

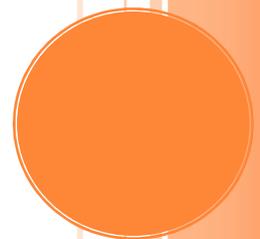
# Intent を発行してみよう

*Android* アプリケーション開発 ハンズオンセミナー

Intent とは、アプリケーションから、新しいアクティビティを起動するためのパラメータです。起動するアクティビティは、同じアプリケーション内のアクティビティでも、他のアプリケーション内のアクティビティでも指定することができます。他のアプリケーションのアクティビティを指定するためには、アクションと、Uri という二つのパラメータを指定します。Android では、あらかじめ規定された Intent にたいする動作が組み込まれています。

日本 Android の会 木南英夫

2009/08/06



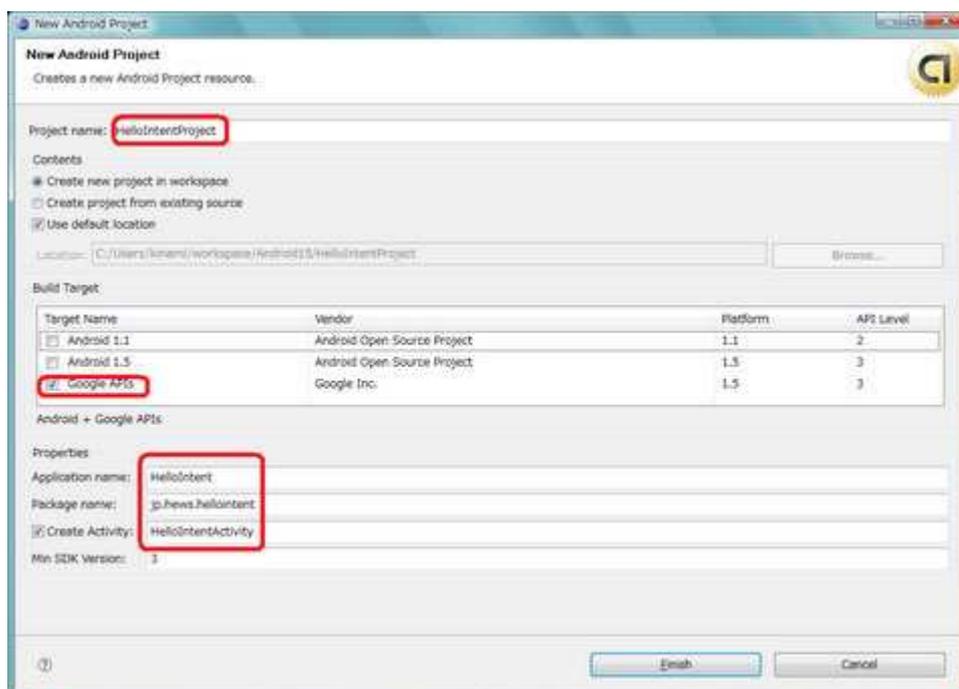
# インテントを発行してみよう

Android アプリケーション開発 ハンズオンセミナー

## プロジェクトを作成する

必要に応じて、File > New > Android Project で新規のプロジェクトを作成します。

ここでは、以下のようなプロジェクトを作成してみます。



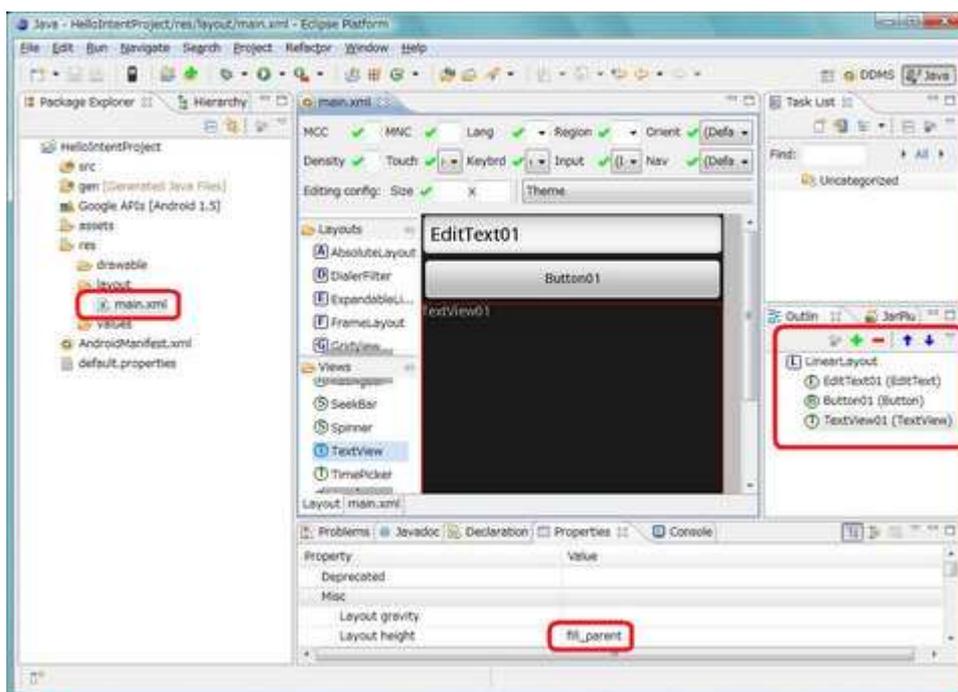
Project Name	HelloIntentProject
Build Target	Google APIs
Application Name	HelloIntent
Package Name	jp.hews.hellointent
Create Activity	HelloIntentActivity
Min SDK Version	3 (Build Target を指定すると自動的に設定される)

## 画面を定義する

画面にボタンを配置するには、レイアウトを定義した XML ファイルを編集します。

まず、レイアウトを定義した `res/layout/main.xml` を開いて、画面上にボタンを配置します。

この画面でボタンを配置するには、GUI で挿入する方法と、下部の「main.xml」タブから直接 XML を編集する方法があります。ここでは、GUI を用いて定義します。



- LinearLayout の最初に EditText(id:EditText01)を挿入します。layout\_width を fill\_parent に変更します。
- 2番目に Button(id:Button01)を挿入します。layout\_width を fill\_parent に変更します。
- 3番目に TextView(id:TextView01)を挿入します。layout\_width と layout\_height を fill\_parent に変更します。

## インテントを作成して発行するメソッドを定義する

`EditText` から文字列を取り出して、`ACTION_VIEW` のインテントを作成して、`startActivity` を呼び出します。

判定できない `URI` の場合には、エラーの内容をテキストビューに表示します。

以下のメソッドを `HelloIntentActivity.java` に定義してみましょう。

```
private void sendIntent() {
    try {
        EditText et = (EditText)findViewById(R.id.EditText01);
        Intent i = new Intent(Intent.ACTION_VIEW,
            Uri.parse(et.getText().toString()));
        startActivity(i);
    } catch (Exception e) {
        TextView t = (TextView)findViewById(R.id.TextView01);
        t.setText(e.toString());
    }
}
```

- `EditText` の内容を `Uri` に変換して、インテントを作成します。
- 作成したインテントを引数にして `startActivity` を呼び出します。
- エラー発生時は、`TextView` にエラーを表示します。
- 必要なクラスのインポートは、`Ctrl-Shift-O` で挿入できます。

## ボタンにリスナーを登録する

ボタンが押された時に、定義した `sendIntent` を呼び出すリスナーを登録します。

```
public class HelloIntentActivity extends Activity implements View.OnClickListener {  
  
    @Override  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);  
  
        Button button = (Button)findViewById(R.id.Button01);  
        button.setOnClickListener(this);  
    }  
  
    public void onClick(View arg0) {  
        sendIntent();  
    }  
}
```

- `HelloIntentActivity.java` を開いて、`onCreate` メソッドでボタンを取り出して、リスナーを登録します。
- ボタンリスナーを作成して、リスナー内の `onClick` メソッドで次で定義する `sendIntent` メソッドを呼び出します。
- `setOnClickListener` でリスナーを登録します。

## URIを入力する

URIを入力して、あらかじめ組み込まれている Uri に対応するアクティビティを起動してみます。

Uri	動作	例
http://web_address	ブラウザを起動する	http://www.google.com/
tel:phone_number	ダイヤル画面を表示します	tel:123456
geo:latitude,longitude	地図を表示します	geo:37,137
content://contacts/people	コンタクトリストを表示します	content://contacts/people/1



## ソースコード

作成したソースコードは、以下のようになっています。

```
package jp.hews.hellointent;

import android.app.Activity;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class HelloIntentActivity extends Activity implements View.OnClickListener {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        Button button = (Button)findViewById(R.id.Button01);
        button.setOnClickListener(this);
    }

    public void onClick(View arg0) {
        sendIntent();
    }

    private void sendIntent() {
        try {
            EditText et = (EditText)findViewById(R.id.EditText01);
            Intent i = new Intent(Intent.ACTION_VIEW,
                Uri.parse(et.getText().toString()));
            startActivity(i);
        } catch (Exception e) {
            TextView t = (TextView)findViewById(R.id.TextView01);

```

```
t.setText(e.toString());  
    }  
}  
}}
```

## 演習問題

- ActrionView 以外のアクションを画面から選択できるようにしてみましょう。
- 自分自身で定義したアクティビティをインテントで起動してみましょう。