# DATAPICKER & TIMEPICKER

##### Chọn Calendar (java.util)

### Disable Date (Sử dụng trong trường hợp muốn ẩn date không cho người dùng chọn vào {Đặt vé máy bay, giới hạn số ngày...})

calendar.getDatePicker().getMinDate([Truyền vào milisecond bắt đầu]);

calendar.getDatePicker().getMaxDate([Truyền vào milisecond kết thúc]);

#### TimePicker Cách 1:

private void ShowTime() {

TimePickerDialog.OnTimeSetListener callBack = new TimePickerDialog.OnTimeSetListener() {

@Override

public void onTimeSet(TimePicker view, int hourOfDay, int minute) {

calendar.set(calendar.HOUR, hourOfDay);

calendar.set(calendar.MINUTE, minute);

txtTime.setText(simpleDateFormatTime.format(calendar.getTime()));

}

};

TimePickerDialog timePickerDialog = new TimePickerDialog(MainActivity.this,

callBack,

calendar.get(calendar.HOUR),

calendar.get(calendar.MINUTE), true); // true : định dạng 24h/ false: 12h

timePickerDialog.show();

}

#### DatePicker Cách 1:

private void ShowDate() {

### // Tạo hàm OnDateSetListener: callBack

final DatePickerDialog.OnDateSetListener callBack = new DatePickerDialog.OnDateSetListener() {

@Override

public void onDateSet(DatePicker view, int year, int month, int dayOfMonth) {

calendar.set(calendar.YEAR, year);

calendar.set(calendar.MONTH, month);

calendar.set(calendar.DAY\_OF\_MONTH, dayOfMonth);

// calendar.set(year, month, dayOfMonth); // Có thể dùng như này cho nhanh

txtDate.setText(simpleDateFormatDate.format(calendar.getTime()));

}

};

### // Khởi tạo DatePickerDialog: hiển thị cửa sổ chọn Date

DatePickerDialog datePickerDialog = new DatePickerDialog(MainActivity.this,

callBack,

calendar.get(calendar.YEAR),

calendar.get(calendar.MONTH),

calendar.get(calendar.DAY\_OF\_MONTH));

datePickerDialog.show();

}

# EXIT APPLICATION

Intent intent = new Intent(Intent.ACTION\_MAIN);

intent.addCategory(Intent.CATEGORY\_HOME);

intent.setFlags(Intent.FLAG\_ACTIVITY\_NEW\_TASK);

startActivity(intent);

# SHARE PREFERENCES

SharedPreferences sharedPreferences;

sharedPreferences = getSharedPreferences("Data", MODE\_PRIVATE);

### // Gán dữ liệu vào

SharedPreferences.Editor editor = sharedPreferences.edit();

editor.putString("TenDN", user);

editor.putString("MatKhau", pass);

editor.putBoolean("Remember", true);

editor.commit(); // Nhớ câu này

### // Xóa dữ liệu trong biến:

editor.remove("TenDN");

### // Lấy dữ liệu ra

edtPass.setText(sharedPreferences.getString("MatKhau", ""));

edtUser.setText(sharedPreferences.getString("TenDN", ""));

ckbRemember.setChecked(sharedPreferences.getBoolean("Remember", false));

# CHECKBOX:

### Có 2 kiểu bắt sự kiện

#### 1. Thực hiện liền ngay khi bấm

cbAndroid.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {

@Override

public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {

### // isChecked tra ket qua khi check

if (isChecked){

// TODO:

}

else {

//TODO:

}

}

});

#### 2. Xác nhận rồi mới thực hiện (Có Checked hay không mới thực hiện, Vd điền Form xong mới lưu xuống SQL)

Sử dụng isChecked == true; hoặc isChecked == false;

# CHECKBOX IN LISTVIEW

for (int i = lvRemember.getChildCount() -1; i >= 0;i--) {

View v = lvRemember.getChildAt(i); // Lấy 1 dòng trong Listview

CheckBox ckbDelete = (CheckBox) v.findViewById(R.id.checkBox);

if (ckbDelete.isChecked()) { // Xác nhận checkbox có check hay không

arrayRemember.remove(i);

}

}

# RADION GROUP

#### 1. Bắt sự kiện CLick trên mỗi Radion Button của RadionGroup

radioGroup.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener() {

@Override

public void onCheckedChanged(RadioGroup radioGroup, int i) {

### // i <=> id của các radionButton

switch (i) {

case R.id.radioButton5:

Toast.makeText(MainActivity.this, "1", Toast.LENGTH\_SHORT).show();

case R.id.radioButton6:

Toast.makeText(MainActivity.this, "2", Toast.LENGTH\_SHORT).show();

}

}

});

#### 2. Lấy ID của radionButton được check

int selectedId = radioGroup.getCheckedRadioButtonId();

# INTENT:

#### 1. TRONG APP

#### Truyền Bundle : Đóng gói các kiểu dữ liệu khác lại trong biến Bundle

### // Truyền:

Intent intent = new Intent(MainActivity.this, SecondActivity.class);

CSinhVien sinhVien = new CSinhVien("Vũ Quốc Thanh", 1412489);

### //Để truyền obj phải là kiểu Serializable. Implement ngay trong lớp CSinhVien luôn

Bundle bundle = new Bundle();

bundle.putString("chuoi", "Vu Thanh"); // Định dạng String

bundle.putInt("so", 247); // Định dạng Int

bundle.putStringArray("arrayName", arrayName); // Định dạng ArrayString (vd: HocSinh[])

bundle.putSerializable("obj",sinhVien); // Định dạng obj

intent.putExtra("data", bundle); // Truyền biến bundle qua màn hình Second

startActivity(intent);

### // Nhận:

Intent myIntent = getIntent(); // Lấy từ Intent

Bundle myBundle = myIntent.getBundleExtra("data"); // Nhận gói Bundle như 1 kiểu dữ liệu bình thường.

if (myBundle != null) { // Có thì lấy, Không có thì bỏ qua không cần xử lý!

### // Tách trong gói Bundle ra từng thành phần đã gửi bên kia

String \_str = myBundle.getString("chuoi");

Integer \_number = myBundle.getInt("so");

arrayName = myBundle.getStringArray("arrayName");

String array = "";

for (int i = 0;i< arrayName.length; i++) {

array = array + arrayName[i] + "\n";

}

CSinhVien sinhVien = (CSinhVien) myBundle.getSerializable("obj");

txtName.setText("Chuoi: " + \_str + "\n" +

"So: " + \_number + "\n" +

"Mang: " + array + "\n" +

"Object: " + sinhVien.getName() + "\n" + sinhVien.getID());

}

# Intent Website:

Intent intent = new Intent(); // do khong rõ truyen từ đâu đến đâu nên k biết

intent.setAction(Intent.ACTION\_VIEW); // Mở ra để làm gì? Ở đây chọn VIEW: CÒn rất nhiều phương thức

intent.setData(Uri.parse("http://vuquocthanh.blogspot.com")); // Đường dẫn tới web

startActivity(intent);

# Intent Messager: Điền sẵn nội dung, không điền người nhận.

Intent intent = new Intent();

intent.setAction(Intent.ACTION\_SEND); // Sent tin nhắn.

intent.putExtra(Intent.EXTRA\_TEXT, "Noi dung: .............");

intent.setType("text/plain"); // kiểu tin nhắn là text/plain

startActivity(intent);

# Intent Message: Mở tin nhắn có nội dung và người nhận.

Intent intent = new Intent();

intent.setAction(Intent.ACTION\_SENDTO);

intent.putExtra("sms\_body","Anh yeu em!"); // Nội dung

intent.setData(Uri.parse("sms:01699716055")); // Số điện thoại nhận

startActivity(intent);

# Intent Email: Gửi tin nhắn qua Gmail

String email = txtSms.getText().toString(); // Lấy nội dung cần gửi trên giao diện

Intent intent = new Intent(Intent.ACTION\_SEND);

intent.setType("text/plain");

intent.putExtra(Intent.EXTRA\_EMAIL, "vqthanh1412489@gmail.com"); // Đại chỉ người gửi

intent.putExtra(Intent.EXTRA\_SUBJECT, "Anh yeu Em!"); // Tiêu đề

intent.putExtra(Intent.EXTRA\_TEXT, email); // Nội dung

startActivity(Intent.createChooser(intent, "Send Email"));

# Intent Settings Call Dialog: Mở hộp thọi Call (4.0)

Intent intent = new Intent();

intent.setAction(Intent.ACTION\_DIAL);

intent.setData(Uri.parse("tel:01699716055");

startActivity(intent);

# Intent Settings Call: Mở hộp thoại rồi gọi luôn.

Intent intent = new Intent();

intent.setAction(Intent.ACTION\_CALL);

intent.setData(Uri.parse("tel:01699716055"));

startActivity(intent);

##### // Nhớ thêm bên Android Menifes dòng này để xin quyền người dùng khi gọi thẳng...:

##### //<uses-permission android:name="android.permission.CALL\_PHONE"></uses-permission>

# Intent Settings Wifi:

#### // Cách 1:

Intent intent = new Intent(Settings.ACTION\_WIFI\_SETTINGS);

startActivity(intent);

#### // Cách 2:

startActivityForResult(new Intent(Settings.ACTION\_WIFI\_SETTINGS), 0);

#### // Cách 3:

Intent intent = new Intent();

intent.setAction(Intent.ACTION\_MAIN);

intent.addCategory(Intent.CATEGORY\_LAUNCHER);

final ComponentName cn = new ComponentName("com.android.settings", "com.android.settings.wifi.WifiSettings");

intent.setComponent(cn);

intent.setFlags(Intent.FLAG\_ACTIVITY\_NEW\_TASK);

startActivity( intent);

# Intent Settings Bluetooth:

Intent intent = new Intent(Settings.ACTION\_BLUETOOTH\_SETTINGS);

startActivity(intent);

# Intent nhận kết quả trả về Thông qua startActivityForResult

#### //Màn Hình chính:

Intent intent = new Intent(MainActivity.this, Main2Activity.class);

startActivityForResult(intent, 1234); // Intent này sẽ nhận về kết quả! Thông qua phương thức setResult bên màn hình bên kia.

### // Mã gửi đi, mã gửi về, dữ liệu nhận về

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

if (requestCode == 1234 && resultCode == RESULT\_OK && data != null) {

String result1 = data.getStringExtra("data"); // Đổ kết quả từ data ra biến bên này!

txtName.setText(result1 + "\n" + result2); // Gán cho biến bên này

}

super.onActivityResult(requestCode, resultCode, data);

}

#### // Màn hình trung gian(Có thể là hộp thoại, camera, gallery...)

String \_name2 = edtName.getText().toString();

String \_id = edtID.getText().toString();

Intent intentReturn = new Intent(); // Nó truyền lại cho màn hình cha nên không cần chỉ số màn hình!

intentReturn.putExtra("data", \_name2);

setResult(RESULT\_OK, intentReturn); // Trả về kết quả lại màn hình cha! Mã code RESULT\_OK để chuyển qua kia lấy mã xác nhận màn hình !

finish(); // Hạn chế gọi Intent về nhé!

# INTETNT CAMERA (<6.0)

Button btnCamera;

ImageView imgHinh;

int REQUEST\_CODE\_CAMERA = 247;

btnCamera.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent intent = new Intent(MediaStore.ACTION\_IMAGE\_CAPTURE);

startActivityForResult(intent, REQUEST\_CODE\_CAMERA);

}

});

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

### // Camera co 3 nut: Hủy, Chup, Chọn, ở đây lấy chọn

if (requestCode == REQUEST\_CODE\_CAMERA && resultCode == RESULT\_OK && data != null) {

Bitmap bitmap = (Bitmap) data.getExtras().get("data"); // Chỉ lấy kiểu chung (Chữ data là mặc định)

imgHinh.setImageBitmap(bitmap);

}

super.onActivityResult(requestCode, resultCode, data);

}

# INTENT CAMERA AND FORDER( > 6.0)

final int REQUEST\_CODE\_CAMERA = 111;

final int REQUEST\_CODE\_FOLDER = 222;

#### 1. Camera

imgButtonCamera.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

ActivityCompat.requestPermissions(

ThemSanPhamActivity.this,

new String[]{android.Manifest.permission.CAMERA},

REQUEST\_CODE\_CAMERA );

}

});

#### 2. Folder

imgButtonFolder.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

ActivityCompat.requestPermissions(

ThemSanPhamActivity.this,

new String[]{android.Manifest.permission.READ\_EXTERNAL\_STORAGE},

REQUEST\_CODE\_FOLDER );

}

});

#### 3. Hàm onRequestPermissionsResult

@Override

public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {

switch (requestCode){

case REQUEST\_CODE\_CAMERA:{

if(grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION\_GRANTED){

Intent intentCamera = new Intent(MediaStore.ACTION\_IMAGE\_CAPTURE);

startActivityForResult(intentCamera, REQUEST\_CODE\_CAMERA);

} else {

Toast.makeText(this, "Không cho phép truy cập Camera", Toast.LENGTH\_SHORT).show();

}

return;

}

case REQUEST\_CODE\_FOLDER:{

if(grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION\_GRANTED){

Intent intentFolder = new Intent(Intent.ACTION\_PICK); // lấy media ra

intentFolder.setType("image/\*"); //Chọn kiểu hình ảnh

startActivityForResult(intentFolder, REQUEST\_CODE\_FOLDER);

} else {

Toast.makeText(this, "Không cho phép truy cập Folder", Toast.LENGTH\_SHORT).show();

}

return;

}

}

super.onRequestPermissionsResult(requestCode, permissions, grantResults);

}

#### 4. Hàm onActivityResult

@Override

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

if(requestCode == REQUEST\_CODE\_FOLDER && resultCode == RESULT\_OK && data != null){

Uri uriFolder = data.getData(); // đường dẫn tới thư mục chứa ảnh

//Cach 1:

imgHinh.setImageURI(uriFolder); // Gán thẳng đường dẫn hình cho imgHinh

//Cách 2:

Picasso.with(MainActivity.this).load(uriFolder).into(imgHinh); // Sử dụng thư viện Picasso

// Cách 3:

// try {

// InputStream inputStream = getContentResolver().openInputStream(uriFolder);

// Bitmap bitmapFolder = BitmapFactory.decodeStream(inputStream);

// imgHinh.setImageBitmap(bitmapFolder);

// } catch (FileNotFoundException e) {

// e.printStackTrace();

// }

}

if(requestCode == REQUEST\_CODE\_CAMERA && resultCode == RESULT\_OK && data != null){

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

imgHinh.setImageBitmap(bitmapCamera);

}

super.onActivityResult(requestCode, resultCode, data);

}

# SHOW PASSWORRD CHECKBOX

cbShowPassword.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {

@Override

public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {

if (isChecked)

etPassword.setTransformationMethod(null);

else

etPassword.setTransformationMethod(PasswordTransformationMethod.getInstance()); // Hiển thị ở dang password

}

});

# MENU ACTIONBAR

#### 1. Tạo layout menu:

<menu xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"> // Thêm dòng này để bỏ lỗi

//<!--Đặt id để bắt sự kiện nhé!-->

// android:orderInCategory="1": Chỉnh thứ tự của menu

<item android:id="@+id/menuSetting"

android:title="Setting" android:orderInCategory="1"></item>

<item android:id="@+id/menuShare"

android:title="Share" android:orderInCategory="3"></item>

//<!--Để android:showAsAction="always" sẽ có lỗi:

// Thêm thư viện xmlns:app="http://schemas.android.com/apk/res-auto" và sử dụng app:showAsAction để sửa-->

<item android:id="@+id/menuReload"

android:title="Reload"

android:icon="@drawable/replay"

app:showAsAction="always"

android:orderInCategory="2"></item>

</menu>

#### 2. Tao Menu: Gọi nó ra

@Override

public boolean onCreateOptionsMenu(Menu menu) {

getMenuInflater().inflate(R.menu.menu\_reload, menu);

return super.onCreateOptionsMenu(menu);

}

#### 3. Bắt sự kiện người dùng chọn Menu:

@Override

public boolean onOptionsItemSelected(MenuItem item) {

switch (item.getItemId()) {

case R.id.menuSetting:

{

//TODO:

break;

}

case R.id.menuReload:

{

//TODO:

break;

}

}

return super.onOptionsItemSelected(item);

}

# MENU CONTEXT (Chỉ Long Click)

#### 1. Khai báo

@Override

public void onCreateContextMenu(ContextMenu menu, View v, ContextMenu.ContextMenuInfo menuInfo) {

getMenuInflater().inflate(R.menu.menu\_edit\_delete, menu);

super.onCreateContextMenu(menu, v, menuInfo);

}

#### 2. Bắt sự kiện Click lên các Opition

@Override

public boolean onContextItemSelected(MenuItem item) {

switch (item.getItemId()) {

case R.id.menuSetting:

{

//TODO:

break;

}

case R.id.menuReload:

{

//TODO:

break;

}

}

return super.onContextItemSelected(item);

}

#### 3. Đăng ký View nào sẽ nhận Menu này

registerForContextMenu(lvSanPham); // Ở đây chọn ListView

# MENU POPUP MENU (Nên gán vào sự kiện Click)

#### 1. Gắn sự kiện cho View (ở đây chọn Listview)

listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {

@Override

public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {

pos = i;

showPopupMenu(view);

}

});

#### 2. Hàm Hiển thị và bắt sự kiện

private void showPopupMenu(View view) {

PopupMenu popup = new PopupMenu(this, view);

MenuInflater inflater = popup.getMenuInflater();

inflater.inflate(R.menu.menu\_popup, popup.getMenu());

popup.show();

### // Bắt sự kiện

popup.setOnMenuItemClickListener(new PopupMenu.OnMenuItemClickListener() {

@Override

public boolean onMenuItemClick(MenuItem menuItem) {

switch(menuItem.getItemId()){

case R.id.call:

{

//TODO:

return true;

}

case R.id.call:

{

//TODO:

return true;

}

default: return false;

}

}

});

}

# TOAST ADVANCE

LayoutInflater inflater = getLayoutInflater();

View layout = inflater.inflate(R.layout.thongbao, (ViewGroup) findViewById(R.id.layoutThongBao));

Toast t = new Toast(this);

t.setView(layout);

t.setGravity(Gravity.CENTER,0,0);

t.show();

# ĐỌC GHI FILE .TXT

#### 1. GHI

// Truyền vào Filename cần tạo và nội dung file

private void WrireFileText(String fileName, String noidung) throws IOException {

FileOutputStream fileOutputStream = openFileOutput(fileName, Context.MODE\_PRIVATE);

fileOutputStream.write(noidung.getBytes());

fileOutputStream.close(); // Nhớ Close();

Toast.makeText(this, "Created " + fileName, Toast.LENGTH\_SHORT).show();

}

#### 2. ĐỌC

# SHUFFLE ARRAYLIST

ArrayList<String> array = new ArrayList<String>();

Collections.shuffle(array);

##### // Lưu ý: Không trộn được String[];

##### // Để trộn được cần chuyển thành ArrayList:

String []arrayString;

array = new ArrayList<>(Arrays.asList(arrayString));

# GET DATA RESOURCE

String chuoi;

chuoi = getResource().getStringArray(R.array.tenMang.get(0); // Lấy 1 phần từ thứ 0 trong mảng nào đó gán vào chuỗi

# Chuyển tên trong Resource Thành id (Thường thì dùng để chuyển tên hình trong drawable):

int id;

id = getResource().getIdentifier( "hinh1","drawable",getPackageName());

#### //1. Tên Resource cần lấy

#### //2. Kiểu : drawable, values, string, layout, mipmap....

#### //3. Tên packKage của Project

# Table Layout trên code:

TableLayout tableLayout = (TableLayout) findViewById(R.id.myTableImage);

int numRow;

int nunCol;

for (int i = 1; i <= numRow; i++){

TableRow tableRow = new TableRow(this); // this: tạo ngay trên cái TableLayout này.

tableLayout.addView(tableRow); // Gắn dòng vào Layout

for (int j = 1; j <= numCol; j++){

ImageView imageView = new ImageView(this); //

final int vitri = numCol \* (i-1) + j - 1; // sao cho biến vị trí chạy từ 0 - > 17 là đc. ở đây chạy theo i và j

int idHinh = getResources().getIdentifier(MainActivity.mangTenHinh.get(vitri), "drawable", getPackageName()); // Lấy hết hình ra

// Gọi lại mảng bên MainActivity . để static public

imageView.setImageResource(idHinh); // gắn vào imageView trong Tablelayout.

// A: Chỉnh kích thước mỗi ô trong TableLayout cố định theo màn hình:

//TableRow.LayoutParams layoutParams = new TableRow.LayoutParams(180, 180); // Kích thước cua từng tấm hình là 180px, 180px

//imageView.setLayoutParams(layoutParams); // Kích hoạt mới dùng được

// Có thể viết gọn: imageView.setLayoutParams(TableRow.LayoutParams(180,180));

// B: Chỉnh kích thước mỗi ô tùy theo độ phân giải màn hình:

//1. Chuyển DP thành PX : để hình không bị vỡ do độ phân giải màn hình:

int dpUnit = 80; // Số này có thể chỉnh cho phù hợp

Resources r = getResources();

float px = TypedValue.applyDimension(TypedValue.COMPLEX\_UNIT\_DIP, dpUnit, r.getDisplayMetrics());

TableRow.LayoutParams layoutParams= new TableRow.LayoutParams((int) px,(int) px);

imageView.setLayoutParams(layoutParams);

tableRow.addView(imageView); // Add hình vào dòng

}

}

# SQLite:

#### Bước 1: Khởi tạo Class SQLite để thực hiện câu lệnh SQLite:

public class SQlite extends SQLiteOpenHelper {

### // Tạo phương thức khởi tạo:

public SQlite(Context context, String name, SQLiteDatabase.CursorFactory factory, int version) {

super(context, name, factory, version);

}

### // Tạo hàm thực hiện câu lệnh KHÔNG trả kết quả: INSERT, DELETE, UPDATE, CREATE

// Tạo hàm thực hiện câu lệnh CÓ trả kết quả về: SELECT

public Cursor GetData(String sql){

SQLiteDatabase database = getReadableDatabase();

return database.rawQuery(sql, null);}

### // Tạo hàm tùy chọn: Thực hiện một lệnh bất kỳ, với kiểu dữ liệu khác (Hình ảnh: BLOB)

public void InsertSach(String Ten, int Gia, byte[] Hinh) {

SQLiteDatabase database = getWritableDatabase();

String sql = "Insert Into Sach Values (null, ?, ?, ?)";

SQLiteStatement statement = database.compileStatement(sql);

statement.clearBindings();

statement.bindString(1, Ten);

statement.bindLong(2, Gia);

statement.bindBlob(3, Hinh);

statement.executeInsert();

}

}

#### Bước 2: Khởi tạo database, Tao bảng (CREATE TABLE), (INSERT INTO), Chọn bảng (SELECT)

SQLite database;

### // Tạo database

database = new SQlite(this, "TruongHoc.sqlite", null, 1);

### // Tạo bảng

database.QueryData("CREATE TABLE IF NOT EXISTS HocSinh(Id INTEGER PRIMARY KEY AUTOINCREMENT, HoTen VARCHAR, NamSinh INTEGER)");

### // Thêm dữ liệu cố định vào bảng

database.QueryData("INSERT INTO HocSinh(Id, HoTen, NamSinh) VALUES(null, 'Khoa Hoc Tu Nhien', 1995)");

### // Button Thêm dữ liệu vào bảng

btnNhap.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

\_name = edtHoTen.getText().toString();

\_namsinh = edtNamSinh.getText().toString();

database.QueryData("INSERT INTO HocSinh(HoTen, NamSinh) VALUES ('"+\_name+"', '"+\_namsinh+"')");

Toast.makeText(MainActivity.this, "Added", Toast.LENGTH\_SHORT).show();

### // Co tra ve

LoadData();

}

});

// Hàm LoadData() : Tất cả dữ liệu từ bảng SQl sẽ được Load lên mảng Array theo danh sách từng đối tượng.

private void LoadData() {

cursorHocSinh = database.GetData("SELECT \* FROM HocSinh");

arrayLisHocSinh.clear(); // Xóa hết dữ liệu trong mảng. tránh bị trùng

### // Nếu chạy còn dữ liệu thì chạy vô while

while (cursorHocSinh.moveToNext()) {

### // Thu tu theo thu tu cua con tro

String hoten = cursorHocSinh.getString(1); // Cot thu 1 //String hoten = cursorHocSinh.getColumnName(1); // Tra ra ten cot, khong tra gia tri trong cot nhe!

int id = cursorHocSinh.getInt(0); // Cot thu 0

int namSinh = cursorHocSinh.getInt(2); // Cot thu 2

// Add mang ne, Không thêm từ mảng dữ liệu được lấy từ SQL

arrayLisHocSinh.add(new CHocSinh(hoten, namSinh));

}

adapterHocSinh.notifyDataSetChanged();

}

# Lấy file trong thư viện Gallery

int REQUESTCODE\_FOLDER = 247;

ibtnFolder.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(Intent.ACTION\_PICK); // Lấy file ra trong thư viện (nhiều cái, có cả hình)

intent.setType("image/\*"); // Lấy cái liên quan tới hình ảnh thôi

startActivityForResult(intent, REQUESTCODE\_FOLDER);

}

});

##### // Nếu có lõi thì cần cấp quyền cho nó để mở thư mục trong máy ra!

##### // Trên máy 5.0 > thì có thể có lỗi

##### // <uses-permission android:name="android.permission.CAMERA"></uses-permission> // Xin quyền camera

##### // <uses-permission android:name="android.permission.READ\_EXTERNAL\_STORAGE"></uses-permission> // Xin quyền mở bộ nhớ của máy

#### // Hứng dữ liệu từ Chọn hình

if (requestCode == REQUESTCODE\_FOLDER && resultCode == RESULT\_OK && data != null) {

Uri uri = data.getData(); // Khái báo đường dẫn trong Android

### // Thêm Try catch lỡ k tìm thấy đướng dẫn trong máy. Đảm bảo app không lỗi

try {

InputStream inputStream = getContentResolver().openInputStream(uri); // Truyền đường đẫn để lấy dữ liệu vào. Chứa vào biến inputStream

Bitmap bitmap = BitmapFactory.decodeStream(inputStream); // Bitmap có tấm hình lấy ra từ biến inputStream

imgSanPham.setImageBitmap(bitmap);

} catch (FileNotFoundException e) {

e.printStackTrace();

}

}

##### // Có thẻ sử dụng Thư viện Picasso(Ngắn hơn)

@Override

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

if (requestCode == REQUEST\_CODE\_FOLDER && resultCode == RESULT\_OK && data != null) {

Uri uri = data.getData();

Picasso.with(MainActivity.this).load(uri).into(imgView);

}

super.onActivityResult(requestCode, resultCode, data);

}

# CONVERT BITMAP TO BYTE[]

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

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

Bitmap bmp = drawable.getBitmap(); // Lấy dữ liệu trên hình chuyển về bitmap

ByteArrayOutputStream stream = new ByteArrayOutputStream();

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

byte[] byteArray = stream.toByteArray();

return byteArray;

}

# CONVERT BYTE[] TO BITMAP

Bitmap bitmap = BitmapFactory.decodeByteArray(sach.getHinh(), 0, sach.getHinh().length); // Chuyển từ Byte thwu -> hết mảng kiểu byte[] chứa hình

imgHinh.setImageBitmap(bitmap);

# SQLite Sử dụng SQLiteManager

#### 1. Hàm kết nối file \*.sqlite với Activity

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);

}

#### 2. Khai báo tên file \*.sqlite + biến database (biến thực thi câu lệnh, hàm) + Cursor (duyệt Bảng dữ liệu)

public static String DatabaseName = "ThuVien.sqlite";

public static SQLiteDatabase database;

public static Cursor cursorBook;

#### 3. Hàm Đưa Dữ liệu lên

public static void loadData() {

cursorBook = database.rawQuery("Select \* From Sach", null);

arrayBook.clear();

while (cursorBook.moveToNext()) {

int \_id = cursorBook.getInt(0);

String \_title = cursorBook.getString(1);

int \_number = cursorBook.getInt(2);

String \_author = cursorBook.getString(3);

byte[] \_avar = cursorBook.getBlob(4);

arrayBook.add(new CBook(\_id, \_title, \_number, \_author, \_avar));

}

bookAdapter.notifyDataSetChanged();

}

#### 4. Thực hiện câu lệnh SELECT

database = Database.initDatabase(EditActivity.this, MainActivity.DatabaseName); // Gắn database vào màn hình cần sử dụng

MainActivity.cursorBook = MainActivity.database.rawQuery("Select \* From Sach Where Id = '" + Id + "'", null);

#### 5. Thực hiện câu lệnh UPDATE

btnSave.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String \_title = edtTitle.getText().toString();

String \_number = edtNumber.getText().toString();

String \_author = edtAuthor.getText().toString();

byte[] \_avar = ImageView\_To\_Byte(imgAvar);

//Khi INSERT thi Update nguyen 1 Object

ContentValues contentValues = new ContentValues();

contentValues.put("Title", \_title);

contentValues.put("Page\_count", \_number);

contentValues.put("Author", \_author);

contentValues.put("Avar", \_avar);

database = Database.initDatabase(EditActivity.this, MainActivity.DatabaseName);

database.update("Book", contentValues, "Id = '"+Id+"'", null);

// Tên bảng, giá trị, Điều kiện where, mảng string (string[]{truyền từng giá trị vào....})

// VD: database.update("Book", contentValues, "Id = ?", string[]{Id + ""});

MainActivity.loadData();

}

});

#### 6. Thực hiện câu lệnh INSERT

btnSave.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

MainActivity.database = DataBase.initDatabase(AddActivity.this, MainActivity.DataBaseName);

String \_name = edtName.getText().toString();

String \_phone = edtPhone.getText().toString();

byte[]\_avar = ImageView\_To\_Byte(imgAvar);

// Khi INSERT thi Update nguyen 1 Object

ContentValues contentValues = new ContentValues();

contentValues.put("Name", \_name);

contentValues.put("Phone", \_phone);

contentValues.put("Avar", \_avar);

MainActivity.database.insert("SinhVien", null, contentValues);

// Tên bảng, null, giá trị thêm

Toast.makeText(AddActivity.this, "Thêm Thành Công", Toast.LENGTH\_SHORT).show();

}

});

#### 6. Thực hiện câu lệnh DELETE

MainActivity.database = DataBase.initDatabase(context, MainActivity.DataBaseName); // Ket noi databasevoi man hinh

MainActivity.database.delete("SinhVien", "Id = '" + student.getId() + "'", null);

// Tên bảng, Điều kiện xóa, có thể để null hoặc là mảng String truyền giá trị của biến

MainActivity.loadData();

# CUSTOM DIALOG

#### 1. Tạo layout cho dialog cần hiển thị lên

#### 2. Tạo trong code .java

Dialog dialog = new Dialog(MainActivity.this); // Không có giao diện, chỉ gọi thôi.Xuất hiện trên màn hình này

### // Gán Layout cho Dialog:

dialog.setContentView(R.layout.custom);

dialog.setTitle("Login");

dialog.setCanceledOnTouchOutside(false); // Không cho kich bên ngoài để out(Dùng tùy trường hơp nhé!)

dialog.show();// Hiện hộp thoại

dialog.dismiss(); // Ẩn hộp thoại có thể dùng cancel();

### // Lấy id bên view dialog.. nhớ sau setContentView(Khai báo những view ben tren dialog)

edtUser = (EditText) dialog.findViewById(R.id.editTextUser);

edtPass = (EditText) dialog.findViewById(R.id.editTextPass);

# CUSTOM DIALOG (Advandce)

private void displayAlertDialog() {

LayoutInflater inflater = getLayoutInflater();

View alertLayout = inflater.inflate(R.layout.dialog\_gmail, null);

EditText edtSubject = (EditText) alertLayout.findViewById(R.id.editTextSubject);

Button btnOke = (Button) alertLayout.findViewById(R.id.buttonOke);

AlertDialog.Builder builder = new AlertDialog.Builder(this);

builder.setTitle("Subject");

builder.setView(alertLayout);

builder.setPositiveButton("Đồng ý", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialogInterface, int i) {

}

});

builder.setNegativeButton("Huy", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialogInterface, int i) {

Toast.makeText(SMSContentActivity.this, "Bạn đã hủy", Toast.LENGTH\_SHORT).show();

}

});

builder.create().show();

}

# BACK DOUBLE

boolean checkClick = false;

@Override

public void onBackPressed() {

if (checkClick) {

super.onBackPressed();

return;

}

checkClick = true;

Toast.makeText(this, "Press Back again to Exit!", Toast.LENGTH\_SHORT).show();

Handler handler = new Handler();

handler.postDelayed(new Runnable() {

@Override

public void run() {

checkClick = false;

}

}, 2000); // Số 2000 để tùy chính khoảng cách 2 lần Back

}

# Lấy đường dẫn từ Resource trong Android

"android.resource://" + getPackageName() + "/" + R.id.tenfile;

# LOAD HÌNH TỪ INTERNET VỀ APP

#### // 1. Xây dựng class AysncTask để chạy đa tiến trình

private class LoadPicture extends AsyncTask<String, String, Bitmap> {

### // Hình ngoài app nhận vào Bitmap

@Override

protected void onPreExecute() {

super.onPreExecute();

}

@Override

protected Bitmap doInBackground(String... strings) {

Bitmap bitmapHinh = null;

try {

URL url = new URL("http://vanhienblog.com/wp-content/uploads/2016/10/hinh-anh-meo-con.jpg"); // Truyền đường dẫn

bitmapHinh = BitmapFactory.decodeStream(url.openConnection().getInputStream()); // Kết nối tới file trên Internet tải hình về Bitmap

} catch (MalformedURLException e) {

e.printStackTrace();

} catch (IOException e) {

e.printStackTrace();

}

return bitmapHinh;

}

@Override

protected void onPostExecute(Bitmap bitmap) { // Hàm nhận kết quả trả về

super.onPostExecute(bitmap);

imgHinh.setImageBitmap(bitmap);

}

#### // 2. Tạo tiến trình nhỏ để load Bitmap vào ImageView

btnDownload.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

runOnUiThread(new Runnable() {

@Override

public void run() {

new LoadPicture().execute(); // Nên truyền tham số vào

}

});

}

});

# ĐỌC THẺ <IMA XML

Pattern pattern = Pattern.compile("<img[^>]+src\\s\*=\\s\*['\"]([^'\"]+)['\"][^>]\*>");

# READ URL INTERNET

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

{

StringBuilder content = new StringBuilder(); // Đọc đẩy vào content

try

{

// create a url object

URL url = new URL(theUrl);

// create a urlconnection object

URLConnection urlConnection = url.openConnection(); // Mở kết nối ra Internet

// wrap the urlconnection in a bufferedreader// Để nội dung đọc liên tục (Buffere) từ connection vào Buffere

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

String line;

// read from the urlconnection via the bufferedreader

while ((line = bufferedReader.readLine()) != null) // Đọc theo từng dòng. Đọc dến dòng tiếp theo thì dừng

{

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

}

bufferedReader.close();

}

catch(Exception e)

{

e.printStackTrace();

}

return content.toString(); // Lấy ra chuỗi đã đọc

}

# XMLDOMParser

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 "";

}

}

# LOAD VIDEO TỪ INTERNET

VideoView mVideoView;

mVideoView = (VideoView) findViewById(R.id.videoView);

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

mVideoView.setMediaController(new MediaController(MainActivity.this)); // Tạo bảng điều khiển

mVideoView.setVideoURI(uri); // Thực thi đường dẫn

mVideoView.start();

# PLAY MP3 TỪ INTERNET

#### //1. Hàm

public void PlayNhacMp3(String url){

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

MediaPlayer mediaPlayer = new MediaPlayer();

mediaPlayer.setAudioStreamType(AudioManager.STREAM\_MUSIC); // Trả lại dữ liệu kiểu audio trong android

try {

mediaPlayer.setDataSource(url); // Gán dữ liệu

mediaPlayer.prepareAsync(); // Duy trì liên tục, cần tạo thêm 1 tiến trình

mediaPlayer.setOnPreparedListener(new MediaPlayer.OnPreparedListener() {

@Override

public void onPrepared(MediaPlayer mp) {

mp.start();

}

});

} catch (IOException e) {

e.printStackTrace();

}

}

#### //2. Application

MediaPlayer mediaPlayer = new MediaPlayer();

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

mediaPlayer = new MediaPlayer();

mediaPlayer.setAudioStreamType(AudioManager.STREAM\_MUSIC);

try {

mediaPlayer.setDataSource(url);

} catch (IOException e) {

e.printStackTrace();

}

try {

mediaPlayer.prepare();

} catch (IOException e) {

e.printStackTrace();

}

### //Button Play & Pause

btnPlay.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

if (mediaPlayer.isPlaying()) {

mediaPlayer.pause();

} else {

mediaPlayer.start();

}

}

});

### // Button Stop

btnPause.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

mediaPlayer.stop();

mediaPlayer = new MediaPlayer();

mediaPlayer.setAudioStreamType(AudioManager.STREAM\_MUSIC);

try {

mediaPlayer.setDataSource(url);

} catch (IOException e) {

e.printStackTrace();

}

try {

mediaPlayer.prepare();

} catch (IOException e) {

e.printStackTrace();

}

}

});

# READ JSONObject

#### 1. Example:

{

"monhoc" : "Lập trình Android" ,

"noihoc" : "Trung Tâm Đào Tạo Tin Học Khoa Phạm"

}

#### 2.Read

JSONObject jsonObject = new JSONObject(s); // Do khai báo là JSON object : Nên nội dung nó tự bỏ ngoặc

// s: Nội dung đọc từ internet

String tenMon = jsonObject.getString("monhoc");

String noiHoc = jsonObject.getString("noihoc");

txtRead.setText(tenMon + "\n" + noiHoc + "\n" + webSite + "\n" + fanPage + "\n" + loGo);

# READ JSONArray

#### 1.Example:

{"danhsach": [

{"khoahoc" : "Lap trinh iOS co ban" },

{"khoahoc" : "Lap trinh iOS nang cao" },

{"khoahoc" : "Lap trinh Android" }

]}

#### 2.Read

JSONObject object = new JSONObject(s); // s: Nội dung đọc về

JSONArray array = object.getJSONArray("danhsach");

for (int i = 0 ; i < array.length();i++ ) {

JSONObject objectKH = array.getJSONObject(i);

String khoaHoc = objectKH.getString("khoahoc");

}

# SEVER PHP CHUYỂN DATA THÀNH JSON

<?php

class SanPham{

var $Id;

var $Tensp;

var $Giasp;

var $Mota;

function SanPham($Id, $Tensp, $Giasp, $Mota){

$this -> Id = $Id;

$this -> Tensp = $Tensp;

$this -> Giasp = $Giasp;

$this -> Mota = $Mota;

}

}

$arraySanPham = array();

// Đưa dữ liệu từ database ra

$connect = mysqli\_connect("localhost","root","","dienthoai");

mysqli\_query($connect, "SET NAMES 'utf8'"); // Lệnh đặt tiếng việt

$query = "SELECT \* FROM sanpham"; // Lệnh lấy all dữ liệu ra

$data = mysqli\_query($connect, $query);

while ($row = mysqli\_fetch\_assoc($data)) { // Tách từng hàng ra từ $data

array\_push($arraySanPham, new SanPham($row['id'], $row['tensp'], $row['giasp'], $row['mota']));

}

echo json\_encode($arraySanPham); // In ra web

?>

# HÀM POST DATA LÊN SERVER BẰNG THƯ VIỆN VOLLEY

RequestQueue requestQueue1 = Volley.newRequestQueue(this);

StringRequest stringRequestDelete = new StringRequest(Request.Method.POST, linkDelete,

new Response.Listener<String>() {

@Override

public void onResponse(String response) {

if (response.equals("1")) {

ShowListView();

Toast.makeText(MainActivity.this, "Deleted: " + arraySanPham.get(index).getTen(), Toast.LENGTH\_SHORT).show();

} else {

Toast.makeText(MainActivity.this, "Delete Fail!", Toast.LENGTH\_SHORT).show();

}

}

},

new Response.ErrorListener() {

@Override

public void fonErrorResponse(VolleyError error) {

Toast.makeText(MainActivity.this, error.toString(), Toast.LENGTH\_SHORT).show();

}

}){

@Override

protected Map<String, String> getParams() throws AuthFailureError {

Map<String, String> params = new HashMap<String, String>();

params.put("Id", String.valueOf(arraySanPham.get(index).getId()));

return params;

}

};

requestQueue1.add(stringRequestDelete);

# HÀM POST DATA LÊN SERVER (DÙNG CHO ASYNCTASK)

private String postData(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();

}

}

# VIDEO YOUTUBE

1. public class MainActivity extends YouTubeBaseActivity

implements YouTubePlayer.OnInitializedListener{...}

2. btnPlay.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

youTubePlayerView.initialize(KEY\_API, MainActivity.this);

}

});

3.

### // Kết nối thành công

@Override

public void onInitializationSuccess(YouTubePlayer.Provider provider, YouTubePlayer youTubePlayer, boolean b) {

youTubePlayer.cueVideo(edtID.getText().toString().trim()); // Phát video nào? Đưa vào ID của Video trên Youtube

}

### // Kết nối thất bại

@Override

public void onInitializationFailure(YouTubePlayer.Provider provider, YouTubeInitializationResult youTubeInitializationResult) {

### // Lỗi người dùng (Mạng)

### // Lỗi do video

if (youTubeInitializationResult.isUserRecoverableError()) { // Trả về true nếu là lỗi người dùng

### // Thông báo lỗi

youTubeInitializationResult.getErrorDialog(MainActivity.this, REQUEST\_CODE\_VIDEO);

} else { // Do video bị lỗi

Toast.makeText(this, "Video Error!", Toast.LENGTH\_SHORT).show();

}

}

@Override

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

if (requestCode == REQUEST\_CODE\_VIDEO) {

### // -> Thử khởi tạo lại video!

youTubePlayerView.initialize(KEY\_API, MainActivity.this); // Key API, Do đã truyền ở trên rồi

}

super.onActivityResult(requestCode, resultCode, data);

}

# PLAY VIDEO YOUTUBE

1. Tạo project trên Google Console

2. Add Dịch vụ: Youtube Data API

3. Lấy Key

4. Nhúng KEY\_API vào App (Sử dụng luôn giao diện VideoView của Youtube) bằng cách add thư viện của nó vào

5. Tìm thư viện Control VideoView Youtube: DOWNLOAD YOUTUBE ANDROID PLAYER API

# PLAY LISTVIDEO YOUTUBE

1. Tạo project trên Google Console

2. Add Dịch vụ: - Youtube Data API

- Google+ API

- Contacts API

3. Lấy Key

4. Nhúng KEY\_API vào App (Sử dụng luôn giao diện VideoView của Youtube) bằng cách add thư viện của nó vào

5. Lấy JSON listVideo của Youtube:

https://www.googleapis.com/youtube/v3/playlistItems?part=snippet&playlistId={IdListVideo}&key={YOUR\_API\_KEY}

6. maxResults = "", Để lấy số lượng max video ra.

# SEARCH VIEW ACTIONBAR, TOOLBAR

#### 1. Tạo layout menu

<menu 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"

tools:context=".HomeActivity">

<item

android:id="@+id/action\_search"

android:icon="@android:drawable/ic\_menu\_search"

android:title="Search"

app:actionViewClass="android.support.v7.widget.SearchView"

app:showAsAction="always" />

</menu>

#### 2. MainActivity

SearchView searchView;

@Override

public boolean onCreateOptionsMenu(Menu menu) {

MenuInflater inflater = getMenuInflater();

inflater.inflate(R.menu.menu\_searchview, menu);

MenuItem searchItem = menu.findItem(R.id.action\_search);

SearchManager searchManager = (SearchManager) getSystemService(Context.SEARCH\_SERVICE);

if (searchItem != null) {

searchView = (SearchView) searchItem.getActionView();

}

if (searchView != null) {

searchView.setSearchableInfo(searchManager.getSearchableInfo(getComponentName()));

searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {

@Override

public boolean onQueryTextSubmit(String query) {

Toast.makeText(MainActivity.this, query.toString(), Toast.LENGTH\_SHORT).show();

return false;

}

@Override

public boolean onQueryTextChange(String newText) {

Log.d("text", newText.toString());

return false;

}

});

}

return super.onCreateOptionsMenu(menu);

}

#### 3. Hàm Filter trong Adapter của ListView

### // Filter Class

public void filter(String charText) {

charText = charText.toLowerCase(Locale.getDefault()); // Đưa về chữ thường để search

videoList.clear(); // Xóa dữ liệu trong mảng

if (charText.length() == 0) {

videoList.addAll(arrayList); // Nếu chưa nhập gì thì lấy kết quả là nguyên gốc đã sao

} else {

for (CVideo video : arrayList) { // Duyệt hết mảng thứ 2

if (video.getName().toLowerCase(Locale.getDefault()) // Tìm theo thuộc tính nào > ở đây tìm theo Title// Chuyển dữ liệu trong Name thành chữ thường

.contains(charText)) {

videoList.add(video); // Add thêm vào mảng để đưa lên Listview

}

}

}

notifyDataSetChanged();

}

# FRAGMENT

# Sent Data from Activity to Fragment

#### 1. Activity:

fragment\_a fragment\_a = (fragment\_a) getFragmentManager().findFragmentById(R.id.fragment\_a);

Bundle bundle = new Bundle();

bundle.putString("key", "Vu Quoc Thanh");

fragment\_a.setArguments(bundle);

#### 2. Fragment

Bundle bundle = getArguments();

String name = bundle.getString("key");

# Sent Data from Fragment to Activity

#### 1. Create: Interface sent data to activity

public interface SentStudent {

public void SendData(CStudent student); // Có thể truyền vào String, ... Đối tượng này sẽ được chuyển sang Activity implement nó

}

#### 2. Fragment

SentStudent sentStudent; // Gọi interface

public void onActivityCreated(Bundle savedInstanceState) {

super.onActivityCreated(savedInstanceState);

sentStudent = (SentStudent) getActivity(); // Constructor

...

}

// Sent data

sentStudent.SendData(arrarStudent.get(position)); // Gọi phương thức SenData của interface, truyền dữ liệu đi

#### 3. Activity

### // Implement methord for activity

public class MainActivity extends AppCompatActivity implements SentStudent{

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

....

}

### // Override SenData Để nhận dữ liệu

@Override

public void SendData(CStudent student) {

### // Data được lưu trong biến student

fragment\_infor fragment\_infor = (fragment\_infor) getFragmentManager().findFragmentById(R.id.fragmentInfor);

### // Check Oriontasion of device

Configuration configuration = getResources().getConfiguration();

if (fragment\_infor != null && configuration.orientation == Configuration.ORIENTATION\_LANDSCAPE) {

// Nếu màn hình nằm ngang thì mới vô If và truyền dữ liệu qua fragment infor

// Or: && fragment\_infor.isInLayout()

// Try catch

fragment\_infor.SetValues(student);

} else {

Intent intent = new Intent(MainActivity.this, DetailActivity.class);

intent.putExtra("student", student); // Truyền tiếp qua fragment tiếp theo

startActivity(intent);

}

}

# Truyền ArrayObject từ Activity sang Fragment

#### 1. implement Parcelable vào Class của đối tượng cần chuyển đi

public class MyParcelable implements Parcelable {

private int mData; // Dữ liệu có thể nhiều hơn

public int describeContents() {

return 0;

}

public void writeToParcel(Parcel out, int flags) {

out.writeInt(mData);

}

public static final Parcelable.Creator<MyParcelable> CREATOR

= new Parcelable.Creator<MyParcelable>() {

public MyParcelable createFromParcel(Parcel in) {

return new MyParcelable(in);

}

public MyParcelable[] newArray(int size) {

return new MyParcelable[size];

}

};

private MyParcelable(Parcel in) {

mData = in.readInt();

}

#### 2. Truyền bên Activity

ArrayList<MyClass> data = new ArrayList<MyClass>();

............

Bundle extras1 = new Bundle();

extras1.putParcelableArrayList("arraylist", data);

Tab1Fragment fg = new Tab1Fragment();

fg.setArguments(extras1);

#### 3. Nhận bên Fragment

Bundle extras = getArguments();

ListView list = (ListView) content.findViewById(R.id.lvMain);

if (extras != null) {

data = extras.getParcelableArrayList("arraylist");

list.setAdapter(new MyAdapter(getActivity(), data));

}

# NAVIGATION VIEW

1. Tạo layout phía trên (Chinh chieu cap). Giống Layout bình thường. tùy ý chỉnh

2. Activity\_main.xml : Add DrawerLayout set id (Delete RelativeLayout...)

Trong Drawerlayout add Thư viện Design -> Add Navigation

+ Add Thưu Viện: File -> ProjectStructer -> App -> Dependence ->Add Design

Bỏ ActionBar -> Thêm ToolBar (Trong layout va Java)

3. Add Toolbar.. Làm việc trên toolbar

setDispalyHomeAsUpEnalble(true)

Activity\_main:

<?xml version="1.0" encoding="utf-8"?>

<android.support.v4.widget.DrawerLayout

android:layout\_height="match\_parent"

android:layout\_width="match\_parent"

xmlns:app="http://schemas.android.com/apk/res-auto"

android:id="@+id/myDrawerLayout"

xmlns:android="http://schemas.android.com/apk/res/android">

<LinearLayout

android:orientation="vertical"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<!--Dung hang moi, -->

<!--?attr/actionBarSize: Lấy thuộc tính của Android-->

<android.support.v7.widget.Toolbar

android:layout\_width="match\_parent"

android:id="@+id/toolBar"

android:layout\_height="?attr/actionBarSize"

android:background="#ff00">

</android.support.v7.widget.Toolbar>

</LinearLayout>

<!--Add Navigation -->

<!--Add Design Library-->

<android.support.design.widget.NavigationView

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:id="@+id/myNavigation"

app:headerLayout="@layout/header\_navigation"

android:layout\_gravity = "start"

app:menu="@menu/menu\_navigation">

</android.support.design.widget.NavigationView>

</android.support.v4.widget.DrawerLayout>

4. MainActivity

# RECYLER VIEW

#### 1. Tạo Layout cho mỗi item : giống Listview

#### 2. Tạo Adapter

public class ShopAdapter extends RecyclerView.Adapter<ShopAdapter.ViewHolder>{

private Context context;

private List<CShop> shopList;

public ShopAdapter(Context context, List<CShop> shopList) { // Constructor

this.context = context;

this.shopList = shopList;

}

// Xóa Item

public void RemoveItem(int position) { // Function Remove : test Effect

shopList.remove(position); // Remove in Array

notifyItemRemoved(position); // Commit Remove - Reload RecyclerView

}

@Override

public ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {

LayoutInflater inflater = LayoutInflater.from(parent.getContext()); // Đưa lên giao diện

View v = inflater.inflate(R.layout.line\_produce, null);

return new ViewHolder(v);

}

@Override

public void onBindViewHolder(ViewHolder holder, int position) {

holder.txtName.setText(shopList.get(position).getName()); // Set Values for itemView

holder.imgIcon.setImageResource(shopList.get(position).getIcon());

}

@Override

public int getItemCount() {

return shopList.size();

}

public class ViewHolder extends RecyclerView.ViewHolder {

TextView txtName; // Ánh xạ

ImageView imgIcon;

public ViewHolder(final View itemView) {

super(itemView);

txtName = (TextView) itemView.findViewById(R.id.textViewName); // Constructor

imgIcon = (ImageView) itemView.findViewById(R.id.imageViewPicture);

itemView.setOnClickListener(new View.OnClickListener() { // Listener Remove itemView

@Override

public void onClick(View view) {

RemoveItem(getAdapterPosition());

Toast.makeText(context, "Remove: " + txtName.getText(), Toast.LENGTH\_SHORT).show();

}

});

}

}

}

#### 3. Kết nối lên Avtivity

public void initView() {

recyclerView.setHasFixedSize(true);// Tối ưu size

LinearLayoutManager linearLayoutManager = new LinearLayoutManager(this, LinearLayoutManager.HORIZONTAL, false); // Định dạng layout của RecyclerView

recyclerView.setLayoutManager(linearLayoutManager); // Gắn layout vào RecyclerView

DividerItemDecoration dividerItemDecoration = new DividerItemDecoration(recyclerView.getContext(), DividerItemDecoration.HORIZONTAL); // Chỉnh Divider cho itemView của RecyclerView

Drawable drawable = ContextCompat.getDrawable(getApplicationContext(), R.drawable.ct\_divider); // Custom Divider

dividerItemDecoration.setDrawable(drawable);

recyclerView.addItemDecoration(dividerItemDecoration); // Add effect default remove, Add , Update......

recyclerView.setItemAnimator(new DefaultItemAnimator()); //

adapterShop = new ShopAdapter(getApplicationContext(), arrayShop); // Theo thông sô bên kia nhé! Tùy vào hàm constructor

recyclerView.setAdapter(adapterShop);

}

# ONLISTENER RecyclerView

#### 1. Thêm Class RecyclerItemClickListener:

import android.content.Context;

import android.support.v7.widget.RecyclerView;

import android.view.GestureDetector;

import android.view.MotionEvent;

import android.view.View;

public class RecyclerItemClickListener implements RecyclerView.OnItemTouchListener {

private OnItemClickListener mListener;

public interface OnItemClickListener {

public void onItemClick(View view, int position);

public void onLongItemClick(View view, int position);

}

GestureDetector mGestureDetector;

public RecyclerItemClickListener(Context context, final RecyclerView recyclerView, OnItemClickListener listener) {

mListener = listener;

mGestureDetector = new GestureDetector(context, new GestureDetector.SimpleOnGestureListener() {

@Override

public boolean onSingleTapUp(MotionEvent e) {

return true;

}

@Override

public void onLongPress(MotionEvent e) {

View child = recyclerView.findChildViewUnder(e.getX(), e.getY());

if (child != null && mListener != null) {

mListener.onLongItemClick(child, recyclerView.getChildAdapterPosition(child));

}

}

});

}

@Override public boolean onInterceptTouchEvent(RecyclerView view, MotionEvent e) {

View childView = view.findChildViewUnder(e.getX(), e.getY());

if (childView != null && mListener != null && mGestureDetector.onTouchEvent(e)) {

mListener.onItemClick(childView, view.getChildAdapterPosition(childView));

return true;

}

return false;

}

@Override public void onTouchEvent(RecyclerView view, MotionEvent motionEvent) { }

@Override

public void onRequestDisallowInterceptTouchEvent (boolean disallowIntercept){}

}

#### 2. Bắt sự kiện bên phía Activity

recyclerViewHL.addOnItemTouchListener(

new RecyclerItemClickListener(getApplicationContext(), recyclerViewHL ,new RecyclerItemClickListener.OnItemClickListener() {

@Override public void onItemClick(View view, int position) {

Toast.makeText(UserActivity.this, "chi so:" + position, Toast.LENGTH\_SHORT).show();

}

@Override public void onLongItemClick(View view, int position) {

Toast.makeText(UserActivity.this, "Chi so long:" + position, Toast.LENGTH\_SHORT).show();

}

})

);

# ADMOD FIREBASE

MobileAds.initialize(getApplicationContext(), "ca-app-pub-6317808399413254~2123658523");

AdView mAdView = (AdView) findViewById(R.id.adViewChangePass);

AdRequest adRequest = new AdRequest.Builder().build();

mAdView.loadAd(adRequest);