

1 .Shared Preferences

Android provides many ways of storing data of an application. One of this way is called Shared Preferences. Shared Preferences allow you to save and retrieve data in the form of key,value pair.

In order to use shared preferences, you have to call a method `getSharedPreferences()` that returns a `SharedPreferences` instance pointing to the file that contains the values of preferences.

```
SharedPreferences sharedPreferences = getSharedPreferences(MyPREFERENCES, Context.MODE_PRIVATE);
```

The first parameter is the key and the second parameter is the MODE. Apart from private there are other modes available that are listed below –

Sr.No	Mode & description
	MODE_APPEND
1	This will append the new preferences with the already existing preferences
	MODE_ENABLE_WRITE_AHEAD_LOGGING
2	Database open flag. When it is set , it would enable write ahead logging by default
	MODE_MULTI_PROCESS
3	This method will check for modification of preferences even if the sharedpreference instance has already been loaded
	MODE_PRIVATE
4	By setting this mode, the file can only be accessed using calling application
	MODE_WORLD_READABLE
5	This mode allow other application to read the preferences
	MODE_WORLD_WRITEABLE
6	This mode allow other application to write the preferences

You can save something in the sharedpreferences by using `SharedPreferences.Editor` class. You will call the `edit` method of `SharedPreferences` instance and will receive it in an editor object. Its syntax is –

```
Editor editor = sharedPreferences.edit();
editor.putString("key", "value");
editor.commit();
```

Apart from the `putString` method , there are methods available in the editor class that allows manipulation of data inside shared preferences. They are listed as follows –

Sr. NO	Mode & description
	apply()
1	It is an abstract method. It will commit your changes back from editor to the sharedPreferences object you are calling
	clear()
2	It will remove all values from the editor
	remove(String key)
3	It will remove the value whose key has been passed as a parameter
	putLong(String key, long value)
4	It will save a long value in a preference editor
	putInt(String key, int value)
5	It will save a integer value in a preference editor
	putFloat(String key, float value)
6	It will save a float value in a preference editor

Example

This example demonstrates the use of the Shared Preferences. It display a screen with some text fields, whose value are saved when the application is closed and brought back when it is opened again.

To experiment with this example, you need to run this on an actual device on after developing the application according to the steps below –

Steps	Description
1	You will use Android studio to create an Android application under a package com.example.sairamkrishna.myapplication.
2	Modify src/MainActivity.java file to add progress code to display the spinning progress dialog.
3	Modify res/layout/activity_main.xml file to add respective XML code.
4	Run the application and choose a running android device and install the application on it and verify the results.

Following is the content of the modified **MainActivity.java**.

```
package com.example.sairamkrishna.myapplication;

import android.content.Context;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
```

```

import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    EditText ed1,ed2,ed3;
    Button b1;

    public static final String MyPREFERENCES = "MyPrefs" ;
    public static final String Name = "nameKey";
    public static final String Phone = "phoneKey";
    public static final String Email = "emailKey";

    SharedPreferences sharedPreferences;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        ed1=(EditText)findViewById(R.id.editText);
        ed2=(EditText)findViewById(R.id.editText2);
        ed3=(EditText)findViewById(R.id.editText3);

        b1=(Button)findViewById(R.id.button);
        sharedPreferences = getSharedPreferences(MyPREFERENCES,
Context.MODE_PRIVATE);

        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String n  = ed1.getText().toString();
                String ph  = ed2.getText().toString();
                String e  = ed3.getText().toString();

                SharedPreferences.Editor editor = sharedPreferences.edit();

                editor.putString(Name, n);
                editor.putString(Phone, ph);
                editor.putString(Email, e);
                editor.commit();

                Toast.makeText(MainActivity.this,"Thanks",Toast.LENGTH_LONG).show();
            }
        });
    }
}

```

Following is the content of the modified main activity fileres/layout/activiy_main.xml.

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"

```

```
        android:paddingBottom="@dimen/activity_vertical_margin"
tools:context=".MainActivity">
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Shared Preference "
    android:id="@+id/textView"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:textSize="35dp" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Tutorials Point"
    android:id="@+id/textView2"
    android:layout_below="@+id/textView"
    android:layout_centerHorizontal="true"
    android:textSize="35dp"
    android:textColor="#ff16ff01" />
```

```
<EditText
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/editText"
    android:layout_below="@+id/textView2"
    android:layout_marginTop="67dp"
    android:hint="Name"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
```

```
<EditText
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/editText2"
    android:layout_below="@+id/editText"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:hint="Pass" />
```

```
<EditText
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/editText3"
    android:layout_below="@+id/editText2"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:hint="Email" />
```

```
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Save"
    android:id="@+id/button"
```

```

        android:layout_below="@+id/editText3"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="50dp" />

</RelativeLayout>

```

Following is the content of the modified content of file **res/values/strings.xml**.

```

<resources>
    <string name="app_name">My Application</string>
</resources>

```

Following is the content default file **AndroidManifest.xml**.

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.sairamkrishna.myapplication" >

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >


        <activity
            android:name=".MainActivity"
            android:label="@string/app_name" >

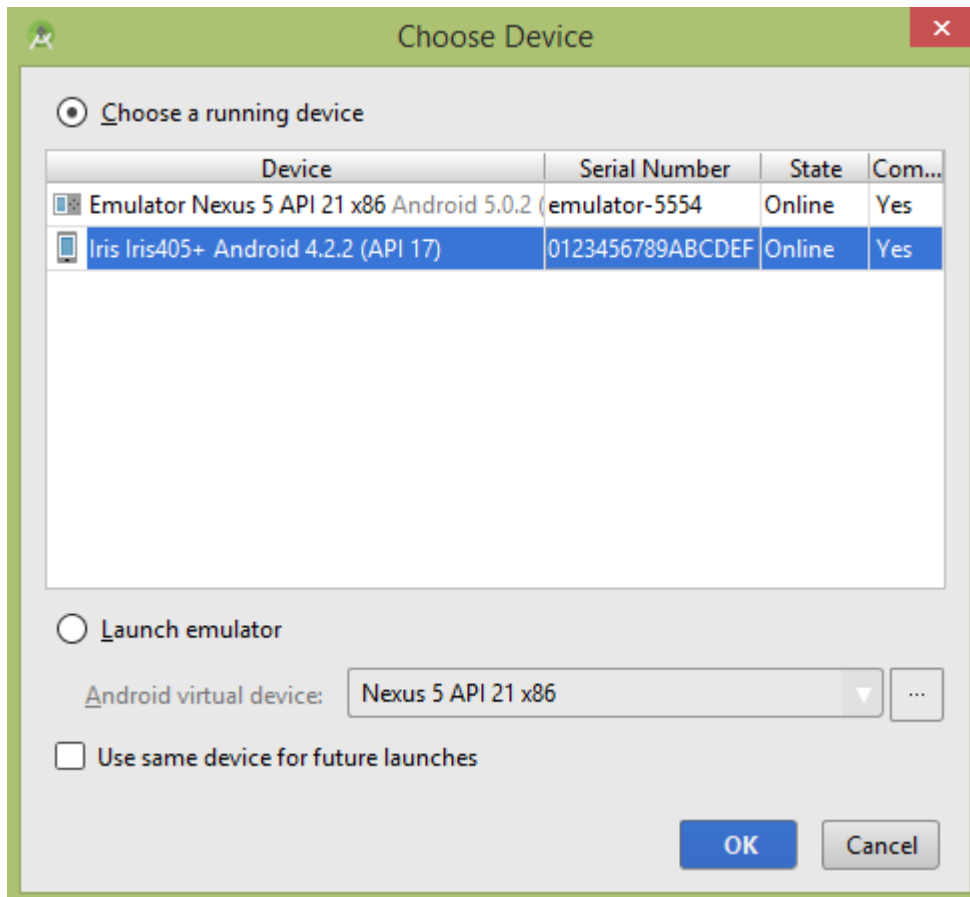
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>

        </activity>

    </application>
</manifest>

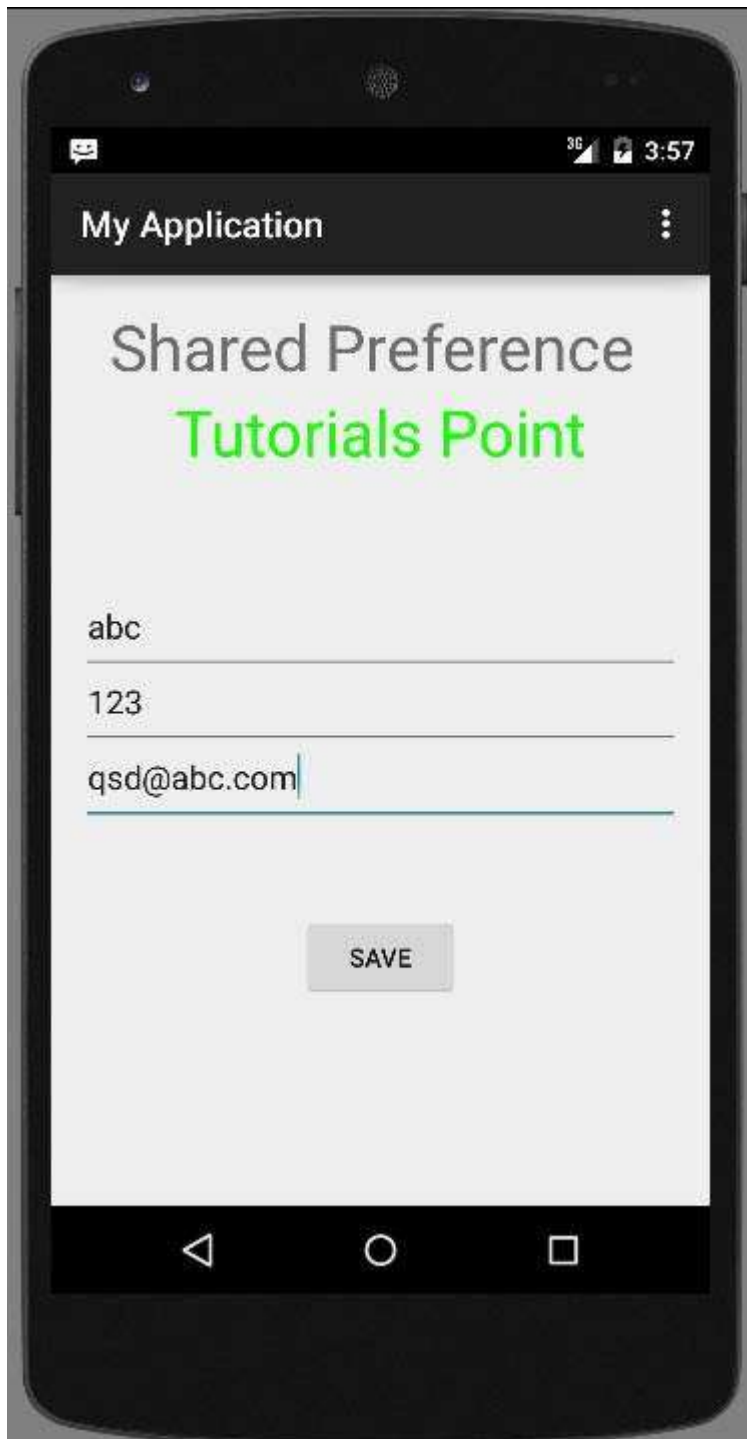
```

Let's try to run your application. I assume you have connected your actual Android Mobile device with your computer. To run the app from Android studio, open one of your project's activity files and click Run  icon from the toolbar. Before starting your application, Android studio will display following window to select an option where you want to run your Android application.



Select your mobile device as an option and then check your mobile device which will display following screen –

Now just put in some text in the field. Like i put some random name and other information and click on save button.



Now when you press save button, the text will be saved in the shared preferences. Now press back button and exit the application. Now open it again and you will see all the text you have written back in your application.

Devices			Threads	Heap	Allocation Tracker	Network Statistics	File Explorer	Emulator Control	Syst
Name			Name						
Nexus 5 API 21 Online	Nexus 3...		Size						Date
com.android.1473	8600		Time						Permissions
com.android.2179	8601		Info						
android.pr.1349	8602		com.android.shell						2015-03-30 14:39 drwxr-x--x
com.android.1927	8603		com.android.smoketest						2015-03-30 14:39 drwxr-x--x
com.google.1800	8604		com.android.smoketest.tests						2015-03-30 14:39 drwxr-x--x
com.android.1995	8605		com.android.soundrecorder						2015-03-30 14:39 drwxr-x--x
com.google.1741	8606		com.android.speechrecorder						2015-03-30 14:39 drwxr-x--x
android.pr.1555	8607		com.android.systemui						2015-03-30 14:40 drwxr-x--x
com.android.2067	8608		com.android.vending						2015-03-30 14:39 drwxr-x--x
com.svox.2200	8609		com.android.vpndialogs						2015-03-30 14:39 drwxr-x--x
system.pr.1241	8610		com.android.wallpaper.livpicker						2015-03-30 14:39 drwxr-x--x
com.android.1499	8611		com.android.webview						2015-03-30 14:39 drwxr-x--x
com.android.1437	8612		com.android.widgetpreview						2015-03-30 14:39 drwxr-x--x
com.android.1949	8613		com.example.android.apis						2015-03-30 14:39 drwxr-x--x
com.android.1374	8614		com.example.android.livecubes						2015-03-30 14:39 drwxr-x--x
com.android.1889	8615		com.example.android.softkeyboard						2015-03-30 14:39 drwxr-x--x
com.android.2083	8616		com.example.sairamkrishna.myapplication						2015-04-02 15:47 drwxr-x--x
com.android.1316	8617		app_webview						2015-04-02 15:50 drwxrwx--x
com.svox.2343	8618		cache						2015-03-30 14:41 drwxrwx--x
com.android.1577	8619		files						2015-04-02 14:25 drwxrwx--x
com.android.1838	8620		mydata						2015-04-02 14:34 -rw-rw-r--
com.android.1714	8621		lib						2015-03-30 14:40 lrwxrwxrwx -> /data/a...
com.google.1588	8622		shared_prefs						2015-04-07 15:40 drwxrwx--x
			MyPrefs.xml						197 2015-04-07 15:40 -rw-rw-r--
			WebViewChromiumPrefs.xml						124 2015-04-02 15:47 -rw-rw-r--
			com.google.android.apps.maps						2015-03-31 10:12 drwxr-x--x
			com.google.android.gms						2015-03-30 14:42 drwxr-x--x