

日本アンドロイドの会

第5回 関西支部セミナー

意外？と簡単GPSアプリ

株式会社ブリリアントサービス
技術部 藤井洋祐



Brilliant Service

■ 自己紹介

- ・ 名前 ふじい ようすけ
- ・ 所属 ぶりりあんとさーびす ぎじゅつぶ
- ・ 職歴 金融端末(日本初ISDN回線接続CD開発)
医療品発注システム(クライアント側)
北米向け携帯電話(PTT、Brewなど)
- ・ アンドロイド歴 半年未満



Brilliant Service

■何を作ろうか、まず考える。

- 今まで作った、もしくは本を見ながら作成したアプリは？
 - 基本のEditor
 - GoogleMapを使ったGPS情報による地図表示アプリ
 - SQLite 第3回セミナーで紹介



Brilliant Service

■更に考える。

- GoogleMapで遊ぼうかな。緯度、経度を適当に入れたら、どこを表示するか。
- 緯度、経度を元にアドレスを取得するメソッドがあったはず。
- 今まで作成したEditorで緯度、経度を入力するようにしよう。
- MapViewに、入力した緯度、経度を渡すようにしよう。
- Activity間の通信は、SQLiteでしよう。



Brilliant Service

■GPS (Global Positioning System)

- 数個のGPS衛星の電波を受け取ることによる
 - 衛星からの時刻信号と位置情報サーバとの通信による
- ※他にも用途によって色々方式あり



Brilliant Service

■ 準備してみよう

- 使用環境
SDK2.0
Eclipse3.5
ADT0.94
- 参考文献
初めてのAndroid（オライリージャパン）



Brilliant Service

■SDK2.0インストール 1/3

- ・今回から、圧縮ファイルの中にプラットフォームが入っていない。
DLする必要がある。。。めんどくさい。

- ・とりあえずWindows環境の場合

<http://developer.android.com/intl/ja/sdk/index.html>

[android-sdk_r3-windows.zip](#)を選択し、ダウンロード。

完了後、解凍し、

~\android-sdk-windows

にあるSDK Setup.exe

コマンドラインからだと以下のようにbat実行。

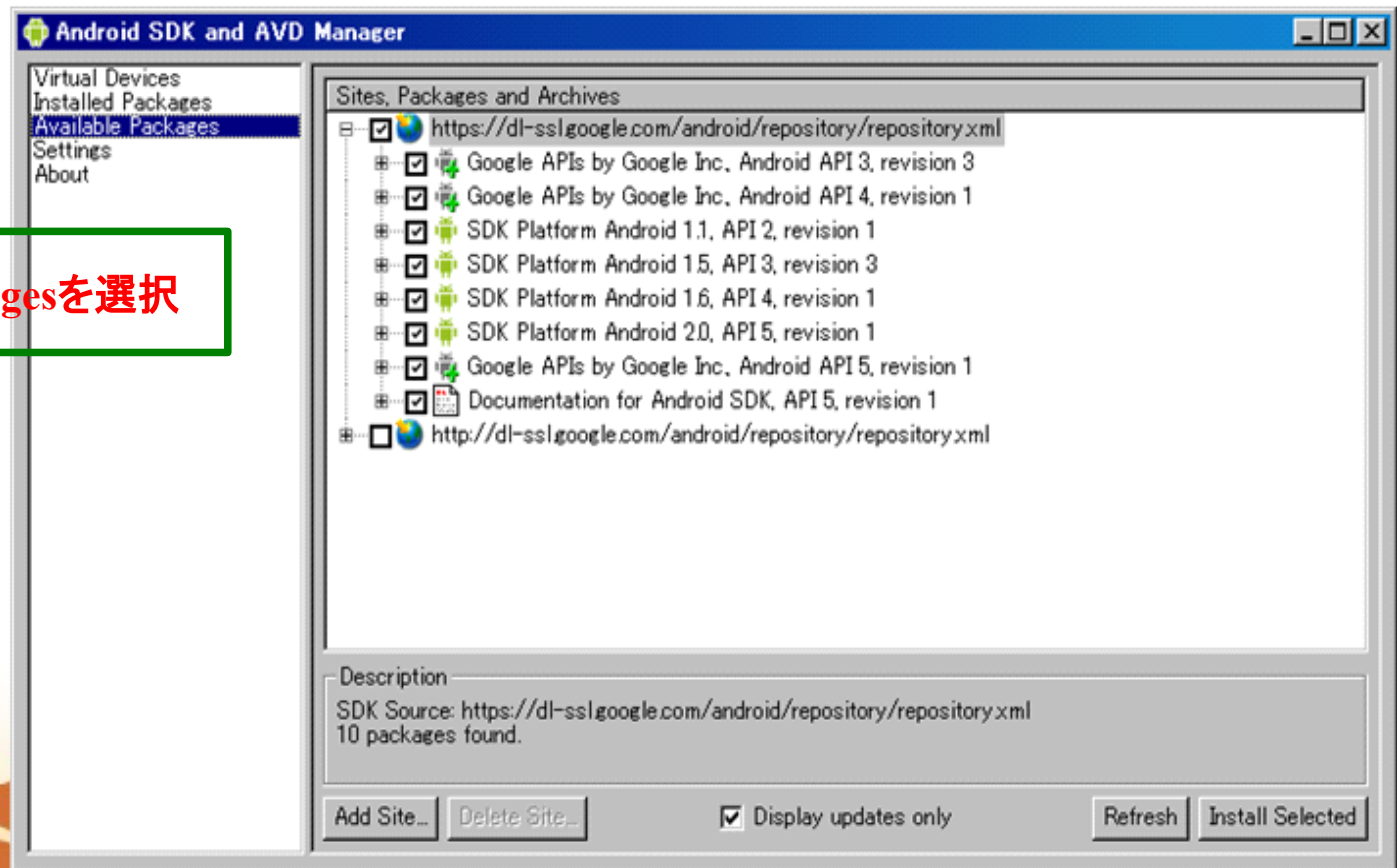
~\tools\android.bat update sdk



Brilliant Service

■SDK2.0インストール 2/3

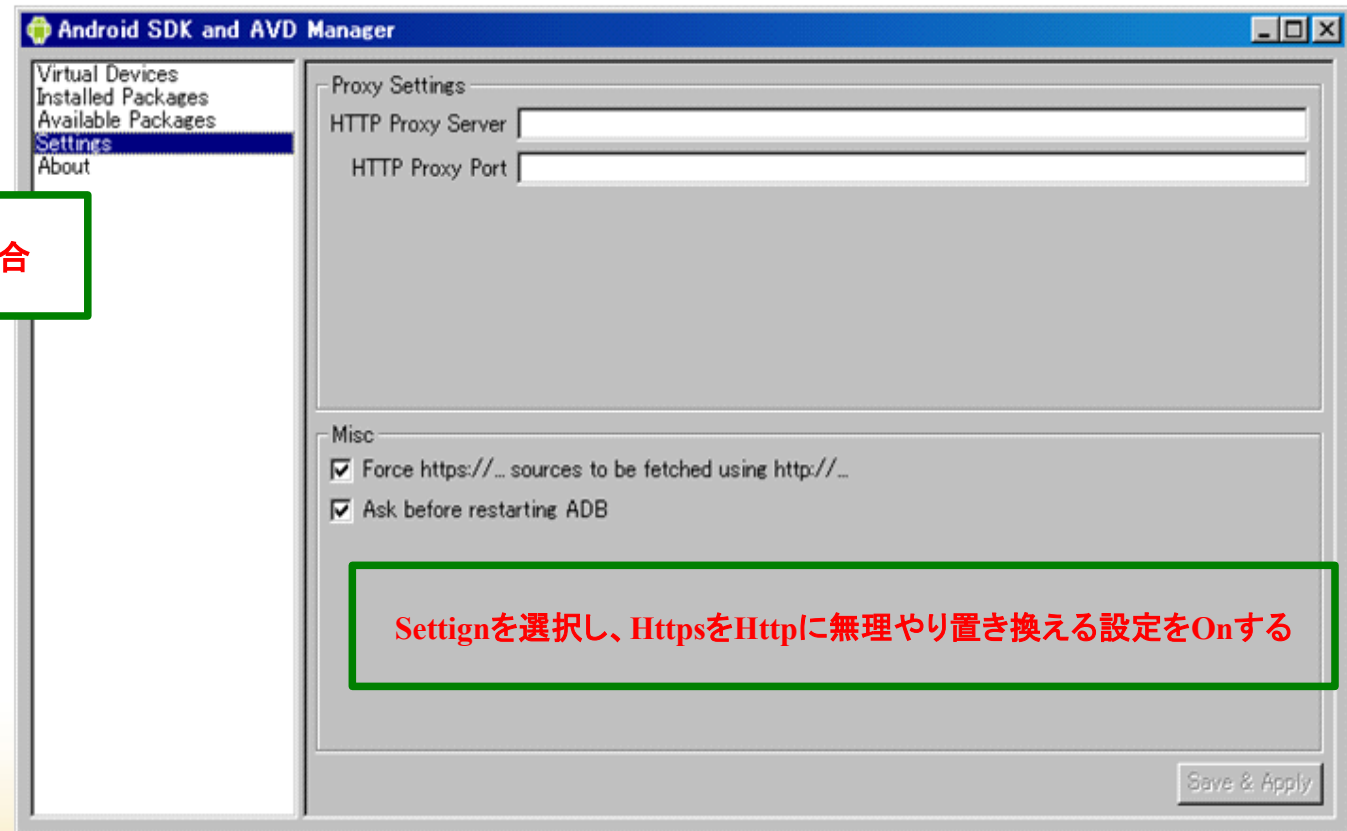
Available Packagesを選択



Brilliant Service

■SDK2.0インストール 3/3

SSLでエラーとか言われた場合



Brilliant Service

■GPSで位置を取得する

- **implements** `LocationListener`

⇒`public class` `GPS_NowLocationMapView` **extends** `MapActivity`
implements `LocationListener`{

- `(LocationManager) getSystemService (`
`Context.`*`LOCATION_SERVICE`*`);`

⇒GPSサービスの登録

- `requestLocationUpdates (`
`LocationManager.`*`GPS_PROVIDER`*`, 0, 0,` **this**`)`

⇒GPS情報変更割り込み許可



Brilliant Service

■LocationListener

1/5

AndroidManifest.xml
追加となるパーミッション

<!-- GPSのパーミッション -->

```
<uses-permission
  android:name="android.permission.ACCESS_FINE_LOCATION">
</uses-permission>
```

<!--GPSアプリ開発用のモック -->

```
<uses-permission
  android:name="android.permission.ACCESS MOCK_LOCATION">
</uses-permission>
```

<!-- Mapがインターネットにアクセスするので。-->

```
<uses-permission android:name=
  "android.permission.INTERNET">
</uses-permission>
```



Brilliant Service

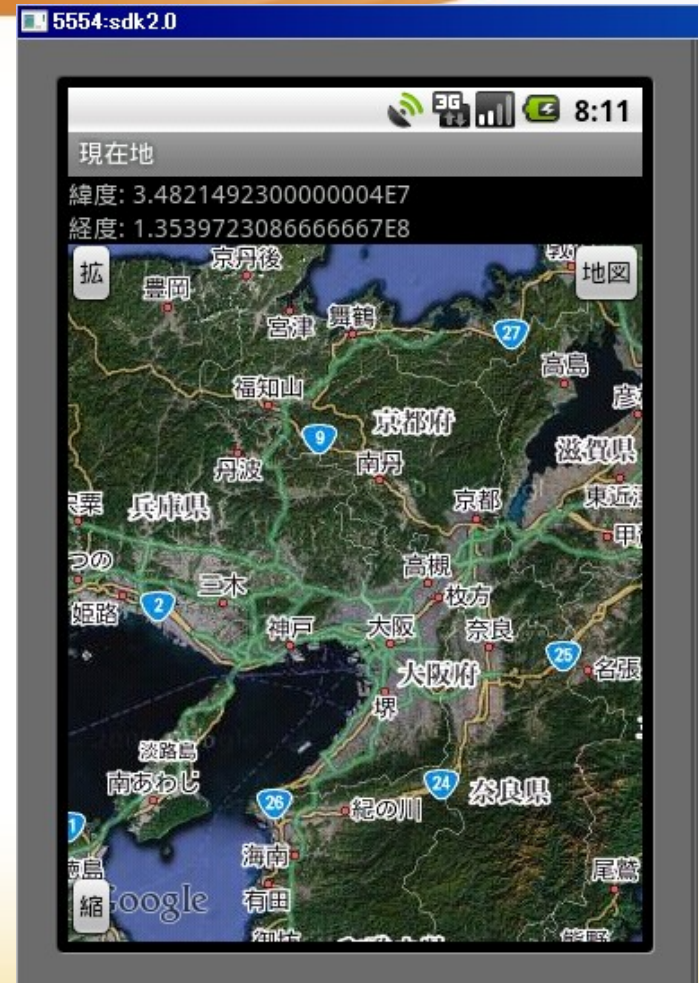
■LocationListener

2/5

レイアウトファイル例

Map部分のみ抜粋

```
<RelativeLayout xmlns:android=
"http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >
    <com.google.android.maps.MapView
        android:id="@+id/myMap"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:enabled="true"
        android:clickable="true"
        android:apiKey="?????"
    />
</RelativeLayout>
```



Brilliant Service

■LocationListener 3/5

日本アンドロイドの会では過去にも同様の勉強会を開催しております。

<http://android.siprop.org/index.php?cmd=list>

勉強会/GPSも参考にしてください。

まずはGPSシステム登録と、ロケーションリスナーの登録をしましょう。

```
public class GPS_NowLocationMapView extends MapActivity implements LocationListener{

    private MapController myMapController;
    private MapView myMap;
    private LocationManager myManager;

    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        /** GPSサービスの登録 */
        myManager =
            (LocationManager) getSystemService(Context.LOCATION_SERVICE);
        /** GPS情報変更割り込み許可 */
        myManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0, this);

        setContentView(R.layout.gps_nowlocationmapview);
        myMap = (MapView)findViewById(R.id.myMap);

        myMapController = myMap.getController();
        myMapController.setZoom(9);
    }
}
```

Brilliant Service

■LocationListener 4/5

onLocationChangedを実装すると位置情報の変化があれば通知されます。

```
public void onLocationChanged(Location location) {  
    GeoPoint myLocation = new GeoPoint(  
        (int) (location.getLatitude()*1E6),  
        (int) (location.getLongitude()*1E6));  
    myMapController.animateTo(myLocation);  
}
```



Brilliant Service

■LocationListener 5/5

今、最新の位置情報が欲しい人は、`getLastKnownLocation()` が使用できる。
ただし、何時間も前のかも知れませので、場合によってはチェックが必要です。

```
Location location = myManager.getLastKnownLocation(LocationManager.GPS_PROVIDER);
if ( location != null ) {
    double latPoint = location.getLatitude();
    double lngPoint = location.getLongitude();

    GeoPoint myLocation = new GeoPoint((int) (latPoint*1E6), (int) (lngPoint*1E6));
    myMapController.setCenter(myLocation);

    if ( location.hasAccuracy() ) {
        float locationAccuracy = location.getAccuracy(); /** GPS 誤差 */
        toast = Toast.makeText(this, "誤差は、 "+Float.toString(locationAccuracy)+"メートルです。", Toast.LENGTH_LONG);
    } else {
        toast = Toast.makeText(this, "GPS が正常に補足できてません。", Toast.LENGTH_LONG);
    }
}
```

■Geocoderでアドレス取得する

- GetFromLocation

緯度、経度からアドレスを返却



Brilliant Service

■Geocoder 1/2

まずは緯度、経度の取得を行ないましょう。

```
MapController myMapController;  
myMapController = myMap.getController();  
-----  
    int text_views[] = {  
        R.id.locationNameText0,  
        R.id.locationNameText1,  
        R.id.locationNameText2,  
        R.id.locationNameText3,  
        R.id.locationNameText4  
    };  
  
    Location location = myManager.getLastKnownLocation("gps");  
    double latPoint = location.getLatitude();  
    double lngPoint = location.getLongitude();
```

■Geocoder 2/2

緯度、経度からGeocoderを使用してアドレス取得

```
Geocoder geocoder = new Geocoder(getApplicationContext(), Locale.JAPAN);

try {
    List<Address> list_address = geocoder.getFromLocation(latPoint, lngPoint, 5);
    if (!list_address.isEmpty()) {
        Address address = list_address.get(0);

        for (int i=0 ; address.getAddressLine(i) != null; i++){
            TextView textview = (TextView)findViewById(text_views[i]);
            textview.setText(address.getAddressLine(i));
        }
    }
} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
```

■Geocoder使用事例 1/

ここでは、数値入力を任意で行い、その数値からGeocoderを使用しアドレスを取得する方法を実施しています。
右記Editorで入力した値をSQLiteにてWriteし、
Map表示用のActivityにて入力された値をReadするという方法を取っています。

入力した値を先に紹介した`getLastKnownLocation()`にて取得した値とイメージしてください。



The screenshot shows an Android application window titled "5554: sdk2.0". The interface has a status bar at the top with "3G", signal strength, and battery icons, and the time "14:16". Below the status bar is a header "位置情報入力" (Location Information Input). The main area contains three input fields: "I D" (empty), "緯度" (Latitude) with the value "37.422006", and "経度" (Longitude) with the value "-122.084095" (highlighted with an orange border). Below these is a "住所" (Address) field (empty). At the bottom are five buttons: "消" (Clear), "変" (Change), "追" (Add), "全消" (Clear All), and "地図" (Map). Below the buttons, the text "1 34.67 133.66 日本 岡山県総社市新本" is displayed.



Brilliant

■Geocoder使用事例 2/

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" >
<TextView android:id="@+id/locationNameLabel"
    android:layout_width="wrap_content" android:layout_height="wrap_content"
    android:textSize="10sp"
    android:text="この場所は、" />
<TextView android:id="@+id/locationNameText0"
    android:layout_width="wrap_content" android:layout_height="wrap_content"
    android:textSize="10sp"/>
<TextView android:id="@+id/locationNameText1"
    android:layout_width="wrap_content" android:layout_height="wrap_content"
    android:textSize="10sp"/>
<TextView android:id="@+id/locationNameText2"
    android:layout_width="wrap_content" android:layout_height="wrap_content"
    android:textSize="10sp"/>
<TextView android:id="@+id/locationNameText3"
    android:layout_width="wrap_content" android:layout_height="wrap_content"
    android:textSize="10sp"/>
<TextView android:id="@+id/locationNameText4"
    android:layout_width="wrap_content" android:layout_height="wrap_content"
    android:textSize="10sp" /></LinearLayout>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent" android:layout_height="fill_parent" >
    <com.google.android.maps.MapView
        android:id="@+id/myMap"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:enabled="true"
        android:clickable="true"
        android:apiKey="???" />
</RelativeLayout>
</LinearLayout>
```



Brilliant Service

■Geocoder使用事例 3/

```
public class GPS_LocationMapView extends MapActivity {
    private static GPS_LocationData locationdatas;
    private static MapController myMapController;
    private static SQLiteDatabase mDB;
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.gps_locationmapview);

        final MapView myMap = (MapView)findViewById(R.id.myMap);
        myMapController = myMap.getController();
        myMapController.setZoom(6);
        myMap.setSatellite(true); /** 衛星写真固定 */

        final Button zoomIn = (Button)findViewById(R.id.buttonZoomIn);
        zoomIn.setOnClickListener(new Button.OnClickListener() {
            public void onClick(View v) {ZoomIn(myMap, myMapController);}
        });
        final Button zoomOut = (Button)findViewById(R.id.buttonZoomOut);
        zoomOut.setOnClickListener(new Button.OnClickListener() {
            public void onClick(View v) {ZoomOut(myMap, myMapController);}
        });
    }
}
```



Brilliant Service



■Geocoder使用事例 4/

```
@Override
protected void onResume() {
    super.onResume();int text_views[] = {
        R.id.locationNameText0,
        R.id.locationNameText1,
        R.id.locationNameText2,
        R.id.locationNameText3,
        R.id.locationNameText4
    };

    locationdatas = new GPS_LocationData(this);
    try {
        /** 書き込み用にオープン */
        mDB = locationdatas.getWritableDatabase();
    } catch (SQLiteException err) {
        /** Open時に例外エラー発生 エラー内容のLog出力 */
        Log.e("hellosqlite",err.toString());
        mDB = locationdatas.getReadableDatabase();/** リードオンリー */
    }
    Cursor c = mDB.rawQuery(GET_VIEW_POINT, null);
```

(次ページに続く)



Brilliant Service

■Geocoder使用事例 5/

```
if (c.moveToFirst()) {
    String str_id = c.getString(0);
    double latPoint = c.getDouble(1);
    double lngPoint = c.getDouble(2);

    GeoPoint myLocation =
    new GeoPoint((int) (latPoint*1E6), (int) (lngPoint*1E6));

    myMapController.setCenter(myLocation);

    Geocoder geocoder = new Geocoder(getBaseContext(),
                                     Locale.JAPAN);

    try {
        List<Address> list_address =
        geocoder.getFromLocation(latPoint, lngPoint, 5);
        if (!list_address.isEmpty()) {
            Address address = list_address.get(0);
            for (int i=0; address.getAddressLine(i) != null; i++) {
                TextView textview =
                    (TextView) findViewById(text_views[i]);
                textview.setText(address.getAddressLine(i));
            }
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```



Brilliant Service

■Geocodingサービス

- 緯度、経度から、アドレスは引っ張ってこれたけど逆はダメ??
⇒AndroidAPIを調査したけど、なさそう。
⇒MAP APIがあり、JavaScriptで使用事例が一杯ある。
- WebViewでScriptを埋めたHTMLを表示させれば良いだけ! ?



Brilliant Service

■WebView 1/4 layout

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

    <WebView
        android:id="@+id/web_view"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
    />

</LinearLayout>
```



Brilliant Service

■ WebView 2/4 Android-Java

```
package fujii.BS_Locator;

import android.app.Activity;
import android.os.Bundle;
import android.webkit.WebView;

public class GPS_AddressSearch extends Activity {
    WebView wv;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        wv = new WebView(this);
        wv.getSettings().setJavaScriptEnabled(true);

        wv.setId(R.id.web_view);
        wv.loadUrl("file:///android_asset/map1_1.html");
        setContentView(wv);
    }
}
```



Brilliant Service



■ WebView 3/4 内部HTML

<http://www.ajaxtower.jp/googlemaps/geocoding/index1.html>
のサンプルコードをそのまま使用させて頂きました。

```
<!DOCTYPE html "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8"/>
    <title>サンプル：住所から座標を取得</title>
    <script src="http://maps.google.com/maps?
file=api&v=2&sensor=false"
      type="text/javascript" charset="utf-8"></script>
    <script src="./code1_1.js" type="text/javascript"></script>
  </head>
  <body onload="initialize()" onunload="GUnload()">
    <div id="map_canvas" style="width: 400px; height: 400px"></div>

    <form>
      <p>
        <input type="text" id="address" size="30" value="住所を入力" />
        <input type="button" id="addressbtn" value="座標取得"
onclick="moveAddress()" />
      </p>
    </form>

  </body>
</html>
```



Brilliant Service

■ WebView 4/4 JavaScript

<http://www.ajaxtower.jp/googlemaps/geocoding/index1.html>
のサンプルコードをそのまま使用させて頂きました。

```
/* Google Maps APIサンプル */
var map;
var geocoder;

function initialize() {
    if (GBrowserIsCompatible()) {
        map = new GMap2(document.getElementById("map_canvas"));
        map.setCenter(new GLatLng(36.004673, 137.351074), 5);

        geocoder = new GClientGeocoder();
    }
}

function moveAddress() {
    var address = document.getElementById("address").value;
    geocoder.getLatLng(address, moveTo);
}

function moveTo(latlng) {
    if (latlng) {
        map.setCenter(latlng, 15);

        map.clearOverlays();
        var marker = new GMarker(latlng);
        map.addOverlay(marker);
    } else {
        alert("住所から緯度経度に変換できません");
    }
}
```



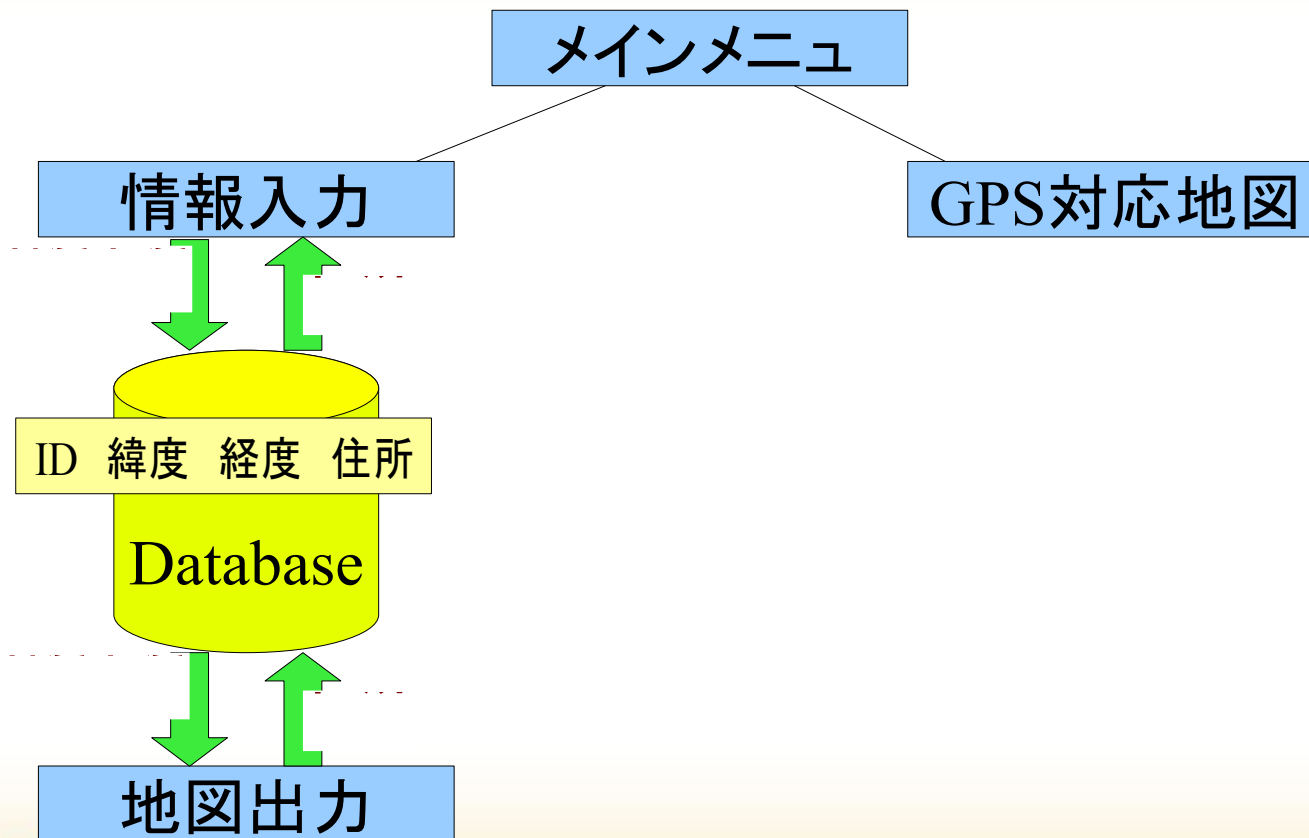
Brilliant Service

■実行結果



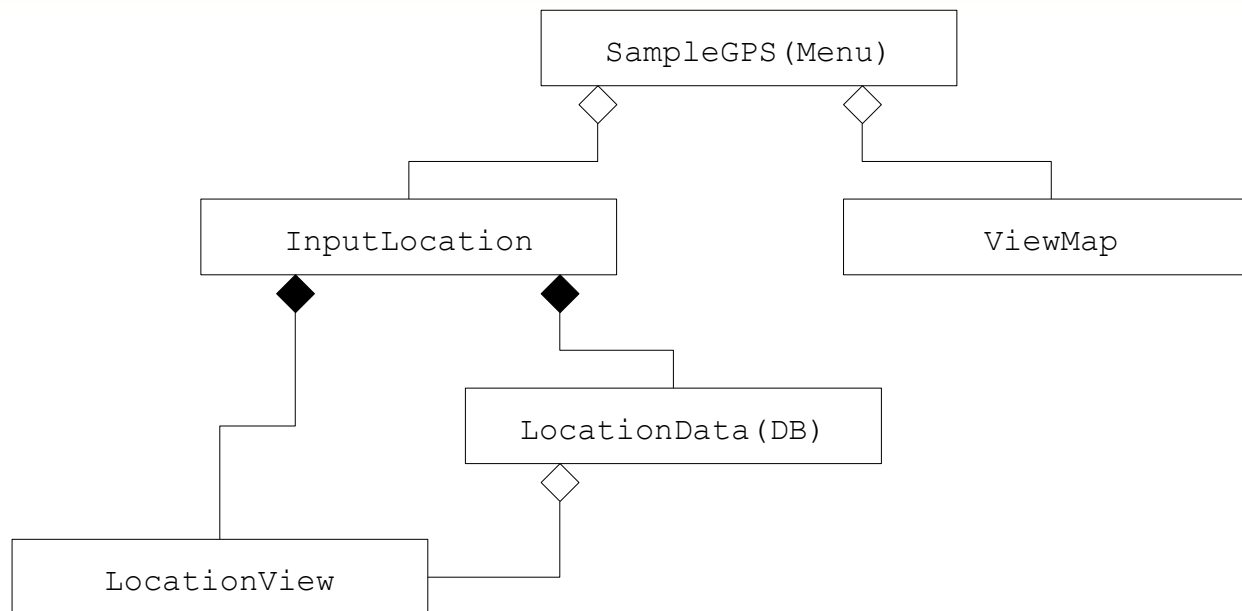
Brilliant Service

■Activity構成イメージ 1/2



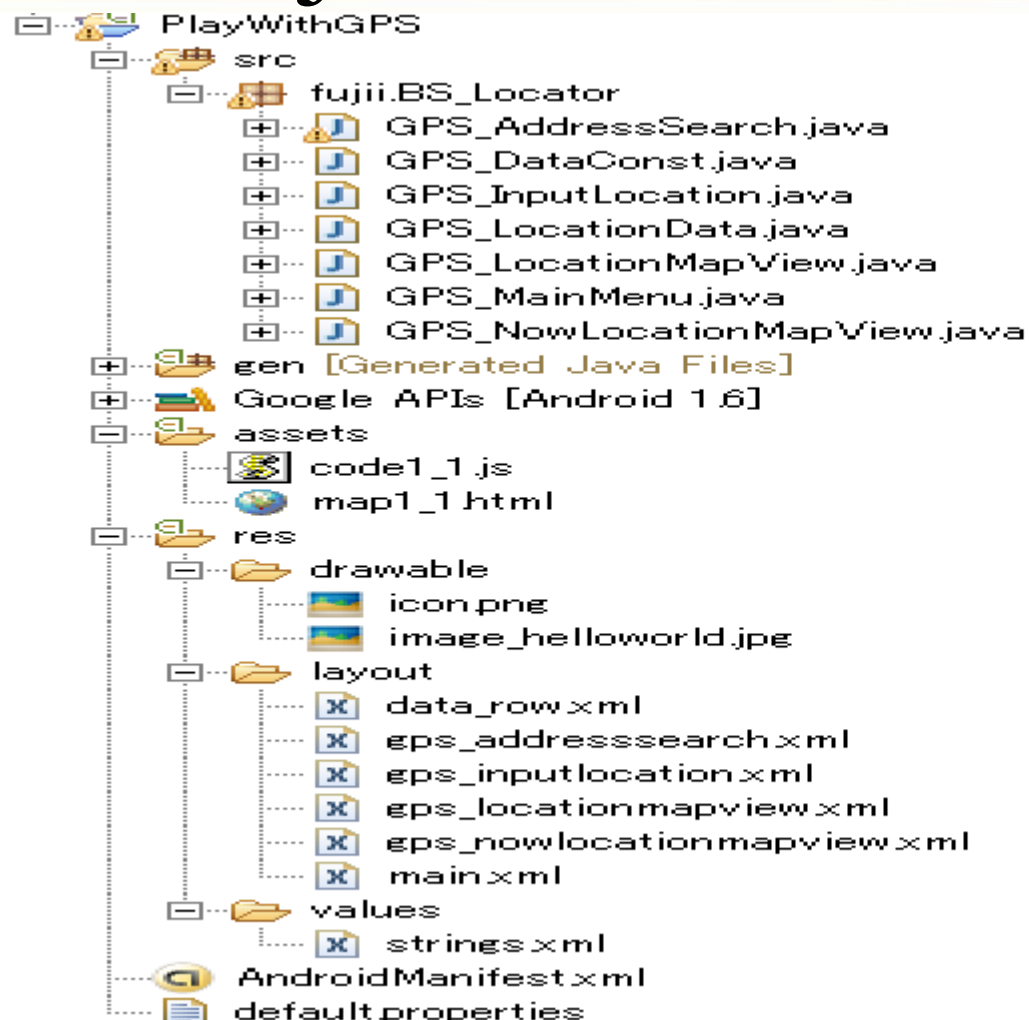
Brilliant Service

■Activity構成イメージ 2/2



Brilliant Service

■Project構成



src)

1. GeocodingMAP API,WebView使用
2. 当プロジェクト使用のSQLite カラム定義
3. 緯度、経度を入力するEditor
4. 当プロジェクト使用のSQLiteヘルパー
5. 3の緯度、経度を元に地図表示
6. 当プロジェクトのMainLauncher
7. GPS情報から現在地表示

assets)

1. JavaScript
2. MapViewでLoadするHTMLファイル

layout)

1. GPS_InputLocationで使用するListレイアウト
2. GPS_AddressSearchのレイアウト
3. GPS_InputLocationのレイアウト
4. GPS_LocationMapViewのレイアウト
5. GPS_NowLocationMapViewのレイアウト

■これから始める方へ

- 「初めてのアンドロイド」を買って、サンプルプログラムをどんどん、とりあえず動かしましょう
- サンプルプログラムを自分なりに変えてみましょう。色とか、レイアウトとか。
- 溜まったサンプルプログラム同士を合体させてみましょう。
- わからないことは、Webで調べるか、日本アンドロイドの会に聞きましょう。とにかく、お！！と思うページが見つかるかも。



Brilliant Service

■これから始める方へ

- 最初からオリジナルも良いですが、ふと立ち止まってください。
- 出来る人のを真似して、自分のものにすればいいんです。
- 日本人の得意技です。
- そのうちオリジナルが出来あがります。
- 気楽にやりましょう。



Brilliant Service

■ありがとうございました



Brilliant Service