Shared Preferences

Introduction

❖ The preferences mechanism allows us to store and retrieve key value pairs of primitive data types. This mechanism is typically used to store application preferences, such as the default text font to be loaded whenever the application starts.

❖ Using this mechanism the preferences will be shared between all activities that belong to our application.

The `SharedPreferences` Interface

❖ This interface includes various methods we can use for maintaining the application shared preferences.

❖ For each set of preference there is a single instance of this class that all clients share.

❖ Modifying preferences must be done through a `SharedPreferences.Editor` object in order to ensure the preference values remain in a consistent state.

The `SharedPreferences` Interface

❖ Extending Activity we can call the `getSharedPreferences`

method passing over the file name that holds the preferences.

```
...
SharedPreferences settings = getSharedPreferences("myfile", 0);
boolean vib = settings.getBoolean("vibration_mode", false);
...
SharedPreferences settings = getSharedPreferences("myfile", 0);
SharedPreferences.Editor editor = settings.edit();
editor.putBoolean("vibration_mode", vib);
editor.commit();
...
```

PreferencesFragment

❖ We can easily create a simple preferences setting screen using this class.

PreferencesFragment

```java
public class MainActivity extends Activity {
    private Button button;
    private TextView textview;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);
    button = (Button) findViewById(R.id.bt);
    textview = (TextView) findViewById(R.id.textview);
    button.setOnClickListener(new OnClickListener() {
        @Override
        public void onClick(View arg0) {
            Intent intent = new Intent(getApplicationContext(),
            MyPreferencesActivity.class);
            startActivityForResult(intent, 0);
        }
    });
}
```

PreferencesFragment

```
@Override
protected void onActivityResult(int requestCode,
    int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    SharedPreferences sharedPreferences = PreferenceManager
        .getDefaultSharedPreferences(this);
    Boolean prefA = sharedPreferences.getBoolean("CHECKBOX_A", false);
    Boolean prefB = sharedPreferences.getBoolean("CHECKBOX_B", false);
    textview.setText("a="+ prefA.toString()+" b="+prefB.toString());
    }
}
```

PreferencesFragment

```
package com.lifemichael.preferencesactivityexample;

import android.app.Activity;
import android.app.FragmentManager;
import android.app.FragmentTransaction;
import android.os.Bundle;

public class MyPreferencesActivity extends Activity {
 @Override
 protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    MyPreferencesFragment prefFragment = new MyPreferencesFragment();
    FragmentManager fragmentManager = getFragmentManager();
    FragmentTransaction fragmentTransaction =
        fragmentManager.beginTransaction();
    fragmentTransaction.replace(android.R.id.content, prefFragment);
    fragmentTransaction.commit();
 }
}
```

PreferencesFragment

```java
package com.lifemichael.preferencesactivityexample;

import android.os.Bundle;
import android.preference.PreferenceFragment;

public class MyPreferencesFragment extends PreferenceFragment {

 @Override
 public void onCreate(Bundle savedInstanceState) {
   super.onCreate(savedInstanceState);
   addPreferencesFromResource(R.xml.preference);
 }
}
```

PreferencesFragment

```xml
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen
xmlns:android="http://schemas.android.com/apk/res/android" >

    <PreferenceCategory android:title="PreferenceCategory A" >
        <CheckBoxPreference
            android:key="CHECKBOX_A"
            android:summary="summary for a checkbox"
            android:title="A Title" />
        <CheckBoxPreference
            android:key="CHECKBOX_B"
            android:summary="summary for b checkbox"
            android:title="B Title" />
    </PreferenceCategory>

</PreferenceScreen>
```
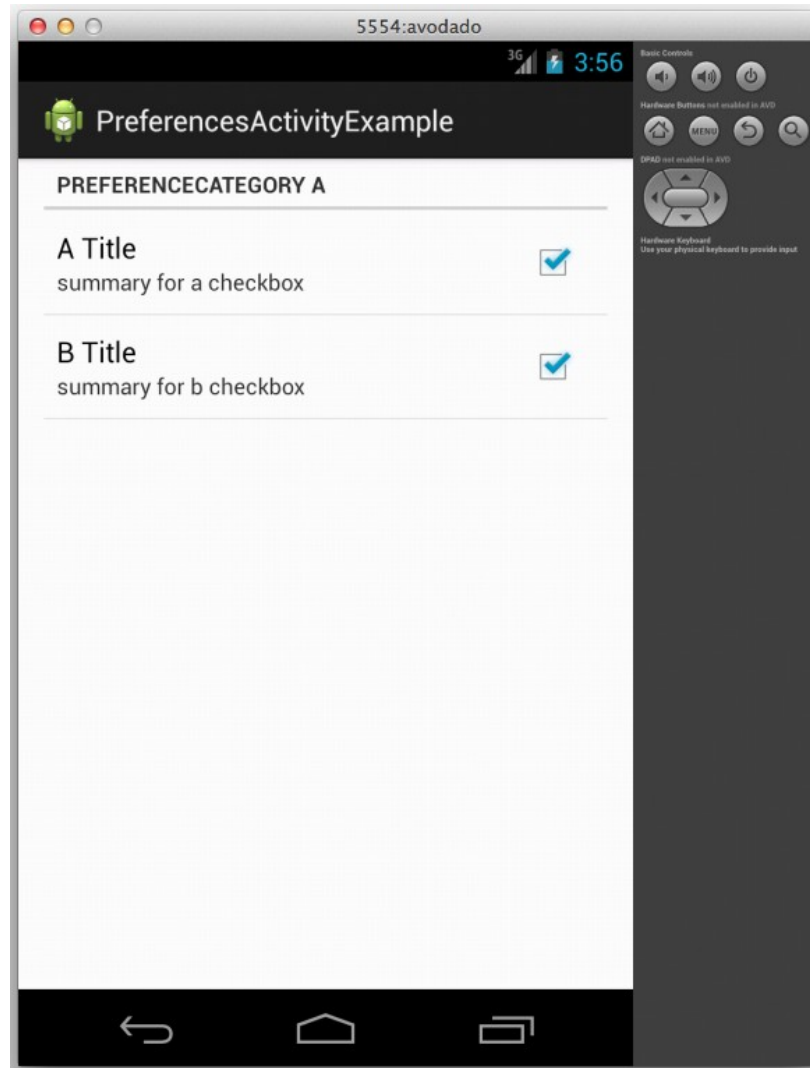
PreferencesFragment

Shared Preferences

© 2008 Haim Michael

Introduction

❖ The preferences mechanism allows us to store and retrieve
   key value pairs of primitive data types. This mechanism is
   typically used to store application preferences, such as the
   default text font to be loaded whenever the application starts.

❖ Using this mechanism the preferences will be shared between
   all activities that belong to our application.

© 2008 Haim Michael

The `SharedPreferences` Interface

❖ This interface includes various methods we can use for maintaining the application shared preferences.

❖ For each set of preference there is a single instance of this class that all clients share.

❖ Modifying preferences must be done through a `SharedPreferences.Editor` object in order to ensure the preference values remain in a consistent state.

The `SharedPreferences` Interface

❖ Extending Activity we can call the `getSharedPreferences` method passing over the file name that holds the preferences.

```
...
SharedPreferences settings = getSharedPreferences("myfile", 0);
boolean vib = settings.getBoolean("vibration_mode", false);
...
SharedPreferences settings = getSharedPreferences("myfile", 0);
SharedPreferences.Editor editor = settings.edit();
editor.putBoolean("vibration_mode", vib);
editor.commit();
...
```

PreferencesFragment

❖ We can easily create a simple preferences setting screen
using this class.

© 2008 Haim Michael

PreferencesFragment

```
public class MainActivity extends Activity {
    private Button button;
    private TextView textview;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);
    button = (Button) findViewById(R.id.bt);
    textview = (TextView) findViewById(R.id.textview);
    button.setOnClickListener(new OnClickListener() {
        @Override
        public void onClick(View arg0) {
            Intent intent = new Intent(getApplicationContext(),
            MyPreferencesActivity.class);
            startActivityForResult(intent, 0);
        }
    });
}
```

PreferencesFragment

```
@Override
protected void onActivityResult(int requestCode,
    int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    SharedPreferences sharedPreferences = PreferenceManager
        .getDefaultSharedPreferences(this);
    Boolean prefA = sharedPreferences.getBoolean("CHECKBOX_A", false);
    Boolean prefB = sharedPreferences.getBoolean("CHECKBOX_B", false);
    textview.setText("a="+ prefA.toString()+" b="+prefB.toString());
    }
}
```

© 2008 Haim Michael

PreferencesFragment

```
package com.lifemichael.preferencesactivityexample;

import android.app.Activity;
import android.app.FragmentManager;
import android.app.FragmentTransaction;
import android.os.Bundle;

public class MyPreferencesActivity extends Activity {
 @Override
 protected void onCreate(Bundle savedInstanceState) {
     super.onCreate(savedInstanceState);
     MyPreferencesFragment prefFragment = new MyPreferencesFragment();
     FragmentManager fragmentManager = getFragmentManager();
     FragmentTransaction fragmentTransaction =
         fragmentManager.beginTransaction();
     fragmentTransaction.replace(android.R.id.content, prefFragment);
     fragmentTransaction.commit();
 }
}
```

© 2008 Haim Michael

PreferencesFragment

```
package com.lifemichael.preferencesactivityexample;

import android.os.Bundle;
import android.preference.PreferenceFragment;

public class MyPreferencesFragment extends PreferenceFragment {

 @Override
 public void onCreate(Bundle savedInstanceState) {
  super.onCreate(savedInstanceState);
  addPreferencesFromResource(R.xml.preference);
 }
}
```

© 2008 Haim Michael

PreferencesFragment

```xml
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen
xmlns:android="http://schemas.android.com/apk/res/android" >

    <PreferenceCategory android:title="PreferenceCategory A" >
        <CheckBoxPreference
            android:key="CHECKBOX_A"
            android:summary="summary for a checkbox"
            android:title="A Title" />
        <CheckBoxPreference
            android:key="CHECKBOX_B"
            android:summary="summary for b checkbox"
            android:title="B Title" />
    </PreferenceCategory>

</PreferenceScreen>
```
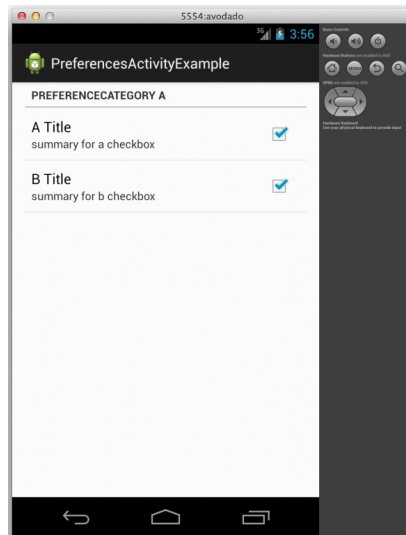
PreferencesFragment

© 2008 Haim Michael