

## Android Explicit Intent Example

Following example shows the functionality of an Android Intent to launch various Android built-in applications.

Now we will learn how to transfer **data**. We will make the simplest application. On the first screen we will enter our name and surname and the second screen will display this data. We will transfer **data inside Intent**.

Main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical">
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:gravity="center_horizontal"
        android:text="Input your Name">
    </TextView>
    <TableLayout
        android:id="@+id/tableLayout1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:stretchColumns="1">
        <TableRow
            android:id="@+id/tableRow1"
            android:layout_width="match_parent"
            android:layout_height="wrap_content">
            <TextView
                android:id="@+id/textView1"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:text="First Name">
            </TextView>
            <EditText
                android:id="@+id/etFName"
                android:layout_width="match_parent"
```

```

        android:layout_height="wrap_content"
        android:layout_marginLeft="5dp">
        <requestFocus>
        </requestFocus>
    </EditText>
</TableRow>
<TableRow
    android:id="@+id/tableRow2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <TextView
        android:id="@+id/textView2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Last Name">
    </TextView>
    <EditText
        android:id="@+id/etLName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="5dp">
    </EditText>
</TableRow>
</TableLayout>
<Button
    android:id="@+id/btnSubmit"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center_horizontal"
    android:text="Submit">
</Button>
</LinearLayout>

```

### **MainActivity.java:**

```

package .....; // default come in your system
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;

```

```

import android.widget.Button;
import android.widget.EditText;

public class MainActivity extends Activity implements OnClickListener {

    EditText etFName;
    EditText etLName;

    Button btnSubmit;

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

        etFName = (EditText) findViewById(R.id.etFName);
        etLName = (EditText) findViewById(R.id.etLName);

        btnSubmit = (Button) findViewById(R.id.btnSubmit);
        btnSubmit.setOnClickListener(this);

    }

    @Override
    public void onClick(View v) {
        Intent intent = new Intent(this, ViewActivity.class);
        intent.putExtra("fname", etFName.getText().toString());
        intent.putExtra("lname", etLName.getText().toString());
        startActivity(intent);
    }
}

```

Define input fields and a button. We assign button a listener - Activity (this). Let's observe **onClick** method implementation. We create **Intent** using the class, not action. If you remember we started to get acquainted with Intent this way. I will remind - it means that the system will look through the manifest-file of our application and will display an Activity if it finds one with the same class.

ViewActivity is not created yet, that's why the code will be underlined in red. However, we can still save the file. We will create this Activity a little bit later and the error will disappear.

So the Intent has been created, let's observe the code further. `putExtra` method is used. It has lots of varieties and is similar to `put` method for **Map**, that is, it **adds** a key-value **pair** to the object. First parameter is **key**(name), the second - **value**. We've put two objects with names *fname* and *lname* into Intent. *fname* contains the value of `etFName` field, *lname* - value of the `etLName` field. The only thing left to do is to **send** the equipped Intent using **startActivity** method.

Let's create the second Activity now. Let's name it **ViewActivity**.

Create layout-file **view.xml** for it:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <TextView
        android:id="@+id/tvView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_marginTop="20dp"
        android:text="TextView"
        android:textSize="20sp">
    </TextView>
</LinearLayout>
```

It is just a TextView here which will display the incoming data.

Create **ViewActivity** class. Write this code:

```
package ru.startandroid.develop.p0281intentextras;
```

```
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.widget.TextView;
```

```
public class ViewActivity extends Activity {
```

```
TextView tvView;
```

```
@Override
```

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.view);
```

```
    tvView = (TextView) findViewById(R.id.tvView);
```

```
    Intent intent = getIntent();
```

```
    String fName = intent.getStringExtra("fname");  
    String lName = intent.getStringExtra("lname");
```

```
    tvView.setText("Your name is: " + fName + " " + lName);
```

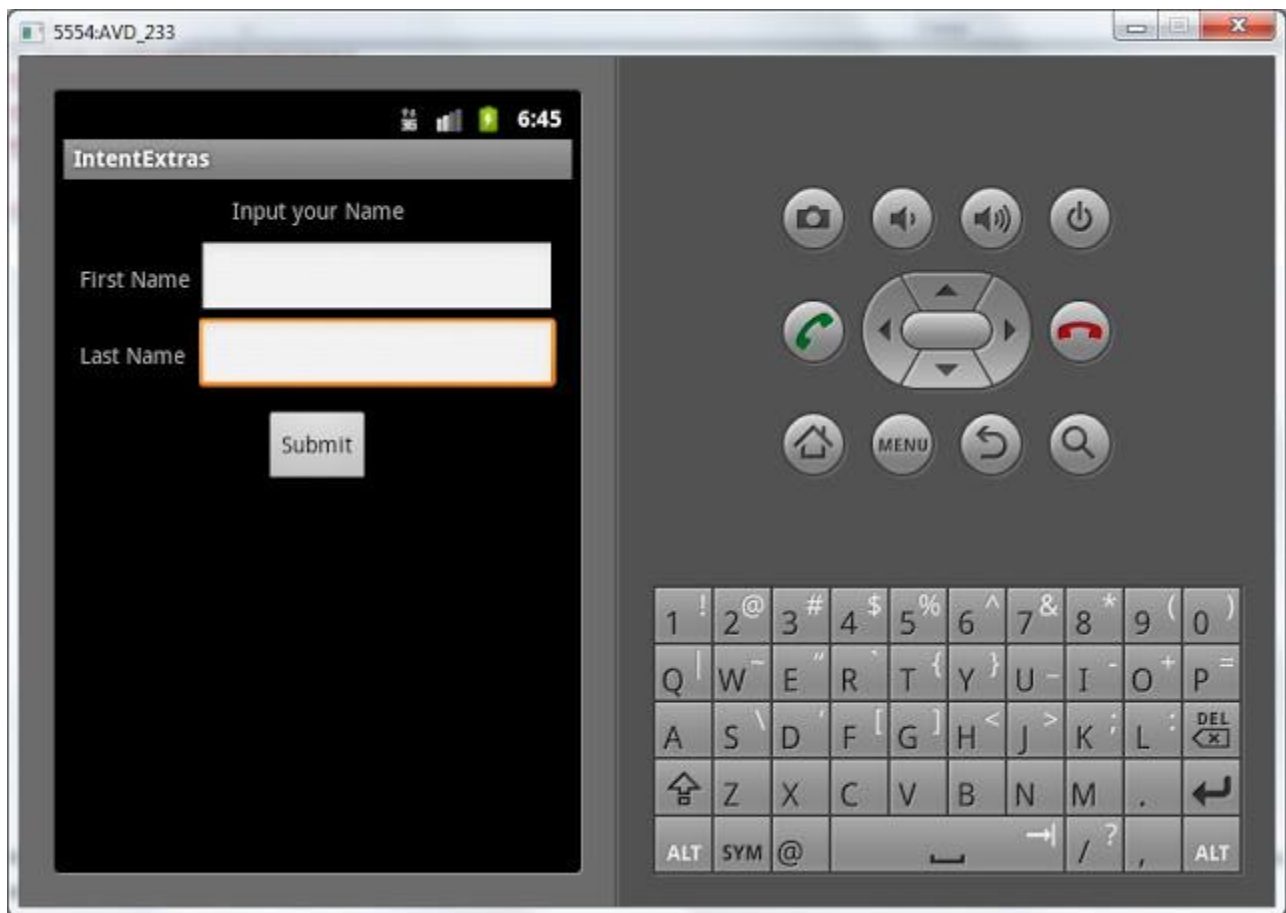
```
}
```

```
}
```

Find TextView, then receive **Intent** and **extract** String objects with names *fName* and *lname* from it. These are the same objects, which we put inside in MainActivity.java. Form the output string into the TextView using the received data.

Don't forget to register Activity inside manifest. This time we don't need any intent filters as we know the name of Activity class and use the explicit invocation. Save everything and run the application.

You will see the following screen:



Fill in the fields with whatever you wish. I will write *John* in the **First Name** field and *Smith* in the **Last Name** field.

Press Submit:

