERVICE關聯

- 選擇google-play-services_lib



GoogleMap/BasicMap

設定APP專案設定檔

設定APP專案設定檔

- 打開AndroidManifest.xml
- 加入以下權限

```
<uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission
android:name="android.permission.INTERNET"/>
<uses-permission
android:name="com.google.android.providers.gsf.permission.READ_GSERVICES"/>
<uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION"/>
<uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
```

設定APP專案設定檔

- 在<application>內加入以下文字資料

```
<meta-data
android:name=“com.google.android.maps.v2.API_
KEY” android:value=“填入API Key"/>
<meta-data
android:name="com.google.android.gms.version"
android:value="@integer/google_play_services_
version"/>
```

程式碼中使用GOOGLE MAP

- Google Map是以Fragment的形式存在的
- 在layout的xml中加入以下程式碼

```
<fragment
    android:id="@+id/map"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    class=
    "com.google.android.gms.maps.SupportMapFragment"/
>
```

程式碼中使用GOOGLE MAP

- 在Activity中，使用setContentView()即可

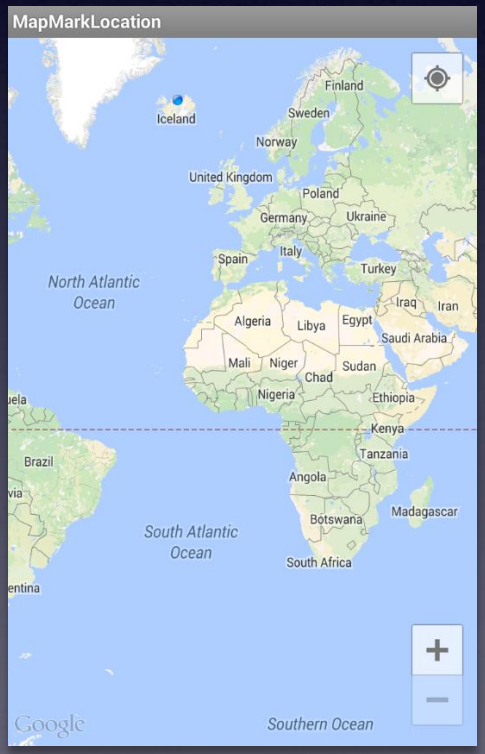
- 執行專案後就可以看到Google Map出現在畫面上了

加入位置圖標與個人位置

對應專案GoogleMap/MapMarkLocation

加入位置圖標與個人位置

- Google Map允許開發者在地圖上加上各種標記
- 也允許開發者很容易的顯示使用者目前所在的位置



取得GOOGLE MAP

```
GoogleMap map = ((SupportMapFragment)
getSupportFragmentManager().findFragmentById(
R.id.map)).getMap();
```

- 在setContentView()後呼叫findFragmentById()可以找到GoogleMap的Fragment
  - Id就帶入在layout的xml設定給map fragment的id

取得GOOGLE MAP

```
GoogleMap map = ((SupportMapFragment)
getSupportFragmentManager().findFragmentById(
R.id.map)).getMap();
```

- getMap()
  - 將取得的Fragment轉型為SupportMapFragment後，可以呼叫getMap()取得Google Map

加入個人位置

```
map.setMyLocationEnabled(true);
```

- 設定setMyLocationEnable()為true，就會自動在地圖右上方出現我的位置按鈕 

加入圖標

```
MarkerOptions options = new MarkerOptions();
options.position(new LatLng(24.989926, 121.545414));
options.title("世新大學");
options.snippet("台北市文山區試院路154巷1弄7號");
map.addMarker(options);
```

- 圖標是MarkOption，直接以new建立

加入圖標

```
MarkerOptions options = new MarkerOptions();
options.position(new LatLng(24.989926, 121.545414));
options.title("世新大學");
options.snippet("台北市文山區試院路154巷1弄7號");
map.addMarker(options);
```

加入圖標

```
MarkerOptions options = new MarkerOptions();
options.position(new LatLng(24.989926, 121.545414));
options.title("世新大學");
options.snippet("台北市文山區試院路154巷1弄7號");
map.addMarker(options);
```

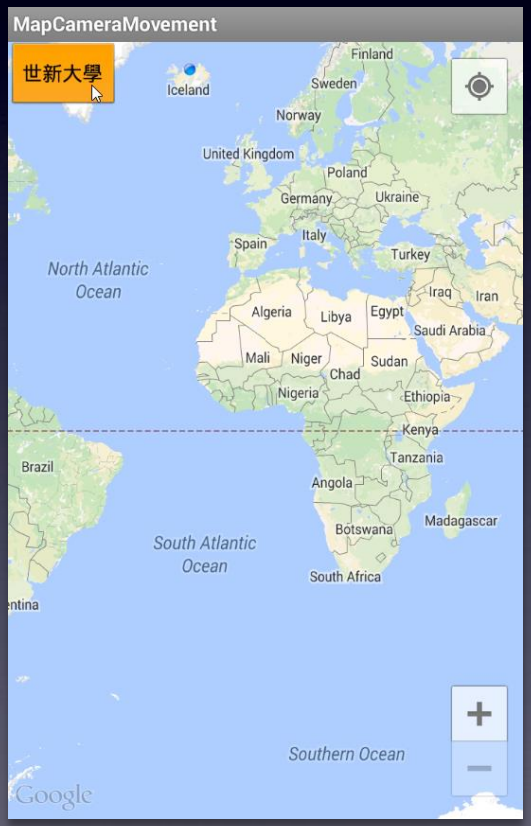
- addMarker(MarkOptions)就可以把圖標加入地圖中

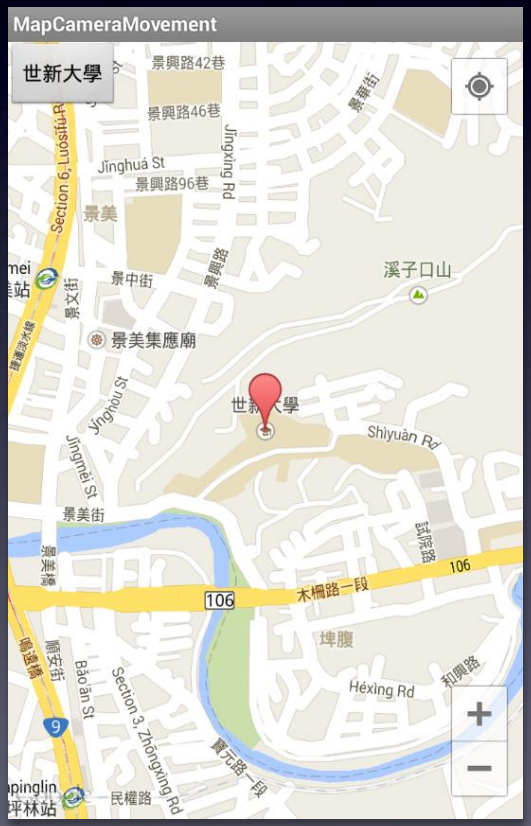
GoogleMap/MapCameraMovement

移動攝影機

移動攝影機

- 要如何讓畫面自動移動到地圖上的指定位置呢？





移動攝影機

```
CameraPosition cameraPosition = new
CameraPosition.Builder().zoom(15.5f).target(mShihHsinU
niversityLocation).build();

CameraUpdate update =
CameraUpdateFactory.newCameraPosition(cameraPosition);

mMap.animateCamera(update);
```

- 首先建立一個CameraPosition，使用CameraPositionBuilder
  - zoom() 標示移動的畫面要放大的程度
  - target() 畫面移動的目標位置
  - build() 依照設定的數值建立CameraPosition

移動攝影機

```
CameraPosition cameraPosition = new
CameraPosition.Builder().zoom(15.5f).target(mShihHsinU
niversityLocation).build();

CameraUpdate update =
CameraUpdateFactory.newCameraPosition(cameraPosition);

mMap.animateCamera(update);
```

- 使用CameraPosition的資訊建立一個CameraUpdate
  - CameraUpdateFactory.newCameraPosition()參數擺入CameraPosition

移動攝影機

```
CameraPosition cameraPosition = new
CameraPosition.Builder().zoom(15.5f).target(mShihHsinU
niversityLocation).build();

CameraUpdate update =
CameraUpdateFactory.newCameraPosition(cameraPosition);

mMap.animateCamera(update);
```

- animateCamera()設定後，畫面馬上就會開始移動

GoogleMap/MapEvent

地圖的觸控事件

GOOGLE MAP觸控事件

- GoogleMap提供多種事件讓開發者可以針對使用者的操作方式顯示不同的行為
- 本例子包含以下功能
  - 點選地圖就在點選處加入Marker
  - 長按地圖就顯示長按處的經緯度位置
  - 可拖移Marker的位置
  - 點一下Marker，Marker就會移除
  - 更換Marker的圖示

GOOGLE MAP觸控事件

```
marker.draggable(true);
BitmapDescriptor icon =
BitmapDescriptorFactory.fromResource(R.drawable.pin);
marker.icon(icon);
```

- 更多Marker的屬性
  - draggable() true表示marker是可以被拖移的
  - icon() 變更Marker的圖示，參數為BitmapDescriptor
    - BitmapDescriptorFactory.fromResource(resource id) 建立marker的圖示

GOOGLE MAP觸控事件

```
map.setOnMapClickListener(new OnMapClickListener() {
      public void onMapClick(LatLng latlng) {
      }
});
```

- setOnMapClickListener()
  - 可以取得點選地圖的事件，參數為OnMapClickListener()
  - onMapClick(LatLng)
    參數會由系統傳入使用者點選的經緯度位置

GOOGLE MAP觸控事件

```
mMap.setOnMapLongClickListener(new
      OnMapLongClickListener() {
      public void onMapLongClick(LatLng latlng) {
      }
});
```

- setOnMapLongClickListener()
  - 可以取得長按地圖的事件，參數為OnMapLongClickListener()
  - onMapLongClick(LatLng)
    參數會由系統傳入使用者長按的經緯度位置

GOOGLE MAP觸控事件

```
mMap.setOnMarkerDragListener(new OnMarkerDragListener() {
        public void onMarkerDragStart(Marker marker) { }
        public void onMarkerDragEnd(Marker marker) {}
        public void onMarkerDrag(Marker marker) {}
});
```

- setOnMarkerDragListener()
  - 得知Marker被拖移的事件，參數為OnMarkerDragListener
  - onMarkerDragStart(Marker) 參數會傳入準備開始被拖移的Marker
  - onMarkerDrag(Marker) 參數會傳入正在被拖移的Marker
  - onMarkerDragEnd(Marker) 參數會傳入結束拖移的Marker

GOOGLE MAP觸控事件

```
mMap.setOnMarkerClickListener(new OnMarkerClickListener() {
        public boolean onMarkerClick(Marker marker) {
                ……
                marker.remove();
                return true;
        }
});
```

- setOnMarkerClickListener()
  - 參數為OnMarkerClickListener()
    - onMarkerClick() 參數會由系統傳入被點選到的Marker
    - 回傳true表示觸控事件已被處理完，不用再傳入到下一層
- marker.remove() 可以將marker從地圖上移除

GoogleMap/MapLayer

地圖圖層

地圖圖層

- 本例子要介紹
  - 開啟3D建築物
  - 顯示交通狀況
  - 顯示室內地圖
  - 讓攝影機視角傾斜看地圖

攝影機傾斜

```
CameraPosition pos = new CameraPosition(new LatLng(25.04188,
121.53309), 19.f, 75.0f, 0f);
CameraUpdate initialize =
CameraUpdateFactory.newCameraPosition(pos);
mMap.moveCamera(initialize);
```

- 四個參數的CameraPosition建構子
  - 參數一、經緯度
  - 參數二、放大的程度
  - 參數三、傾斜程度
  - 參數四、方位，0度到359度

顯示建築物、交通狀況、室內地圖

```
mMap.setBuildingsEnabled(boolean);
mMap.setTrafficEnabled(isChecked);
mMap.setIndoorEnabled(isChecked);
```

- setBuildingEnabled()決定是否顯示建築物
- setTrafficEnabled()決定是否顯示交通狀況
- setIndoorEnabled()如果放大程度夠，剛好地圖上的建築物有申請室內導航，就會顯示室內地圖

室內地圖

- Google提供室內地圖的功能，讓一般用戶可以申請
  - http://www.google.com/maps/about/partners/indoormaps

- 本例子以經緯度37.614631, -122.385153的San Francisco International Airport為例子

更多資料

更多資料

- http://developer.android.com/google/play-services/maps.html