

Bài 07: ASSET VÀ SHAREPREFERENCES

GVGD: ThS. Đặng Thế Hân

Biên soạn: ThS. Giang Hào Côn

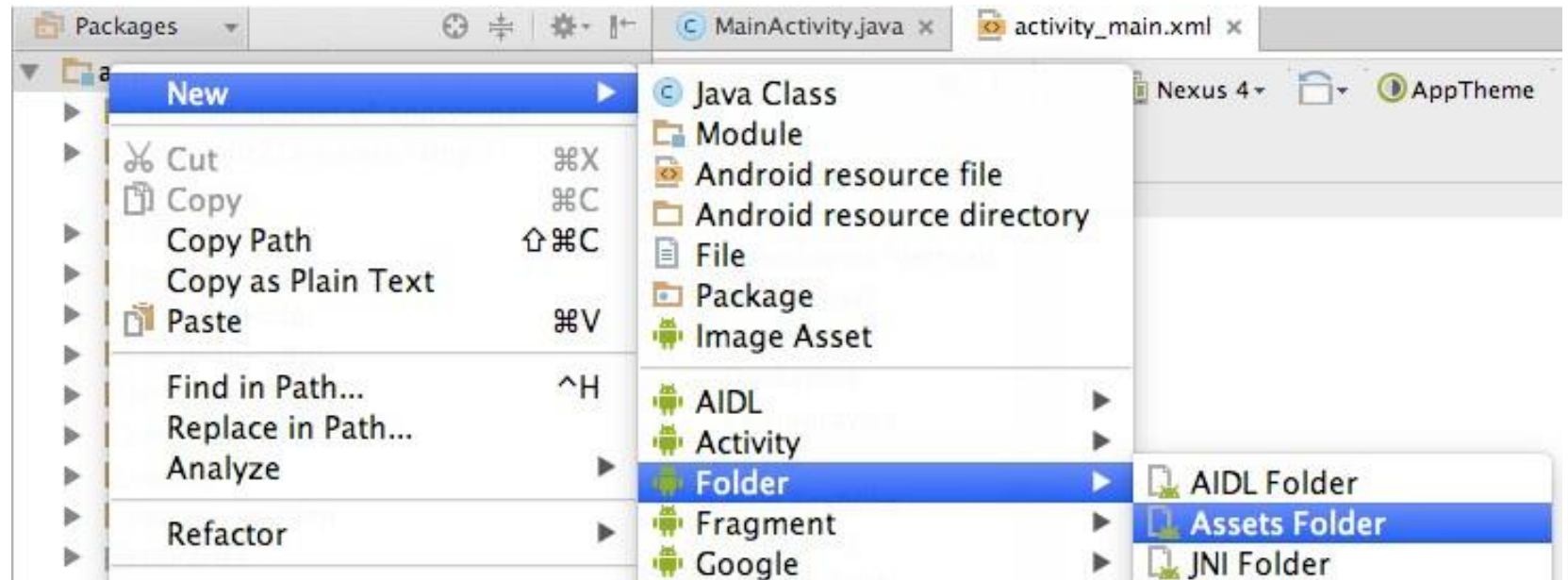
Mục tiêu

Cung cấp cho sinh viên kiến thức cơ bản về Assets Folder trong ứng dụng Android và cách sử dụng SharedPreferences trong ngôn ngữ lập trình ứng dụng trên Android.

- 1) Assets Folder
- 2) SharedPreferences

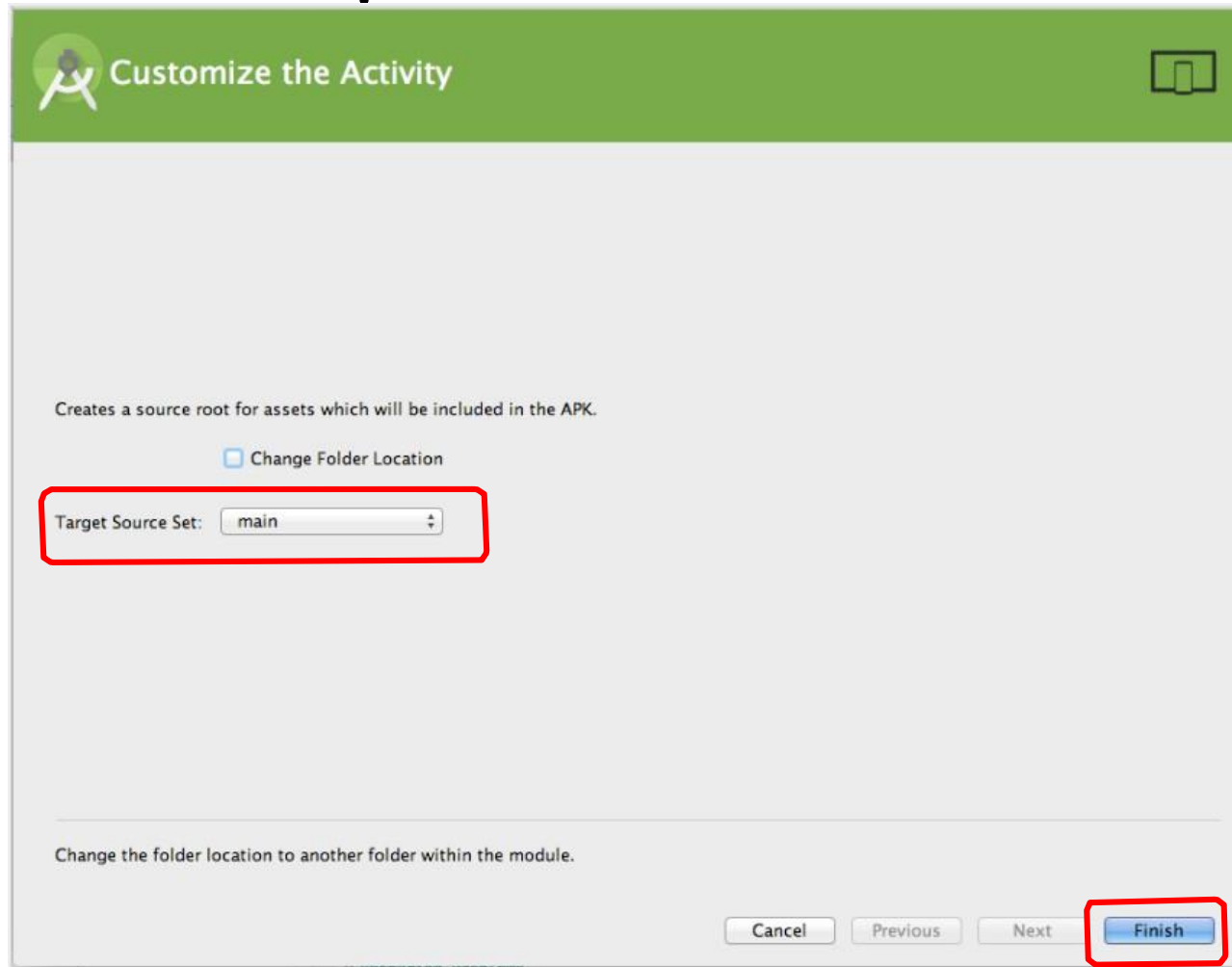
6.1/ Assets Folder

- Là thư mục dùng lưu trữ dữ liệu, các dữ liệu dạng chỉ đọc, không được ghi. Trước khi làm việc cần tạo sẵn dữ liệu chép vào thư mục này để chương trình sử dụng khi cần. **Trình tự tạo thư mục Assets**
 - Right Click vào thư mục gốc của App và chọn: **New >> Folder >> Assets Folder**

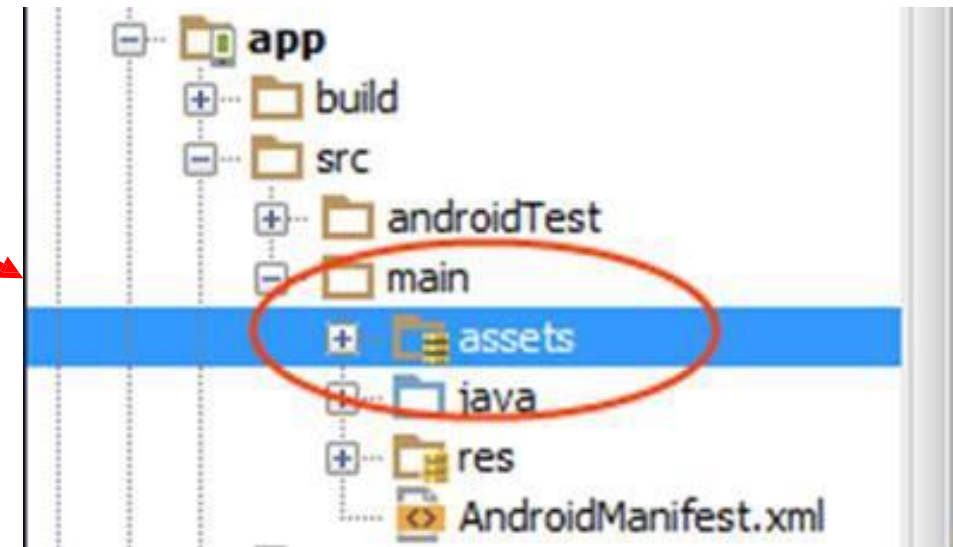


6.1/ Assets Folder

- Xuất hiện cửa sổ :

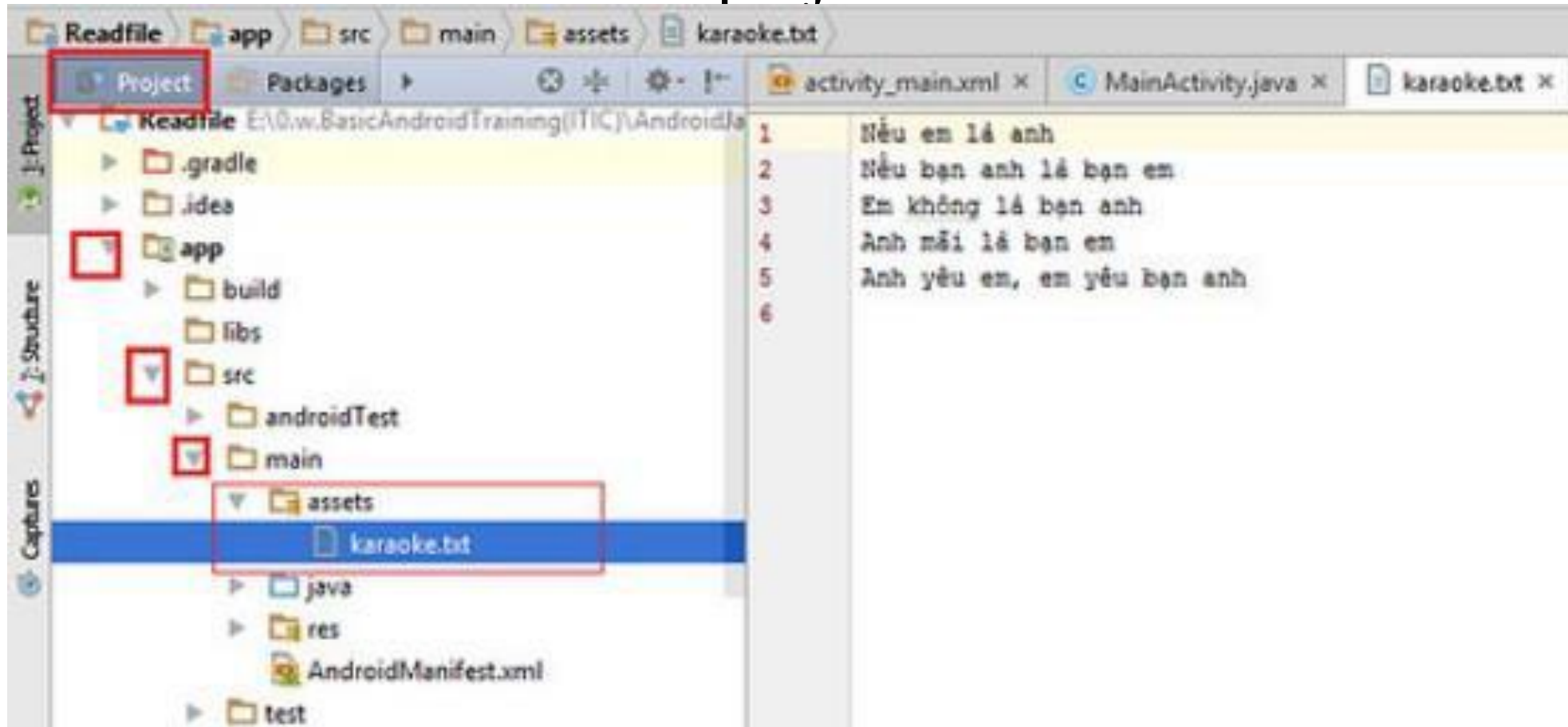


- Tại mục **Target Source Set**: chọn **Main**.
- Bấm nút **Finish**.



6.1/ Assets Folder

Ví dụ 1: Đọc nội dung file text từ thư mục Assets, giả sử ta có file karaoke.txt với cấu trúc project như sau :



6.1/ Assets Folder

Bước 1: trong file “activity_main.xml” tạo 1 textView để hiển thị nội dung tập tin karaoke.txt:

```
<?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:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.wosea.readfile.MainActivity">

    <TextView android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        android:id="@+id/tvText" />

</RelativeLayout>
```

6.1/ Assets Folder

Bước 2: trong file “activity_main.java” viết code như sau:

```
package gianghaocon.wosea.readfile;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.TextView;

import java.io.IOException;
import java.io.InputStream;

public class MainActivity extends AppCompatActivity {

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

        tvMyText=(TextView)findViewById(R.id.tvText);
        Doctaptin();
    }
}
```

```
public void Doctaptin()
{
    try {
        InputStream in= getAssets().open("karaoke.txt");
        byte[] buffer=new byte[80];
        int len=0;
        while((len=in.read(buffer))>0)
        {
            String mystring=new String(buffer);
            tvMyText.append(mystring);
        }
        in.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```


6.1/ Assets Folder

Ví dụ 2: xây dựng App có mẫu sau đây:



Image View

Text View

Bước 1: Tạo Project tên là ViduAssetsFolder

Bước 2: Tạo thư mục Asset.

Bước 3: Chép file **Text.txt** và **avatar.jpg** vào trong thư mục Assets.

Bước 4: Tại file giao diện **.XML** ta chèn 1 **ImageView** để hiển thị file hình **avatar.jpg** và 1 **TextView** để hiển thị nội dung file **Text.txt**

6.1/ Assets Folder

Ví dụ 2: xây dựng App có mẫu sau đây:

Bước 5: Mở tập tin Activity_main.java ta viết code sau.

```
package gianghaocon.android.study;

import java.io.IOException;
import java.io.InputStream;
import android.app.Activity;
import android.graphics.drawable.Drawable;
import android.os.Bundle;
import android.widget.ImageView;
import android.widget.TextView;

public class Main extends Activity {

    ImageView mImage;
    TextView mText;

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

        mImage = (ImageView)findViewById(R.id.image);
        mText = (TextView)findViewById(R.id.text);
        loadDataFromAsset();
    }
}
```

```
public void loadDataFromAsset() {
    try {
        InputStream is = getAssets().open("text.txt");
        int size = is.available();           // check size
        byte[] buffer = new byte[size];      // create buffer for IO
        is.read(buffer);                     // get data to buffer
        is.close();                          // close stream
        mText.setText(new String(buffer));    // set result to TextView
    }

    try {
        InputStream ims = getAssets().open("avatar.jpg");
        Drawable d = Drawable.createFromStream(ims, null);
        mImage.setImageDrawable(d);
    }
    catch(IOException ex) { return; }
}
```

6.1/ Assets Folder

Ví dụ 3: xây dựng App đọc tập tin .HTML:

Giả sử ta có tập tin myfile.html với nội dung như sau:

```
<h1>Heading 1</h1>
<h2>Heading 2</h2>
<p>This is some html. Look, here's an <u>underline</u>.</p>
<p>Look, this is <em>emphasized.</em> And here\\'s some <b>bold</b>.</p>
<p>Here are UL list items:
<ul>
  <li>One</li>
  <li>Two</li>
  <li>Three</li>
</ul>
<p>Here are OL list items:
<ol>
  <li>One</li>
  <li>Two</li>
  <li>Three</li>
</ol>
```

6.1/ Assets Folder

Ví dụ 3: xây dựng App đọc tập tin .HTML:

Bước 1: Tạo project mới tên là **HtmlExample**

Bước 2: Tạo thư mục **Assets** và chép tập tin **myfile.html** vào thư mục Assets vừa tạo.

Bước 3: mở file **activity_main.xml** và viết thêm code như sau:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="abhiandroid.com.htmlexample.MainActivity">

    <WebView
        android:id="@+id/simpleWebView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="20dp" />

</RelativeLayout>
```

6.1/ Assets Folder

Ví dụ 3: xây dựng App đọc tập tin .HTML:

Bước 4: mở file **MainActivity.java** và viết code như sau:

```
package gianghaocon.com.htmlexample;

import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.webkit.WebView;

public class MainActivity extends AppCompatActivity {

    WebView webView;

    public String fileName = "myfile.html";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        webView = (WebView) findViewById(R.id.simpleWebView);
        webView.getSettings().setJavaScriptEnabled(true);
        webView.loadUrl("file:///android_asset/" + fileName);
    }
}
```

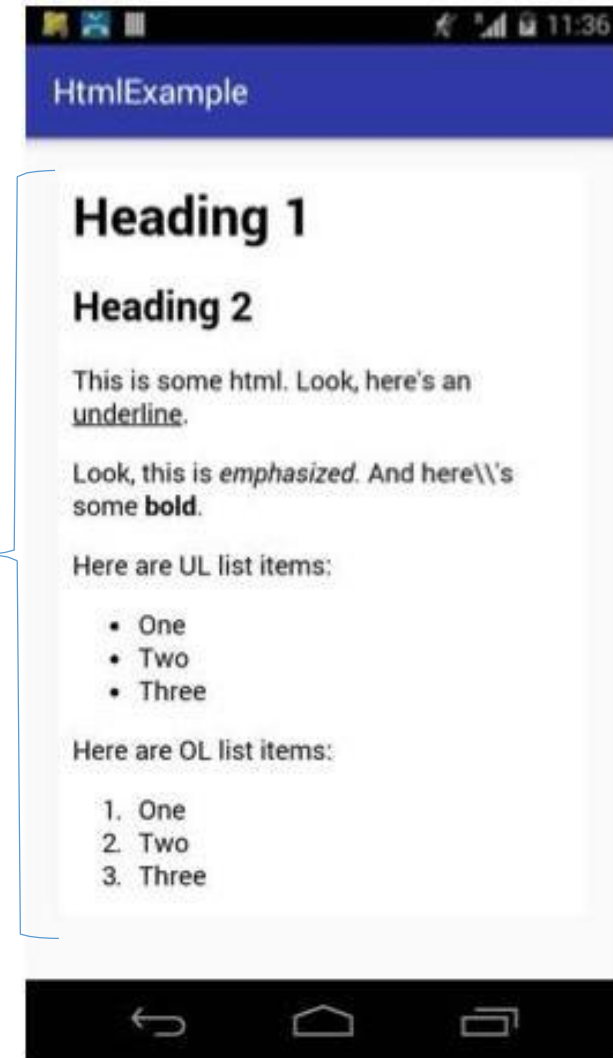
6.1/ Assets Folder

Ví dụ 3: xây dựng App đọc tập tin .HTML:

Bước 5: Chạy App sẽ hiện ra kết quả sau:

Web View

```
web =(WebView)findViewById(R.id.webview);  
WebView web = new WebView();  
web.loadUrl("file:///android_asset/sample.html");
```



6.2/ SharedPreferences trong Android

- **SharedPreferences** trong Android là một **class Interface** cho phép đọc và lưu dữ liệu bằng các cặp **key và value** dưới dạng file **xml** trong ứng dụng Android. Ngoài ra, dữ liệu của nó cũng có thể lưu là ở dưới dạng nguyên thủy như: **int, float, string, boolean, long**.
- ▪ Vì ổ chứa gốc của SharedPreferences cũng chính là ứng dụng Android, nên khi ta xóa ứng dụng hoặc dữ liệu thì file này cũng hoàn toàn biến mất. Để lưu được, ta phải chuyển qua kiểu String rồi save bằng **thư viện Gson**.
- Vậy, khi nào thì nên sử dụng SharedPreferences trong Android?

6.2/ SharedPreferences trong Android

- Thực tế, khi lập trình một ứng dụng nào đó, ta sẽ gặp phải một số những dữ liệu nhỏ mà ta phải lưu lại xuyên suốt vòng đời của ứng dụng. Nhưng nó quá ít để lưu lại trong database, lưu ở bộ nhớ trong hay ngoài của ứng dụng ư? Cũng không được, vì sẽ gây lãng phí và rất dễ bị mất dữ liệu. **Lúc này, SharedPreferences trong Android sẽ giúp bạn khắc phục điều đó!.**
- Mặc dù, SharedPreferences rất dễ sử dụng, nhưng code của nó sẽ trong ứng dụng Android lại rất **rắc rối và rườm rà.**

6.2/ SharedPreferences trong Android

Lưu trữ dữ liệu xuống SharedPreferences.

- **Bước 1:** Đầu tiên, để lưu dữ liệu xuống thì ta sẽ khởi tạo một đối tượng (biến) thuộc kiểu Shared Preferences, với cú pháp như sau:

```
SharedPreferences shp = getSharedPreferences(SHARED_PREFERENCES_NAME, Context.MODE_PRIVATE);
```

Tên biến do người dung tự đặt

là tên của file Shared Preferences mà ta sẽ lưu xuống thiết bị

data/data/[application package name]/shared_prefs/shared_preferences_name.xml

đây là một chế độ bảo mật dữ liệu trong android, chỉ cho ứng dụng hiện tại truy cập vào file Shared Preferences này thôi mà không một ứng dụng nào có quyền truy cập vào được.

6.2/ SharedPreferences trong Android

Lưu trữ dữ liệu xuống SharedPreferences.

- **Bước 2:** Tạo đối tượng Editor từ biến **shp** chúng ta đã tạo ở trên, mục đích của đối tượng Editor này là để chúng ta có thể mở file **shared_preferences_name.xml** ra và đưa dữ liệu vào, cú pháp như sau:

```
shp.Editor editor = shp.edit();
```

Để đưa dữ liệu vào chúng ta sẽ làm viết như sau:

```
editor.put?(String key, value)
```

Giá trị sẽ lưu
vào biến

float, string, int, boolean, long.

là tên biến ta sẽ lưu xuống, ví dụ ta muốn lưu họ tên nào đó xuống thì bạn sẽ đặt là “hoten”.

6.2/ SharedPreferences trong Android

Lưu trữ dữ liệu xuống SharedPreferences.

- Ví dụ chi tiết như sau:

```
SharedPreferences shp = getSharedPreferences(SHARED_PREFERENCES_NAME, Context.MODE_PRIVATE);  
shp.Editor editor = shp.edit();  
editor.putString("Hoten", "Giang Hào Côn");  
editor.putBoolean("GioiTinh", true);  
editor.putInt("MaMon", 100);  
editor.putFloat("Diem", 9.5f);
```

Sau khi đã put dữ liệu xong thì ta sẽ gọi hàm **commit()** hoặc là **apply()** để lưu xuống ổ đĩa.

6.2/ SharedPreferences trong Android

Lưu trữ dữ liệu xuống SharedPreferences.

- Sự khác nhau giữa **commit()** và **apply()**:

Commit()	Apply()
hoạt động theo cơ chế đồng bộ Synchronous , nếu như ta khởi tạo 2 editor để chỉnh sửa dữ liệu thì editor nào thực hiện trước sẽ làm trước và editor đến sau sẽ làm sau. Khi thi hành xong thì trả về là true hay false nếu như thành công hoặc thất bại .	hoạt động theo cơ chế không đồng bộ Asynchronous , và dù có thành công hay không thì hàm apply() sẽ không trả về giá trị nào cả.

6.2/ SharedPreferences trong Android

Lấy dữ liệu từ SharedPreferences.

```
<kiểu> <tên biến> = shp.get?("preferences_name", default_value)
```

float, string, int,
boolean, long.

Giá trị mặc định này sẽ được dùng nếu
preferences_name chưa được lưu.

```
SharedPreferences shp = getSharedPreferences(SHARED_PREFERENCES_NAME, Context.MODE_PRIVATE);  
boolean gioitinh = shp.getBoolean("gioitinh", true);  
int mamon = shp.getInt("MaMon", 0);  
float diem = shp.getFloat("diem", 0.0f);  
String hoten = shp.getString("hoten", "");
```

6.2/ SharedPreferences trong Android

Cách dữ liệu đã lưu xuống SharedPreferences.

```
editor.remove(String key);
```

Và trước hợp ta muốn xoá đi hết tất cả giá trị đã lưu thì ta dùng cú pháp:

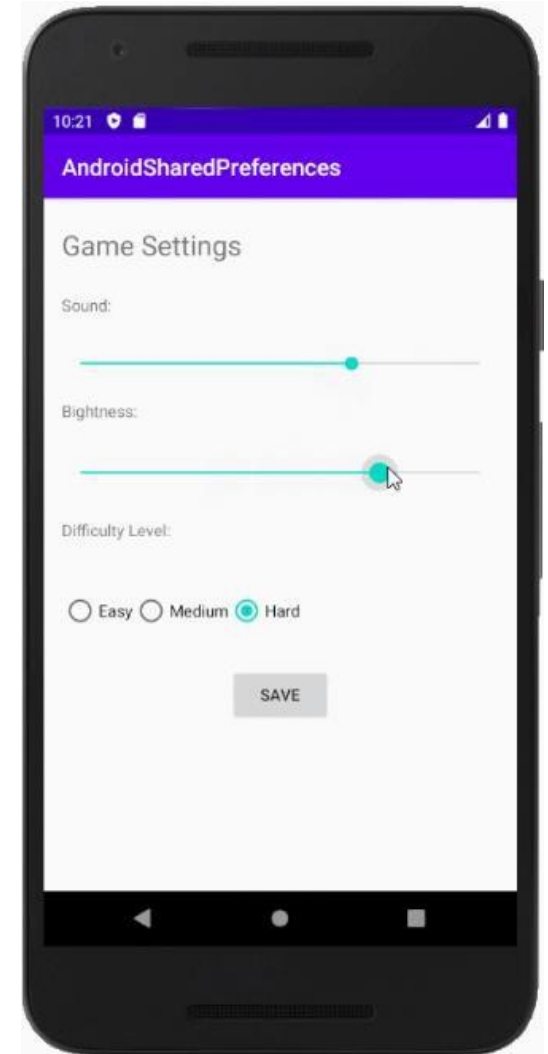
```
editor.clear();
```

Và cũng lại gọi **editor.apply()** hoặc **editor.commit()** để lưu thay đổi.

6.2/ SharedPreferences trong Android

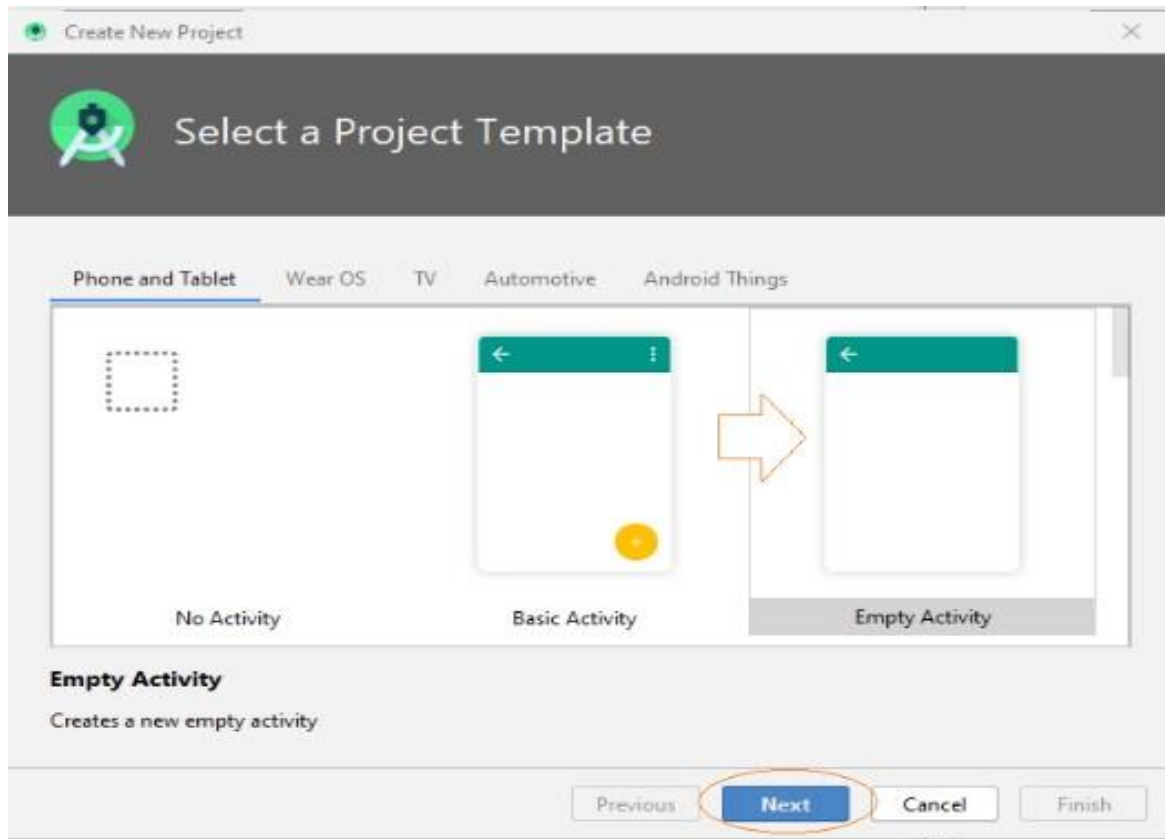
Ví dụ sử dụng SharedPreferences

Giả sử ta đang chơi một trò chơi trên Android, trước khi chơi trò chơi đó ta lựa chọn các thông số của trò chơi chẳng hạn **độ sáng** trong trò chơi, **mức độ âm lượng**, và **độ khó**. Sau khi chơi xong ta tắt trò chơi và có thể tiếp tục chơi vào ngày hôm sau. **SharedPreferences** cho phép bạn lưu lại các các thông số ta đã thiết lập trước đó, để cho phép khi ta chơi lại các thiết lập đó có thể sử dụng mà không cần phải thiết lập lại.



6.2/ SharedPreferences trong Android

▪ **Bước 1:** Tạo mới Project có tên AndroidSharedPreferences

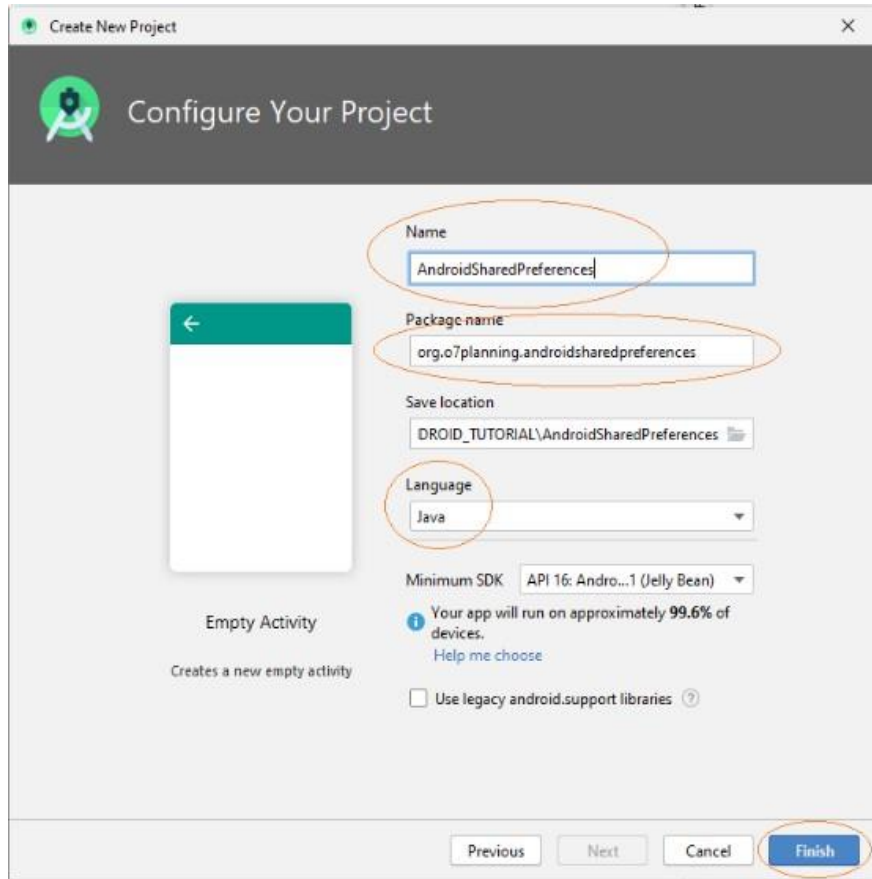


>> **File** >> **New** >> **New Project**

Chọn **Empty Activity** Và bấm nút **Next**

6.2/ SharedPreferences trong Android

▪ **Bước 1:** Tạo mới Project có tên AndroidSharedPreferences



▪ Name:

▪ AndroidSharedPreferences

▪ Package name:

org.gianghaocon.androidsharepreference

Language

Java

Bấm nút Next

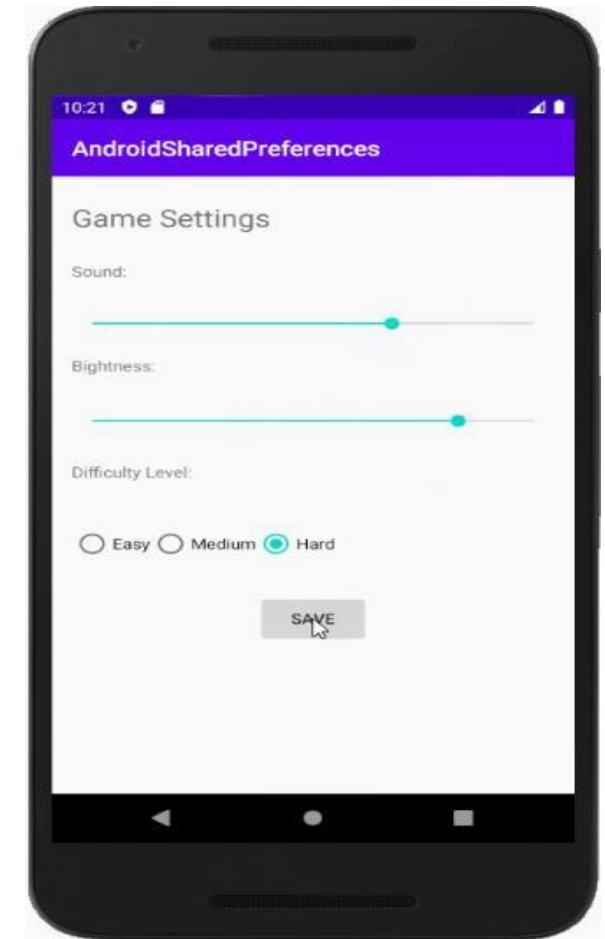
6.2/ SharedPreferences trong Android

Bước 2: Mở file [activity_main.xml](#) và chèn các TextView, Radio Button, SeekBar, Button như mẫu.

Bước 3: Mở file [MainActivity.java](#) và viết code.

Bước 4: Chạy ứng dụng xuất hiện giao diện sau

Test thử bằng cách chạy ứng dụng và chỉnh các thông số rồi bấm nút Save, thoát khỏi ứng dụng và chạy lại.



Câu hỏi thảo luận

- 1) Trình bày cách sử dụng Assets Folder và cho ví dụ cụ thể về cách sử dụng Assets Folder
- 2) Trình bày SharedPreferences và cho ví dụ cụ thể về cách sử dụng SharedPreferences.