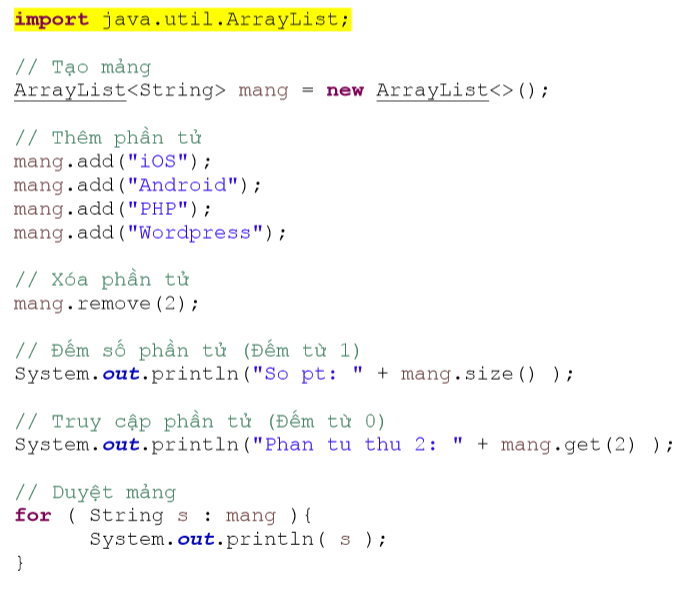
|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **CÀI ĐẶT PHẦN MỀM ĐỂ LẬP TRÌNH ANDROID** |

Bước 1: Cài JDK (là bộ Java) <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

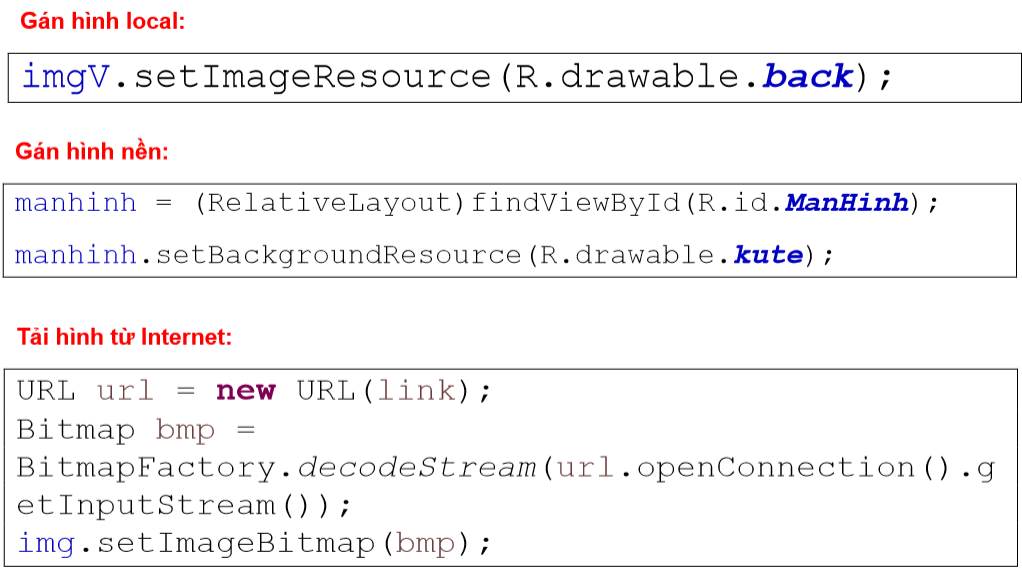
Bước 2: Cài máy ảo GenyMotion: <https://www.genymotion.com/>

Bước 3: Cài Android Studio: <https://developer.android.com/sdk/index.html>

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **MẢNG - ARRAYLIST** |

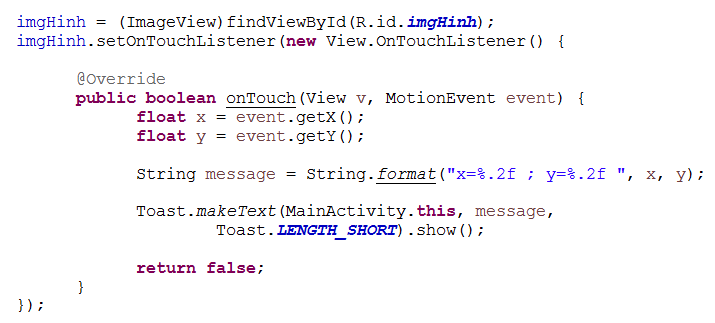


|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **IMAGEVIEW** |

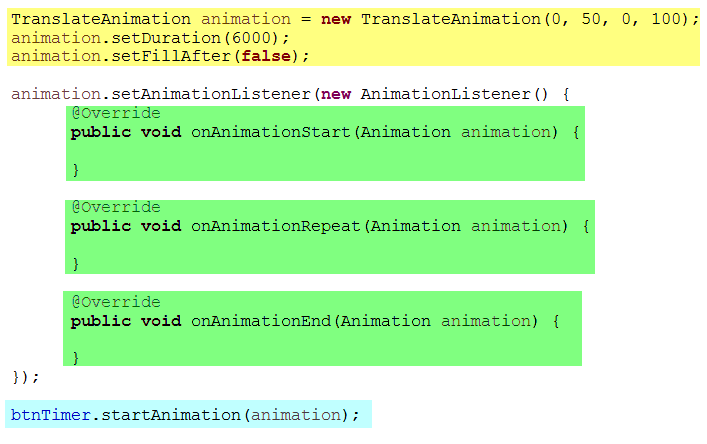


Lưu ý: Hình load từ internet từ Android 4. trở lên bắt buộc phải dùng ASYNC (Bài Async ở dưới)

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **BẮT TỌA ĐỘ KHI CHẠM TAY VÀO MÀN HÌNH** |



|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **ANIMATION** |



|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **TIMER** |

new CountDownTimer(5000, 100) {

public void onTick(long millisUntilFinished) {

txtvSo.setText("còn lại: " + millisUntilFinished / 100);

}

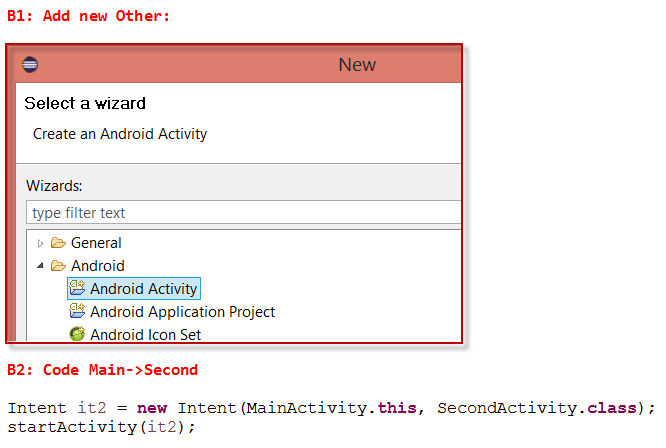
public void onFinish() {

txtvSo.setText("done!");

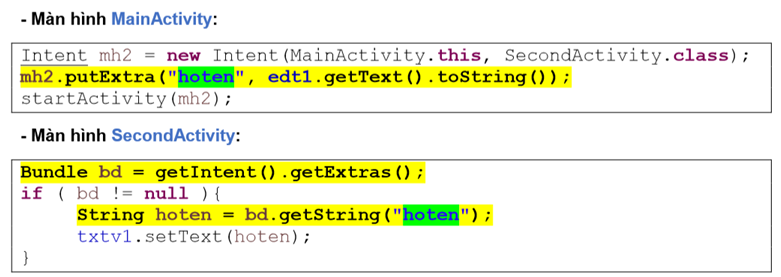
}

}.start();

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **DI CHUYỂN GIỮA CÁC MÀN HÌNH** |



|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **TRUYỀN THAM SỐ GIỮA CÁC MÀN HÌNH** |



|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **ALERT DIALOG** |

Button btnNut;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

btnNut = (Button)findViewById(R.id.btnNut);

btnNut.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

showAlert();

}

});

}

public void showAlert(){

AlertDialog.Builder builder = new Builder(this);

builder.setPositiveButton("ON", new OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

// TODO Auto-generated method stub

}

});

builder.setTitle("Tieu de");

builder.setMessage("Noi dung ne");

AlertDialog ad = builder.create();

ad.show();

}

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **LIST VIEW CƠ BẢN** |

B1: Kéo listview vào

<ListView   
android:id=*"@+id/listviewMonHoc"*  
android:layout\_width=*"match\_parent"*  
android:layout\_height=*"wrap\_content"*  
android:layout\_centerHorizontal=*"true"*  
android:layout\_centerVertical=*"true"* >  
</ListView>

B2: Viết code code:

ListView monhocLV = (ListView)findViewById(R.id.listviewMonHoc);

String[] monhoc = new String[]{

"iOS cơ bản",

"iOS nâng cao",

"PHP & MySQL",

"Joomla",

"Wordpress"

};

ArrayAdapter adapter = new ArrayAdapter(this, android.R.layout.simple\_list\_item\_1, monhoc);

monhocLV.setAdapter(adapter);

\*\*\*\* Bắt sự kiện chọn 1 item trong listview

monhocLV.setOnItemClickListener(new OnItemClickListener() {

@Override

public void onItemClick(AdapterView parent, View view, int position, long id) {

// TODO Auto-generated method stub

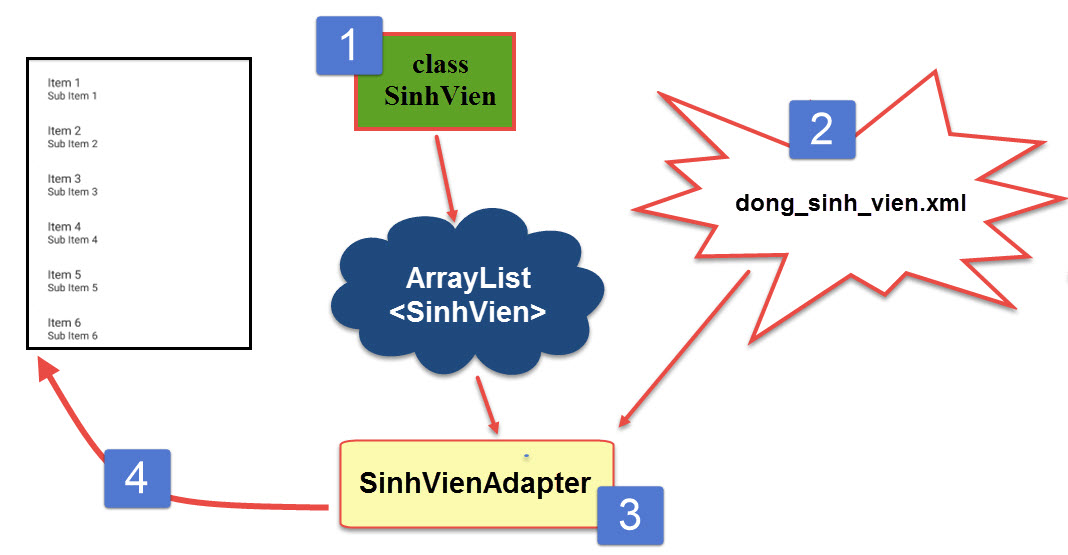
Toast.makeText(getApplicationContext(), monhoc[position], Toast.LENGTH\_SHORT).show();

}

});

"Mảng monhoc phải thêm **final**"

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **LIST VIEW NÂNG CAO - CUSTOM ADAPTER** |



<http://khoapham.vn/KhoaPhamTraining/android/Custom.txt>

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **TOAST** |

- Toast cơ bản:  
Toast toast = Toast.*makeText*(MainActivity.**this**, "Hello World", Toast.***LENGTH\_SHORT***);  
toast.show();

- Chèn View vào bên trong Toast:  
LayoutInflater inflater = getLayoutInflater();  
View layoutQC = inflater.inflate(R.layout.thongbao, (ViewGroup)findViewById(R.id.layoutThongbao));  
Toast t = new Toast(MainActivity.this);  
t.setView(layoutQC);  
t.show();

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **ÉP KIỂU DỮ LIỆU** |

Ép số: Integer.*parseInt*("1987")

Ép chuỗi: String.*valueOf*(10)

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **PLAY MP3 FROM INTERNET + LOCAL** |

- From local:  
MediaPlayer song = MediaPlayer.*create*(MainActivity.**this**, R.raw.***nuocmat***);   
song.start();  
 **Stop:**   
onPause();   
song.pause();

- From Internet:

public void PlayNhacMp3(String url){

//url = "http://khoapham.vn/download/vietnamoi.mp3";

MediaPlayer mediaPlayer = new MediaPlayer();

mediaPlayer.setAudioStreamType(AudioManager.STREAM\_MUSIC);

try {

mediaPlayer.setDataSource(url);

mediaPlayer.prepareAsync();

mediaPlayer.setOnPreparedListener(new MediaPlayer.OnPreparedListener() {

@Override

public void onPrepared(MediaPlayer mp) {

mp.start();

}

});

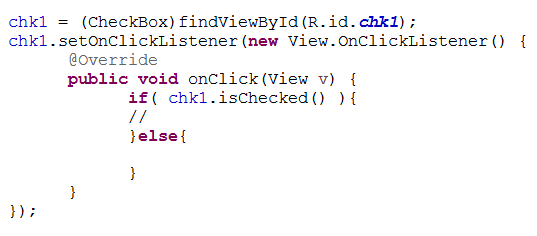
} catch (IOException e) {

e.printStackTrace();

}

}

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **CHECKBOX** |



|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **SAVE + READ FILE TXT** |

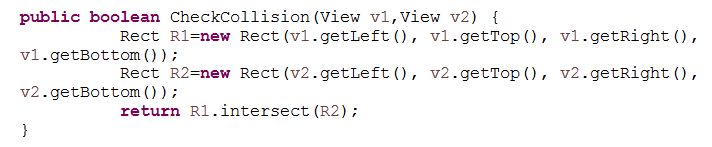
- Save file txt

FileOutputStream fos = openFileOutput("**khoapham.txt**", Context.***MODE\_PRIVATE***);   
fos.write(noidung.getBytes());  
fos.close()

- Read file txt

FileInputStream fis = openFileInput("**khoapham.txt**");   
BufferedReader br = **new** BufferedReader(**new** InputStreamReader(**new** DataInputStream(fis)));   
String line = "";   
**while**( (line = br.readLine()) != **null** ){   
txtvNoiDung.append(line);   
txtvNoiDung.append("\n");   
}

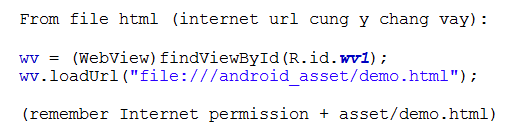
|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **CHECK 2 VIEW CHẠM NHAU** |



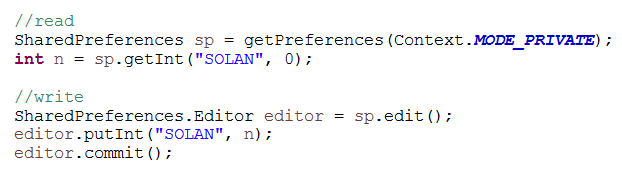
|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **KÍCH THƯỚC CỦA MÀN HÌNH THIẾT BỊ** |

Display display = ((WindowManager) getSystemService(***WINDOW\_SERVICE***)).getDefaultDisplay();   
**int** width = display.getWidth();   
**int** height = display.getHeight();   
**int** ori = display.getOrientation();

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **WEBVIEW: URL INTERNET & LOCAL HTML** |

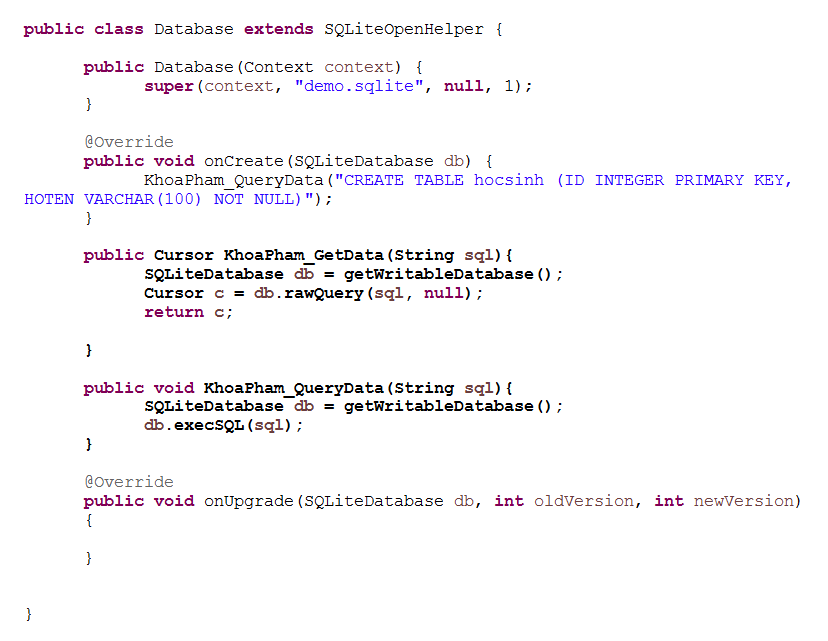


|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **SHARED PREFERENCE** |

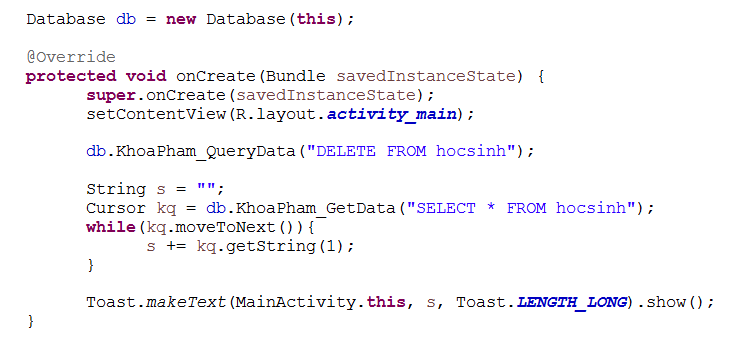


|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **SQLITE** |

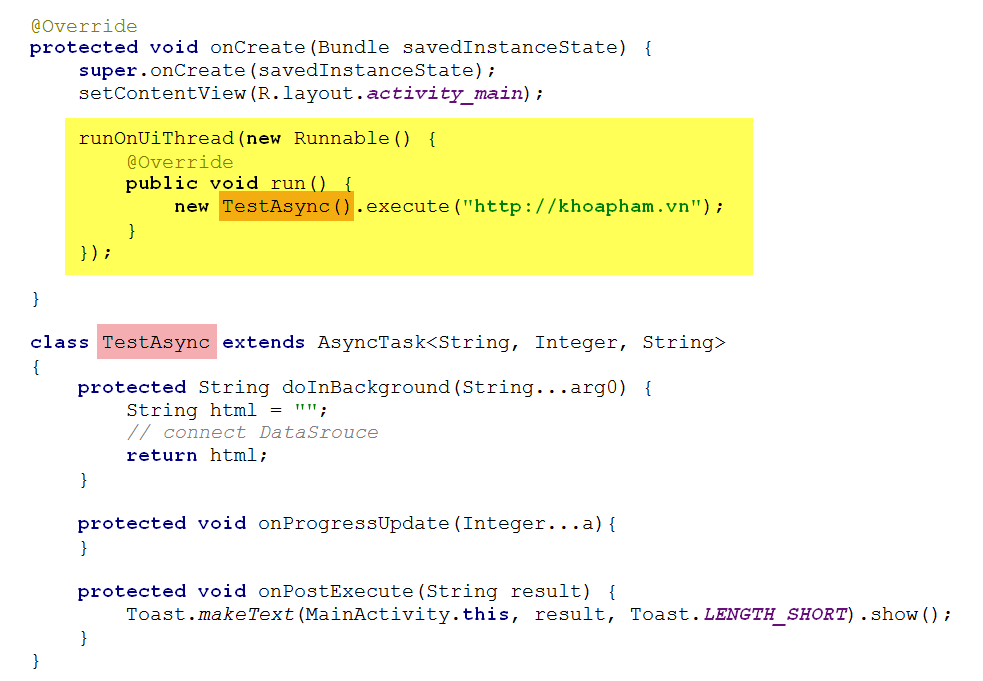
- Tạo class Database



- Sử dụng class Database



|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **ASYNC** |

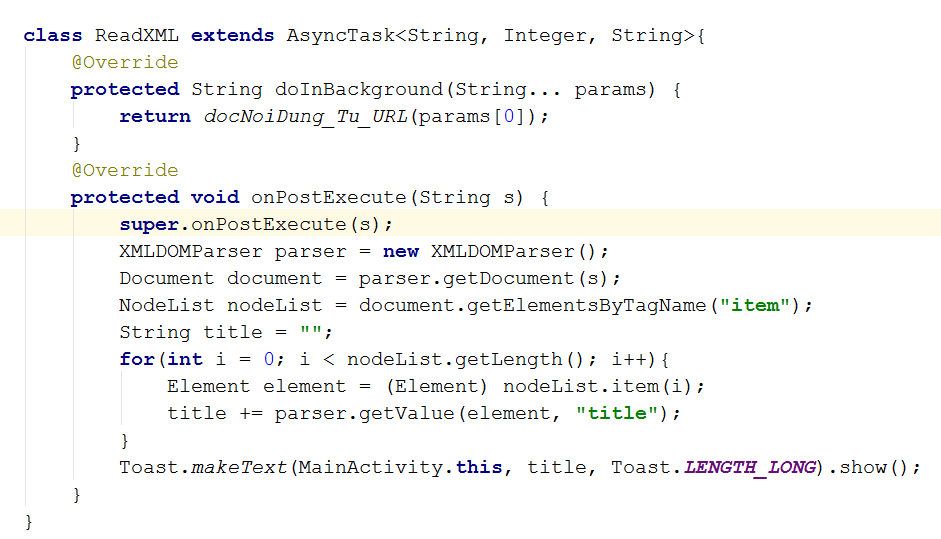
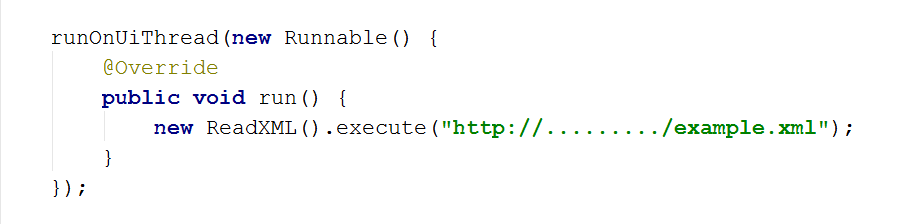


|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **XMLDOMPARSER** |

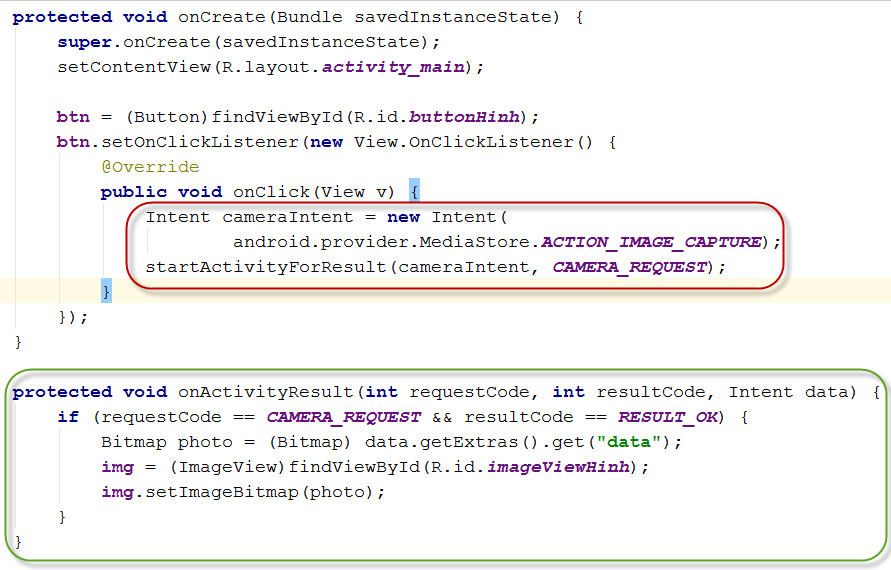
**B1: Tạo class XMLDOMParser**

1. <http://khoapham.vn/KhoaPhamTraining/android/XMLDOMParser.rtf>

**B2: MainActivity**

1. **Tạo hàm docNoiDung\_Tu\_URL(String theUrl) - [ THAY THẾ CHO getXmlFromUrl ]**  
   Lưu ý: Hàm này tìm ở phía dưới.
2. **Tạo Async:**  
   
3. **OnCreate**  
   

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **KÍCH HOẠT CAMERA CHỤP HÌNH** |



Code:

Intent cameraIntent = **new**Intent(android.provider.MediaStore.***ACTION\_IMAGE\_CAPTURE***);  
startActivityForResult(cameraIntent, ***CAMERA\_REQUEST***);

protected void onActivityResult(int requestCode, int resultCode, Intent data) {

if (requestCode == CAMERA\_REQUEST && resultCode == RESULT\_OK) {

Bitmap photo = (Bitmap) data.getExtras().get("data");

img = (ImageView)findViewById(R.id.imageViewHinh);

img.setImageBitmap(photo);

}

}

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **TEXT TO SPEED** |

TextToSpeech t1;  
@Override  
protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 t1=new TextToSpeech(getApplicationContext(), new TextToSpeech.OnInitListener() {  
 @Override  
 public void onInit(int status) {  
 if(status != TextToSpeech.ERROR) {  
 t1.setLanguage(Locale.ENGLISH);  
 t1.speak("I've been lonely with you", TextToSpeech.QUEUE\_FLUSH, null)  
 }  
 }  
 });  
}

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **SEND POST ĐẾN WEB SERVER** |

Note: Phải sử dụng Asysnc

// link: truyền link nhận POST

private String makePostRequest(String link){

HttpURLConnection connect;

URL url = null;

try {

url = new URL(link);

} catch (MalformedURLException e) {

e.printStackTrace();

return "Error!";

}

try {

// cấu hình HttpURLConnection

connect = (HttpURLConnection)url.openConnection();

connect.setReadTimeout(10000);

connect.setConnectTimeout(15000);

connect.setRequestMethod("POST");

// Gán tham số vào URL

Uri.Builder builder = new Uri.Builder()

.appendQueryParameter("thamSo1", "KhoaPhamTraining")

.appendQueryParameter("thamSo2", "90 Lê Thị Riêng");

String query = builder.build().getEncodedQuery();

// Mở kết nối gửi dữ liệu

OutputStream os = connect.getOutputStream();

BufferedWriter writer = new BufferedWriter(

new OutputStreamWriter(os, "UTF-8"));

writer.write(query);

writer.flush();

writer.close();

os.close();

connect.connect();

} catch (IOException e1) {

e1.printStackTrace();

return "Error!";

}

try {

int response\_code = connect.getResponseCode();

// kiểm tra kết nối ok

if (response\_code == HttpURLConnection.HTTP\_OK) {

// Đọc nội dung trả về

InputStream input = connect.getInputStream();

BufferedReader reader = new BufferedReader(new InputStreamReader(input));

StringBuilder result = new StringBuilder();

String line;

while ((line = reader.readLine()) != null) {

result.append(line);

}

return result.toString();

}else{

return "Error!";

}

} catch (IOException e) {

e.printStackTrace();

return "Error!";

} finally {

connect.disconnect();

}

}

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **ĐỌC NỘI DUNG CỦA MỘT URL INTERNET** |

private String docNoiDung\_Tu\_URL(String theUrl){

StringBuilder content = new StringBuilder();

try {

// create a url object

URL url = new URL(theUrl);

// create a urlconnection object

URLConnection urlConnection = url.openConnection();

// wrap the urlconnection in a bufferedreader

BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(urlConnection.getInputStream()));

String line;

// read from the urlconnection via the bufferedreader

while ((line = bufferedReader.readLine()) != null){

content.append(line + "\n");

}

bufferedReader.close();

}

catch(Exception e) {

e.printStackTrace();

}

return content.toString();

}

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **IMAGEVIEW TO BYTE ARRAY** |

public byte[] ImageView\_To\_Byte(ImageView imgv){

BitmapDrawable drawable = (BitmapDrawable) imgv.getDrawable();

Bitmap bmp = drawable.getBitmap();

ByteArrayOutputStream stream = new ByteArrayOutputStream();

bmp.compress(Bitmap.CompressFormat.PNG, 100, stream);

byte[] byteArray = stream.toByteArray();

return byteArray;

}

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **BYTE ARRAY TO IMAGEVIEW** |

Bitmap bitmap = BitmapFactory.decodeByteArray(BYTE[], 0, BYTE[].length);

imgv.setImageBitmap(bitmap);

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **CONFIG PARSE CONNECT** |

- Add class MyApplication

public class MyApplication extends Application {

@Override

public void onCreate(){

super.onCreate();

Parse.initialize(this, "xxx", "yyy");

}

}

- Trong Manifest

android:name="MyApplication"

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **PLAY MP3 FROM BYTE[]** |

private void playMp3FromByte(byte[] mp3SoundByteArray) {

try {

File tempMp3 = File.createTempFile("kurchina", "mp3", getCacheDir());

tempMp3.deleteOnExit();

FileOutputStream fos = new FileOutputStream(tempMp3);

fos.write(mp3SoundByteArray);

fos.close();

MediaPlayer mediaPlayer = new MediaPlayer();

FileInputStream fis = new FileInputStream(tempMp3);

mediaPlayer.setDataSource(fis.getFD());

mediaPlayer.prepare();

mediaPlayer.start();

} catch (IOException ex) {

String s = ex.toString();

ex.printStackTrace();

}

}

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **CONVERT FILE LOCAL TO BYTE[]** |

public byte[] FileLocal\_To\_Byte(String path){

File file = new File(path);

int size = (int) file.length();

byte[] bytes = new byte[size];

try {

BufferedInputStream buf = new BufferedInputStream(new FileInputStream(file));

buf.read(bytes, 0, bytes.length);

buf.close();

} catch (FileNotFoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

return bytes;

}

// path

outputFile = Environment.getExternalStorageDirectory().

getAbsolutePath() + "/filename.3gpp";

filename.3gpp nằm trong res

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **PLAY FILE MP4 LOCAL VÀ INTERNET** |

**Mp4 Local:**

mVideoView.setVideoURI(Uri.parse("android.resource://" + getPackageName() + "/" + R.raw.test));

mVideoView.start();

// media controler

MediaController mediaController = new MediaController(this);

mediaController.setMediaPlayer(mVideoView);

mVideoView.setMediaController(mediaController);

**MP4 INTERNET:**

Uri uri = Uri.parse("http://khoapham.vn/download/vuoncaovietnam.mp4");

mVideoView.setMediaController(new MediaController(MainActivity.this));

mVideoView.setVideoURI(uri);

mVideoView.start();

Lưu ý:

1 - Nhớ xin Permission Internet

2 - mVideoView: là đối tượng VideoView

3 - Dừng phim lại: mVideoView.stopPlayback();

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **VIẾT ỨNG DỤNG GHI ÂM** |

**Bước 1: Lấy Permissions**

< uses-permission android:name="android.permission.RECORD\_AUDIO" />

< uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE"/>

**Bước 2: MainActivity:**

**1. Biến toàn cục:**

private MediaRecorder myRecorder;

private MediaPlayer myPlayer;

private String outputFile = null;

**2. Hàm OnCreate:**

outputFile = Environment.getExternalStorageDirectory().getAbsolutePath() + "/khoaphamvn.3gpp";

myRecorder = new MediaRecorder();

myRecorder.setAudioSource(MediaRecorder.AudioSource.MIC);

myRecorder.setOutputFormat(MediaRecorder.OutputFormat.THREE\_GPP);

myRecorder.setAudioEncoder(MediaRecorder.OutputFormat.AMR\_NB);

myRecorder.setOutputFile(outputFile);

**Bước 3: Các hàm xử lý:**

**1. Bắt đầu ghi âm**

public void start(View view){

try {

myRecorder.prepare();

myRecorder.start();

} catch (IllegalStateException e) {

e.printStackTrace();

} catch (IOException e) {

e.printStackTrace();

}

Toast.makeText(getApplicationContext(), "Start recording...",

Toast.LENGTH\_SHORT).show();

}

**2. Dừng ghi âm**

public void stop(View view){

try {

myRecorder.stop();

myRecorder.release();

myRecorder = null;

Toast.makeText(getApplicationContext(), "Stop recording...",

Toast.LENGTH\_SHORT).show();

} catch (IllegalStateException e) {

e.printStackTrace();

} catch (RuntimeException e) {

e.printStackTrace();

}

}

**3. Play âm thanh đã thu**

public void play(View view) {

try{

myPlayer = new MediaPlayer();

myPlayer.setDataSource(outputFile);

myPlayer.prepare();

myPlayer.start();

Toast.makeText(getApplicationContext(), "Start play the recording...",Toast.LENGTH\_SHORT).show();

} catch (Exception e) {

e.printStackTrace();

}

}

**4. Dừng Play:**

public void stopPlay(View view) {

try {

if (myPlayer != null) {

myPlayer.stop();

myPlayer.release();

myPlayer = null;

Toast.makeText(getApplicationContext(), "Stop playing the recording...", Toast.LENGTH\_SHORT).show();

}

} catch (Exception e) {

e.printStackTrace();

}

}

|  |  |
| --- | --- |
| http://khoapham.vn/KhoaPhamTraining/android/snipet/android.png | **SPINNER** |

**1. Cầu hình Spinner dạng cơ bản:**

spinner1 = (Spinner)findViewById(R.id.spinnerChon);

List list = new ArrayList();

list.add("list 1");

list.add("list 2");

list.add("list 3");

ArrayAdapter dataAdapter = new ArrayAdapter(this,

android.R.layout.simple\_spinner\_item, list);

dataAdapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

spinner1.setAdapter(dataAdapter);

spinner1.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {

@Override

public void onItemSelected(AdapterView parent, View view, int position, long id) {

Toast.makeText(getApplicationContext(), "" + position, Toast.LENGTH\_SHORT).show();

}

@Override

public void onNothingSelected(AdapterView parent) {

}

});

**2. Bắt giá trị trong Spinner:**

String.valueOf(spinner1.getSelectedItem()); // TPHCM

String.valueOf(spinner1.getSelectedItemPosition()); // 0

**CODE THƯỜNG DÙNG**

* 1. **XML DOMPARSER**

**import** android.util.Log;  
**import** org.w3c.dom.Document;  
**import** org.w3c.dom.Element;  
**import** org.w3c.dom.Node;  
**import** org.w3c.dom.NodeList;  
**import** org.xml.sax.InputSource;  
**import** org.xml.sax.SAXException;  
**import** java.io.IOException;  
**import** java.io.StringReader;  
**import** javax.xml.parsers.DocumentBuilder;  
**import** javax.xml.parsers.DocumentBuilderFactory;  
**import** javax.xml.parsers.ParserConfigurationException;  
**public class** XMLDOMParser {  
 **public** Document getDocument(String xml)  
 {   
 Document document = **null**;  
 DocumentBuilderFactory factory = DocumentBuilderFactory.*newInstance*();  
 **try**{  
 DocumentBuilder db = factory.newDocumentBuilder();  
 InputSource is = **new** InputSource();  
 is.setCharacterStream(**new** StringReader(xml));  
 is.setEncoding(**"UTF-8"**);  
 document = db.parse(is);  
 }**catch**(ParserConfigurationException e)  
 {  
 Log.*e*(**"Error: "**, e.getMessage(), e);  
 **return null**;  
 }  
 **catch** (SAXException e) {  
 Log.*e*(**"Error: "**, e.getMessage(), e);  
 **return null**;  
 }  
 **catch**(IOException e){  
 Log.*e*(**"Error: "**, e.getMessage(), e);  
 **return null**;  
 }  
 **return** document;  
 }  
 **public** String getValue(Element item, String name)  
 {  
 NodeList nodes = item.getElementsByTagName(name);  
 **return this**.getTextNodeValue(nodes.item(0));  
 }  
 **private final** String getTextNodeValue(Node elem) {  
 Node child;  
 **if**( elem != **null**){  
 **if** (elem.hasChildNodes()){  
 **for**( child = elem.getFirstChild(); child != **null**; child = child.getNextSibling() ){  
 **if**( child.getNodeType() == Node.***TEXT\_NODE*** ){  
 **return** child.getNodeValue();  
 }  
 }  
 }  
 }  
 **return ""**;  
 }  
}

* 1. **DATABase SQLite**

public static SQLiteDatabase initDatabase(Activity activity, String databaseName){

try {

String outFileName = activity.getApplicationInfo().dataDir + "/databases/" + databaseName;

File f = new File(outFileName);

if(!f.exists()) {

InputStream e = activity.getAssets().open(databaseName);

File folder = new File(activity.getApplicationInfo().dataDir + "/databases/");

if (!folder.exists()) {

folder.mkdir();

}

FileOutputStream myOutput = new FileOutputStream(outFileName);

byte[] buffer = new byte[1024];

int length;

while ((length = e.read(buffer)) > 0) {

myOutput.write(buffer, 0, length);

}

myOutput.flush();

myOutput.close();

e.close();

}

} catch (IOException e) {

e.printStackTrace();

}

return activity.openOrCreateDatabase(databaseName, Context.MODE\_PRIVATE, null);

}

* 1. **DATABase SQLite**

package com.example.nguyentuan.tracnghiem1.question;

/\*\*

\* Created by NguyenTuan on 11/10/2016.

\*/

import android.annotation.TargetApi;

import android.content.Context;

import android.database.sqlite.SQLiteDatabase;

import android.database.sqlite.SQLiteException;

import android.database.sqlite.SQLiteOpenHelper;

import android.os.Build;

import java.io.File;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.InputStream;

import java.io.OutputStream;

import java.sql.SQLException;

public class DBHelper extends SQLiteOpenHelper {

private static String DB\_PATH = "/data/data/com.example.nguyentuan.tracnghiem1.question/databases/";

private static String DB\_NAME = "dbtracnghiem.sqlite";

private static final int DATABASE\_VERSION = 1;

private SQLiteDatabase myDataBase;

private final Context myContext;

public DBHelper(Context context) {

super(context, DB\_NAME, null, DATABASE\_VERSION);

this.myContext = context;

}

public void openDataBase() throws SQLException {

//Open the database

String myPath = DB\_PATH + DB\_NAME;

myDataBase = SQLiteDatabase.openDatabase(myPath, null, SQLiteDatabase.OPEN\_READONLY);

}

@TargetApi(Build.VERSION\_CODES.JELLY\_BEAN)

public void deleteDataBase() {

String myPath = DB\_PATH + DB\_NAME;

SQLiteDatabase.deleteDatabase(new File(myPath));

}

@Override

public synchronized void close() {

if (myDataBase != null)

myDataBase.close();

super.close();

}

private boolean checkDataBase() {

SQLiteDatabase checkDB = null;

try {

String myPath = DB\_PATH + DB\_NAME;

checkDB = SQLiteDatabase.openDatabase(myPath, null, SQLiteDatabase.OPEN\_READONLY);

} catch (SQLiteException e) {

//database chua ton tai

}

if (checkDB != null)

checkDB.close();

return checkDB != null ? true : false;

}

private void copyDataBase() throws IOException {

//mo db trong thu muc assets nhu mot input stream

InputStream myInput = myContext.getAssets().open(DB\_NAME);

//duong dan den db se tao

String outFileName = DB\_PATH + DB\_NAME;

//mo db giong nhu mot output stream

OutputStream myOutput = new FileOutputStream(outFileName);

//truyen du lieu tu inputfile sang outputfile

byte[] buffer = new byte[1024];

int length;

while ((length = myInput.read(buffer)) > 0) {

myOutput.write(buffer, 0, length);

}

//Dong luon

myOutput.flush();

myOutput.close();

myInput.close();

}

public void createDataBase() throws IOException {

boolean dbExist = checkDataBase(); //kiem tra db

if (dbExist) {

//khong lam gi ca, database da co roi

// copyDataBase();

} else {

this.getReadableDatabase();

try {

copyDataBase(); //chep du lieu

} catch (IOException e) {

throw new Error("Error copying database");

}

}

}

@Override

public void onCreate(SQLiteDatabase db) {

}

@Override

public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

}

}