

**Một số phím tắt thường dùng**

CTRL + ALT + L	format code
ALT + insert	Generate method
CTRL + Y	Delete line
CTRL + D	Copy line
CTRL + F9	buld App
SHIFT + F10	buld and run App
ALT + left or right key	Change tab
CTRL + /	Tạo hoặc xóa comment

**Tài liệu tham khảo:**

<https://developer.android.com/studio/intro>

<https://developer.android.com/studio/intro>

## BÀI THỰC HÀNH SỐ 1

Tạo dự án Android với công cụ Android Studio and SDK Tools.

### 1. Mục tiêu

- Biết cách khởi tạo dự án Android
- Biết cách khởi chạy thiết bị ảo
- Biết tạo giao diện một số view đơn giản
- Biết cách lập trình xử lý sự kiện đơn giản

### 2. Nội dung:

#### Bài 1.

Cài đặt AndroidStudio, cài đặt môi trường, cài đặt và tạo máy ảo chạy nền tảng android.

#### Hướng dẫn:

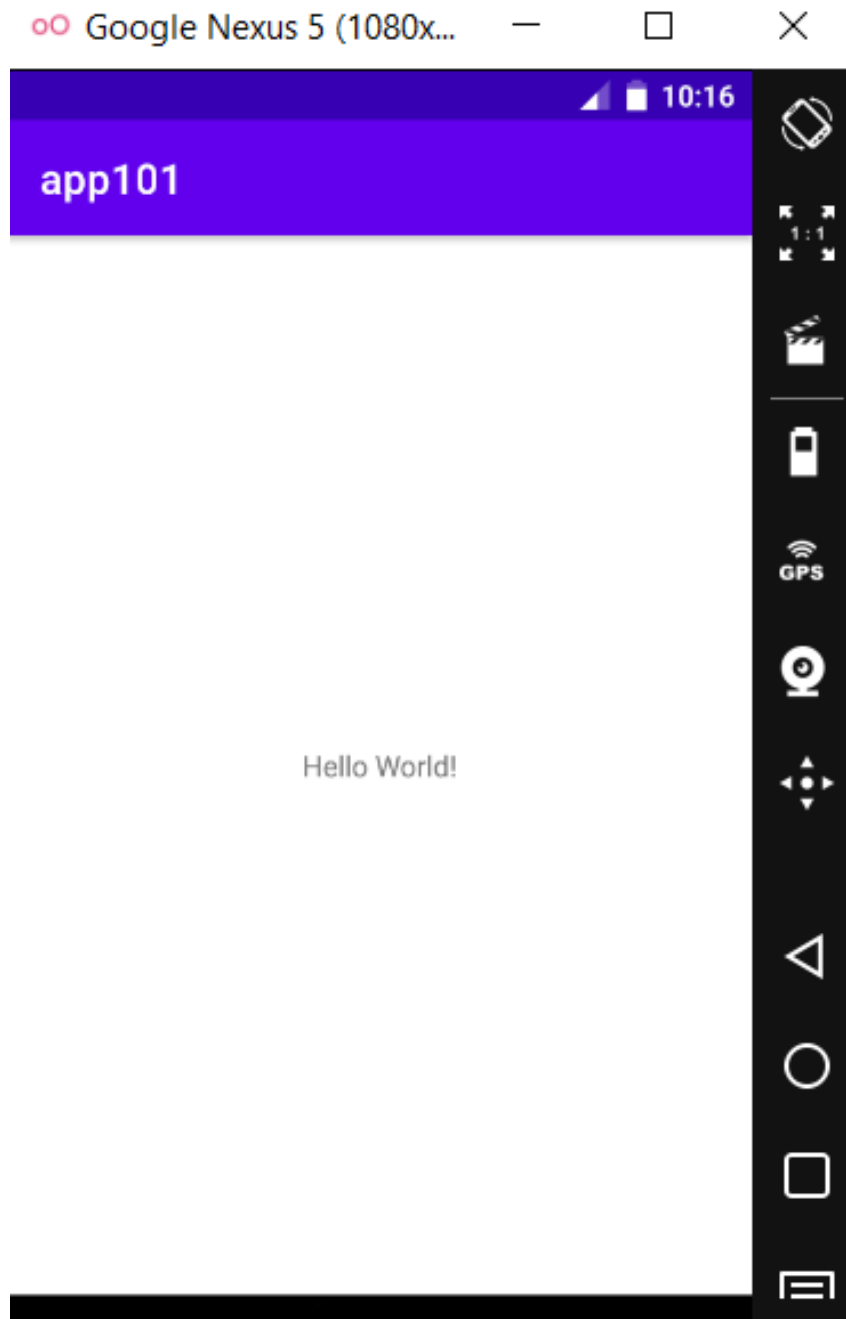
Tải các bộ cài đặt tại các địa chỉ:

<https://drive.google.com/drive/folders/1-R-rLeBaI87uT31p4KfW-ZGMDHCtwj?usp=sharing>

Xem video hướng dẫn cài đặt và tạo biến môi trường

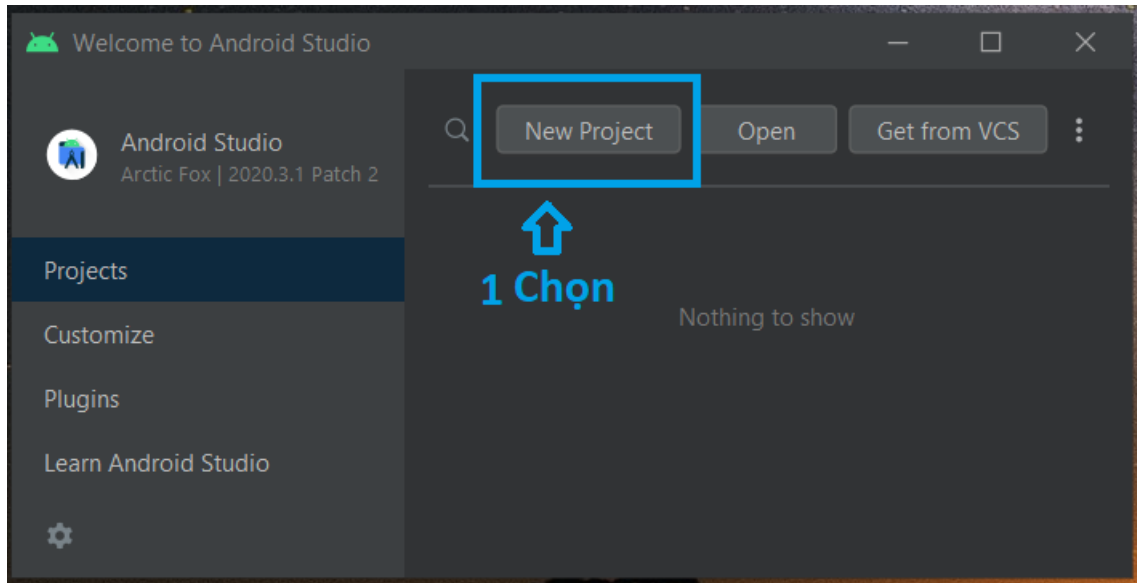
#### Bài 2.

Tạo project bằng Android Studio, chạy chương trình bằng máy ảo hiển thị dòng chữ "**Hello World**".

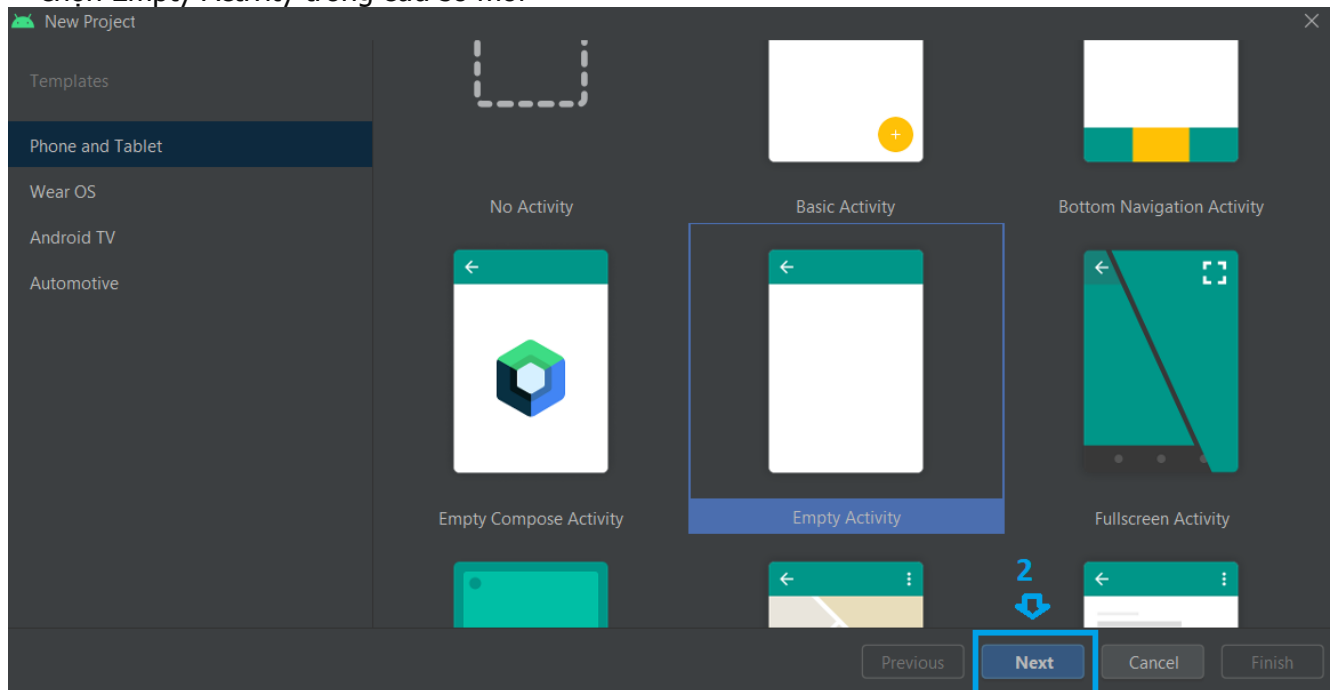


**Hướng dẫn:**

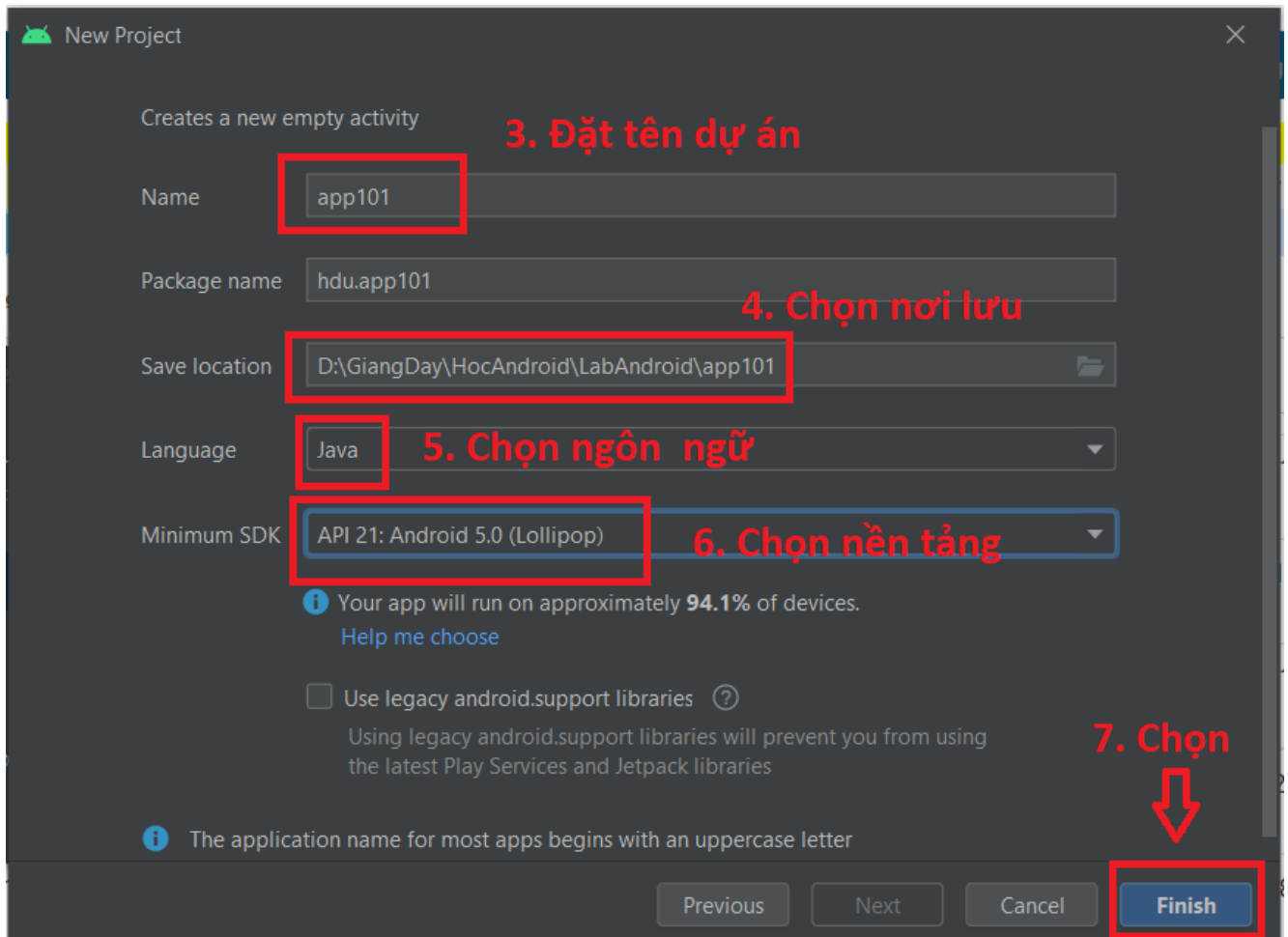
Bước 1. Tạo project: Khởi động android Studio, chọn New Project



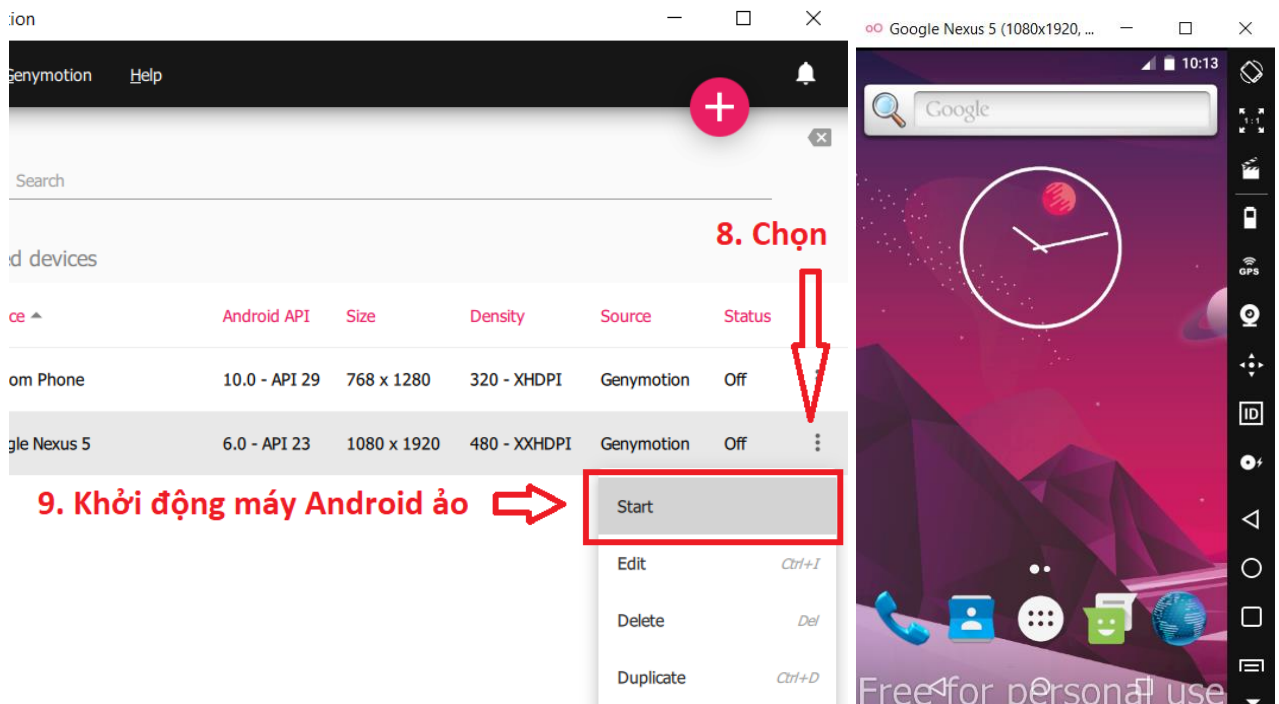
=> chọn Empty Activity trong cửa sổ mới



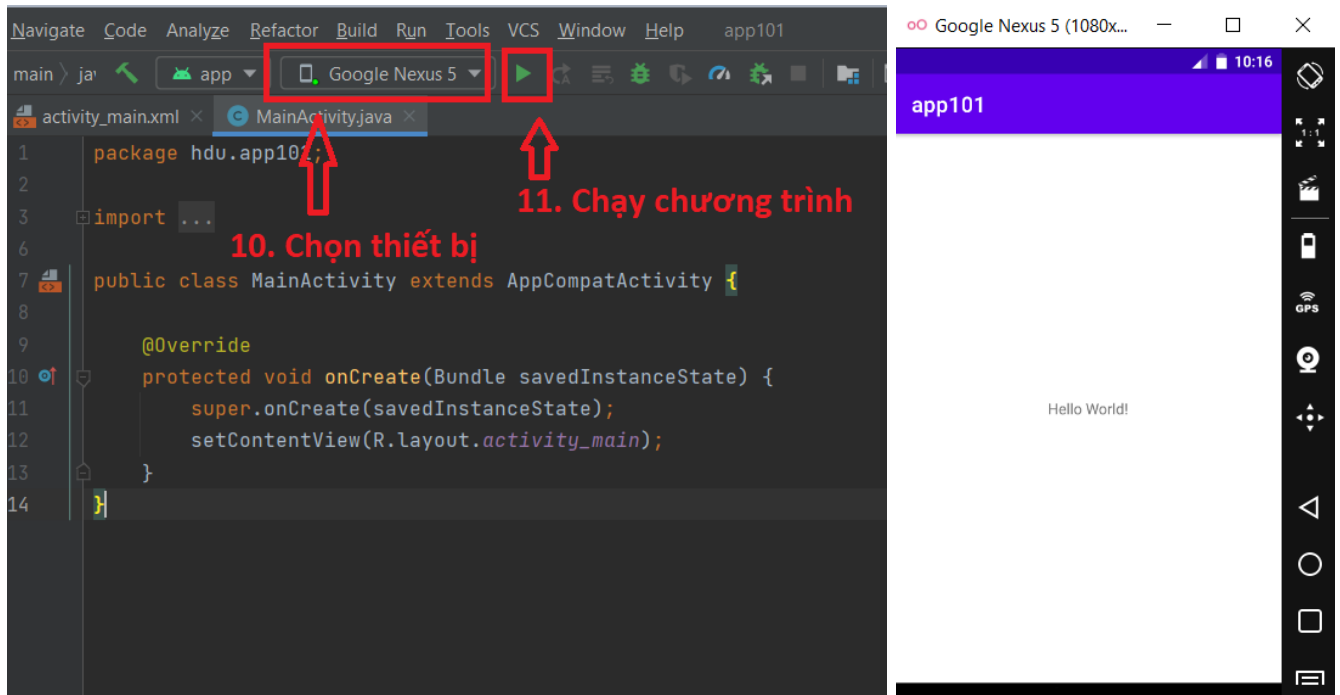
=> Đặt tên dự án, chọn nơi lưu, chọn nền tảng thiết bị Android



Bước 2. Khởi động chương trình máy ảo **Genymotion**, khởi động máy android ảo.



Bước 3. Biên dịch chương trình, chọn máy android ảo và chạy chương trình bằng máy android ảo



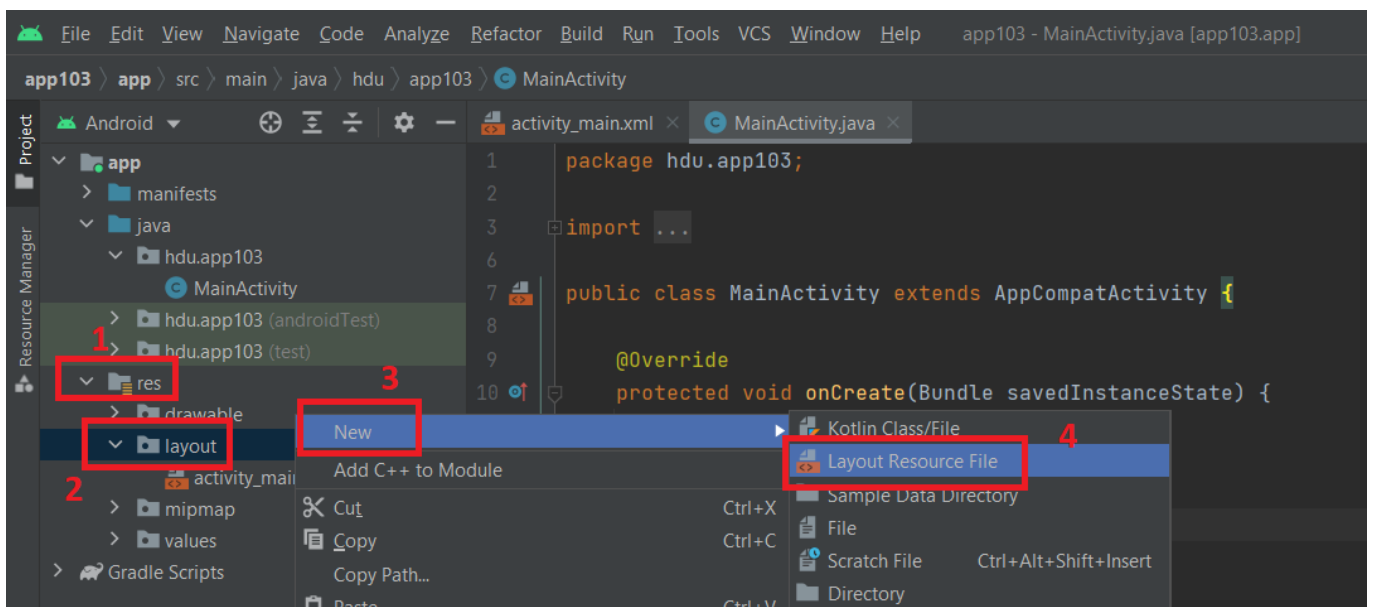
### Bài 3.

Tạo project bằng Android Studio, thêm mới **layout Activity**, có chứa text "**Hello world**". Khai báo để **layout Activity** mới thêm hoạt động khi chạy chương trình.

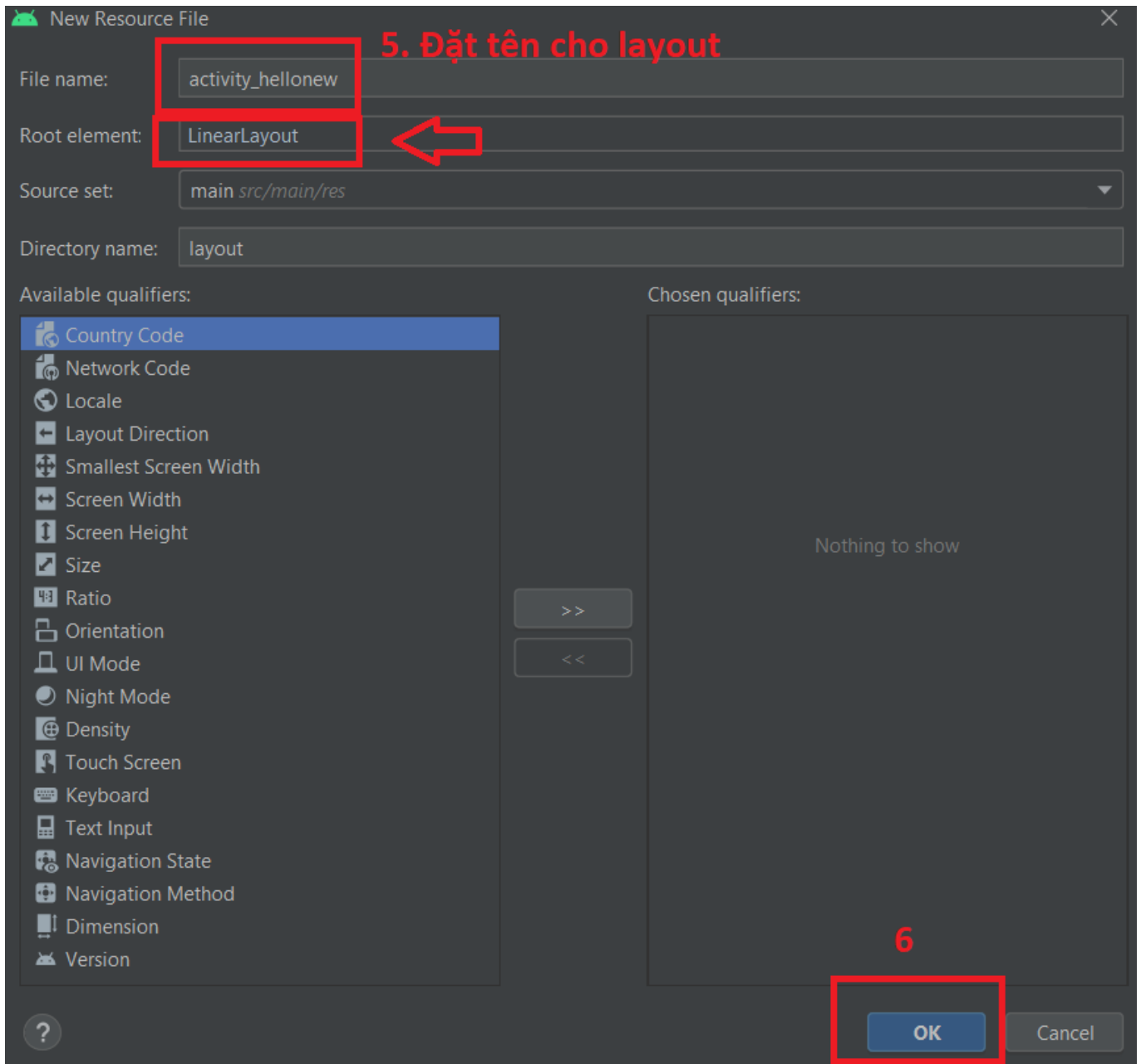
Hướng dẫn:

Bước 1. Tạo project bằng Android Studio như hướng dẫn bước 1 và bước 2 của bài 1

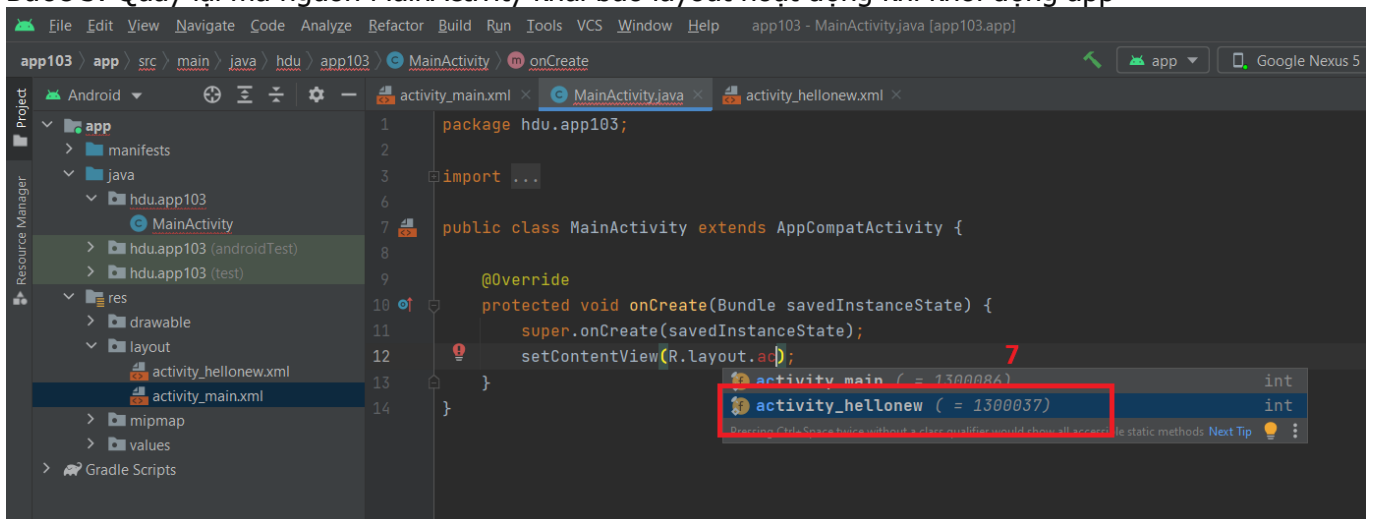
Bước 2. Click Chuột phải vào thư mục **layout** trong **res** chọn **New => Layout Resource File**



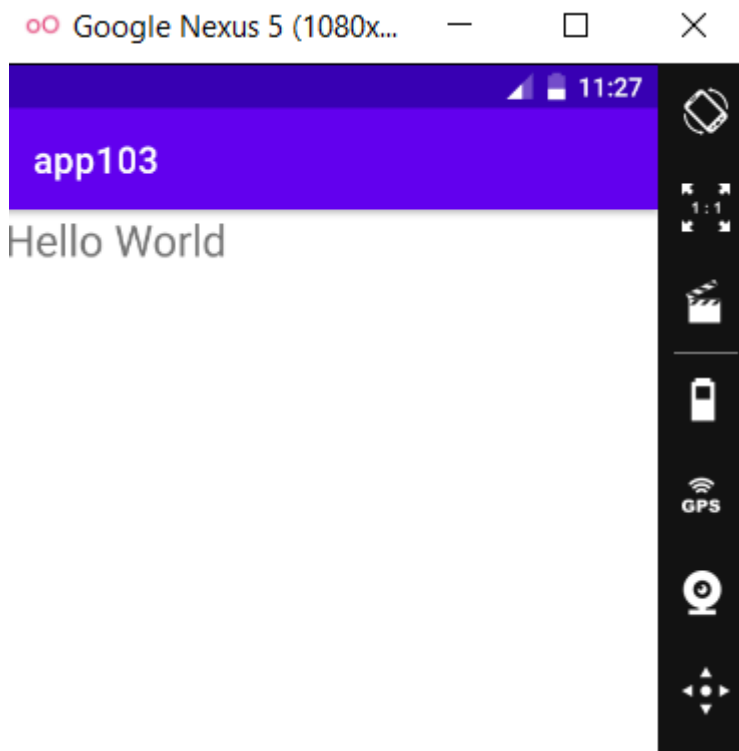
Trong cửa sổ mới đặt tên layout, lưu ý chọn Root element: **LinearLayout**



Bước 3: Quay lại mã nguồn MainActivity khai báo layout hoạt động khi khởi động app



Bước 4: Biên dịch và chạy chương trình hiển thị kết quả



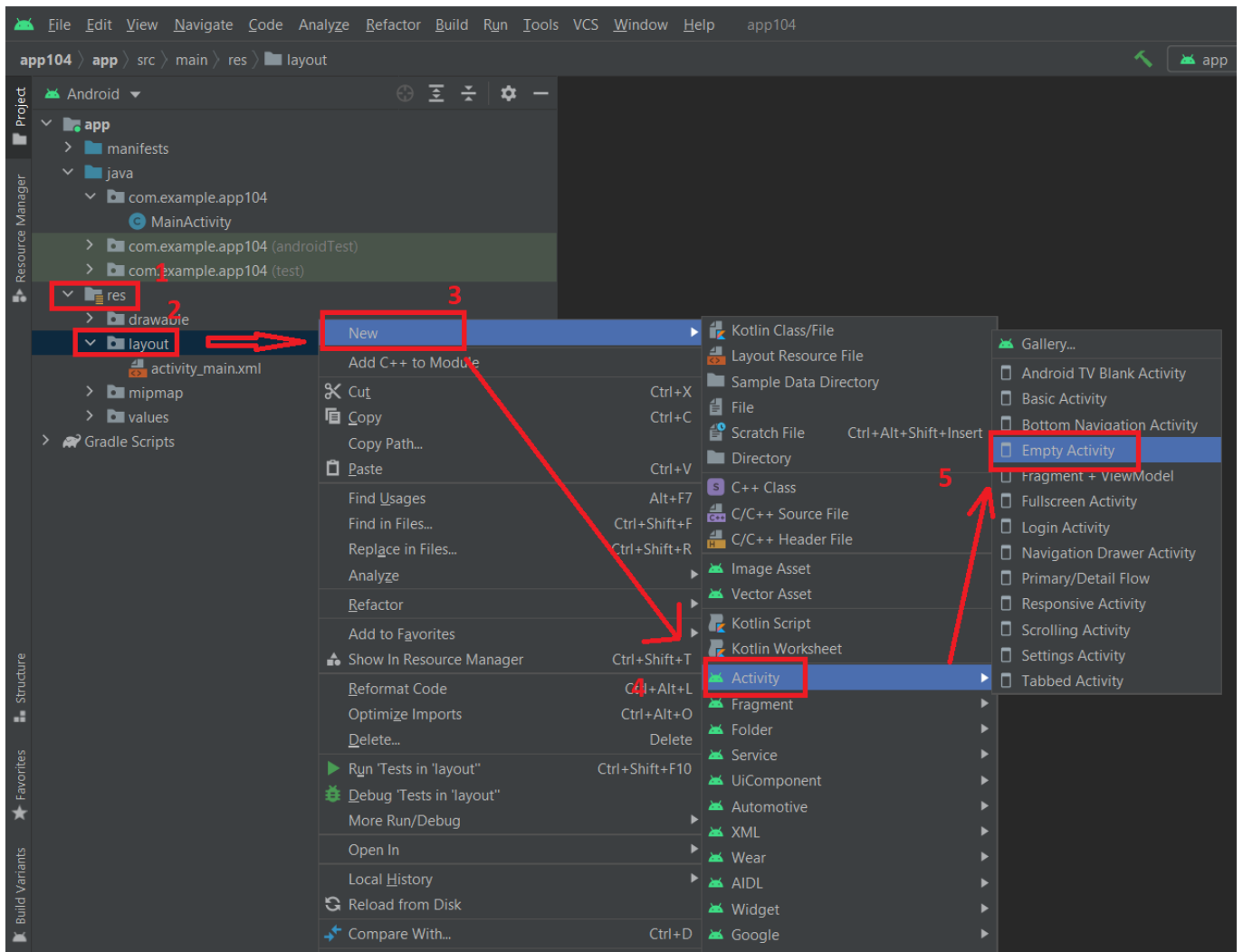
#### Bài 4.

Tạo project bằng Android Studio, thêm mới một **Empty Activity** có chứa một số điều khiển cơ bản như textview, button...định vị các điều khiển trên màn hình, Khai báo để activity **mới thêm** hoạt động khi chạy chương trình lần đầu tiên.

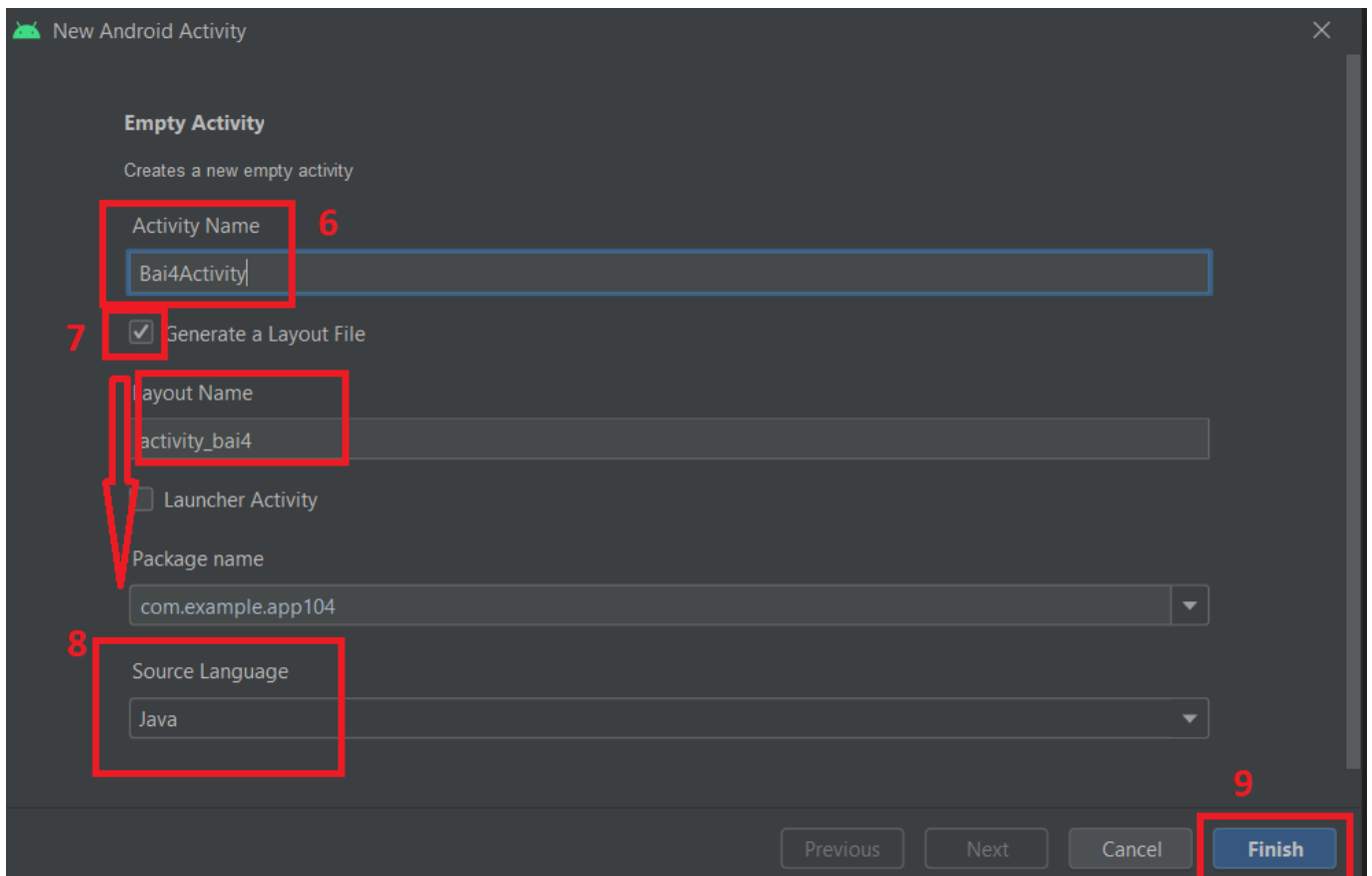
Hướng dẫn:

Bước 1. Tạo **project** bằng Android Studio và khởi động máy ảo **Genymotion** (như hướng dẫn bước 1 và bước 2 của bài 1)

Bước 2. Click Chuột phải vào thư mục **layout** trong **res** chọn **New** => **Activity** => **Empty Activity**



Trong cửa sổ mới thực hiện các bước như hình 4.2 đặt tên cho activity mới (6) ....

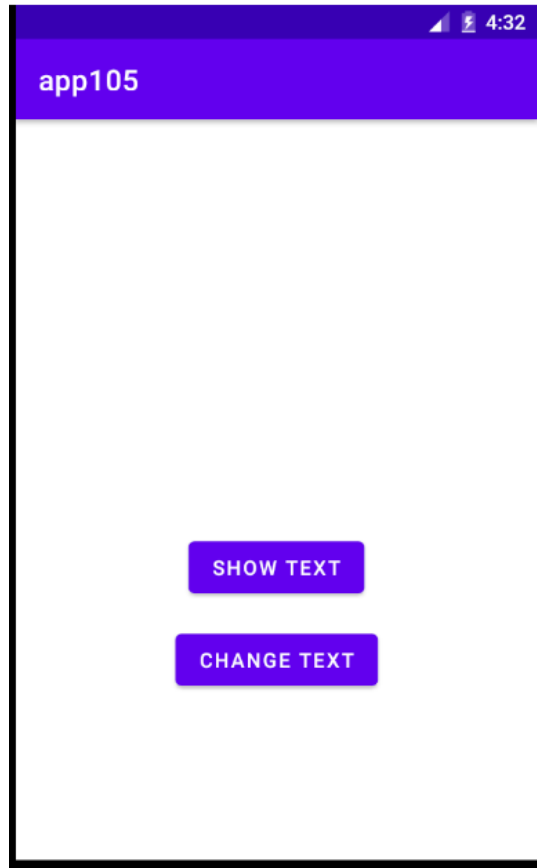




Thực hiện chạy chương trình và xem kết quả

### Bài 5.

Tạo project bằng Android Studio có giao diện như hình 5.1



Hình 5.1

Thực hiện các yêu cầu sau

- Khi bấm vào button **Show Text** sẽ hiển thị dòng chữ hiển thị dòng chữ "**Hello World**" như hình 5.2, đồng thời **LogCat** của Android Studio hiển thị dòng chữ "**Hello World**" trong cửa sổ logcat

Hello World!

SHOW TEXT

CHANGE TEXT

- Khi bấm vào button **Change Text** sẽ hiển thị dòng chữ hiển thị dòng chữ "**Khoa CNTT&TT**" thay thế text "**Hello World**" đồng thời thay thế trong logcat.

Hướng dẫn:

Bước 1. Tạo **project** bằng Android Studio và khởi động máy ảo **Genymotion** (như hướng dẫn bước 1 và bước 2 của bài 1)

Bước 2. Mở tệp **activity\_main** trong **layout** của **res** để ở chế độ xem Design thêm 2 button và đặt tên như yêu cầu

Bước 3: xử lý sự kiện click chuột cho 2 nút bấm: tham khảo mã nguồn đặt trong **MainActivity** như sau

Lưu ý: Đọc và tìm hiểu thêm một số sự kiện Log cơ bản:

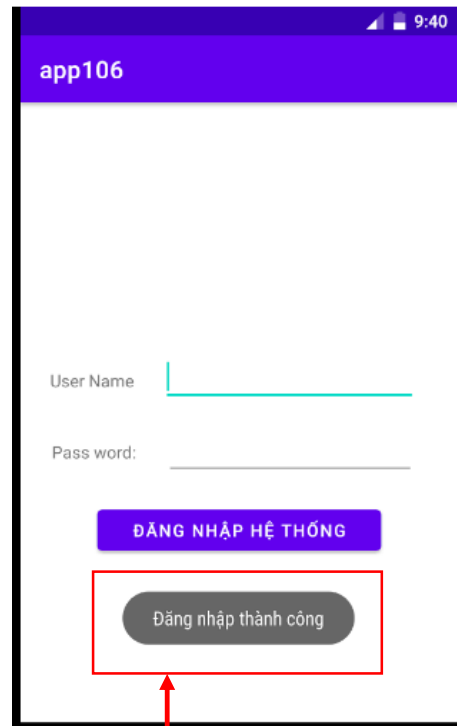
- Log.e(String, String) (error)
- Log.w(String, String) (warning)
- Log.i(String, String) (information)
- Log.d(String, String) (debug)
- Log.v(String, String) (verbose)

## Bài 6.

Tạo project bằng Android Studio có giao diện như hình 6.1, thực hiện các yêu cầu sau:



Hình 6.1



Hình 6.2

khi bấm vào button **Login** => **Toast** sẽ hiển thị dòng chữ "**Đăng nhập thành công.**" Tại máy android chạy project **như hình 6.2.**

**Hướng dẫn:** Bắt sự kiện click của button trong MainActivity (tham khảo mã nguồn)

```
// hiển thị Toast thông báo "Đăng nhập thành công"
final Button button = findViewById(R.id.btDangNhap);
button.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        Toast.makeText(context: MainActivity.this, text: "Đăng nhập thành công",
            Toast.LENGTH_LONG).show();
    }
});
```

Hoặc cách khác:

- Thêm onclick trong tệp Activity\_layout sử dụng hàm "ShowMessenger".

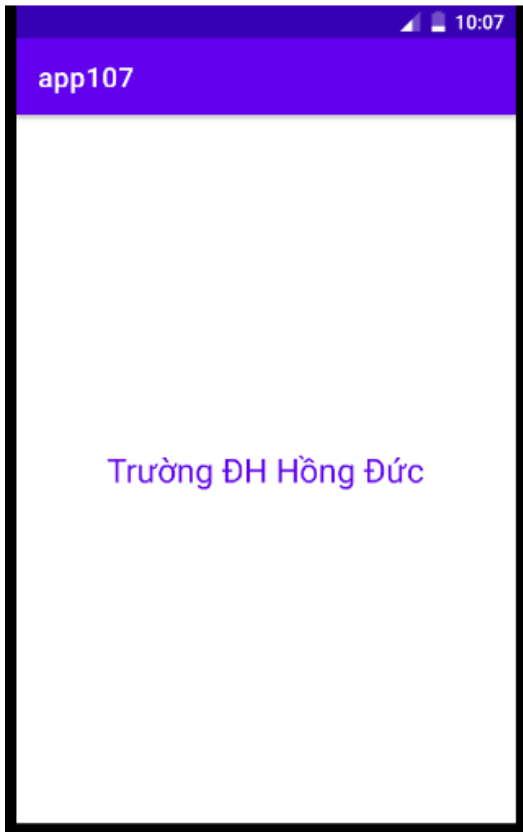
```
<Button
    android:onClick="ShowMessenger"
```

- Thêm hàm "ShowMessenger" trong MainActivity:

```
public void ShowMessenger(View v) {
    Toast.makeText(context: MainActivity.this, text: "Đăng nhập thành công",
        Toast.LENGTH_LONG).show();
    //ShowToast("Đăng nhập thành công");
}
```

## Bài 7.

Tạo project bằng Android Studio có giao diện khi chạy chương trình hiển thị dòng chữ "**Trường ĐH Hồng Đức**" (như hình 7.1). Sử dụng **LogCat** của Android Studio hiển thị thông tin của một số hàm thể hiện hoạt động (Vòng đời) của 1 ứng dụng android trên thiết bị di động.



Hình 7.1



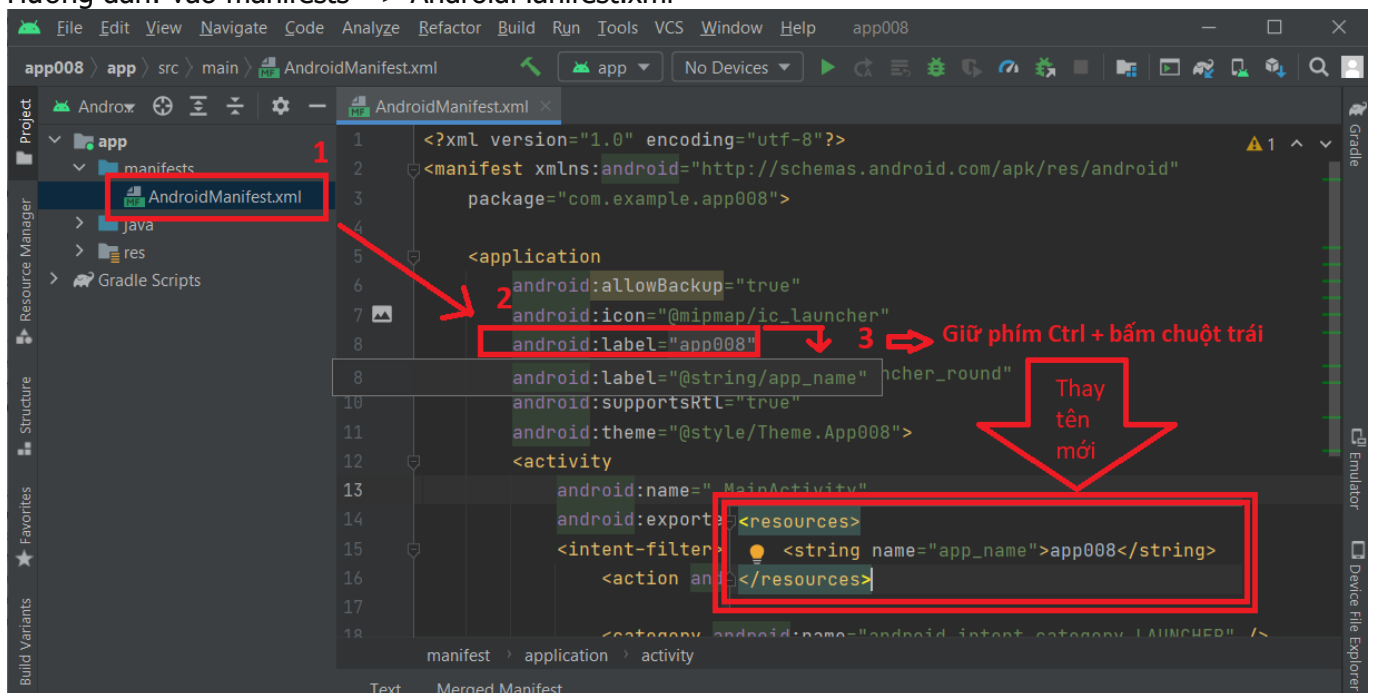
Hình 7.2

Hướng dẫn: Sử dụng một số phương thức thể hiện hoạt động (vòng đời) của 1 ứng dụng (tham khảo mã nguồn như hình 7.2)

### Bài 8.

Tạo project và chạy chương trình hiển thị dòng chữ "**Lập trình ứng dụng di động**". Thay đổi tên hiển thị của ứng dụng, thay đổi biểu tượng (Icon) hiển thị của ứng dụng trên thiết bị di động.

Hướng dẫn: vào manifests => AndroidManifest.xml



Lưu ý: tên App dựa vào Activity chạy chính để hiển thị

### Bài 9.

Thao tác chức năng Import và Export project Android Studio

## BÀI THỰC HÀNH SỐ 2

Xây dựng ứng dụng Android với các điều khiển cơ bản

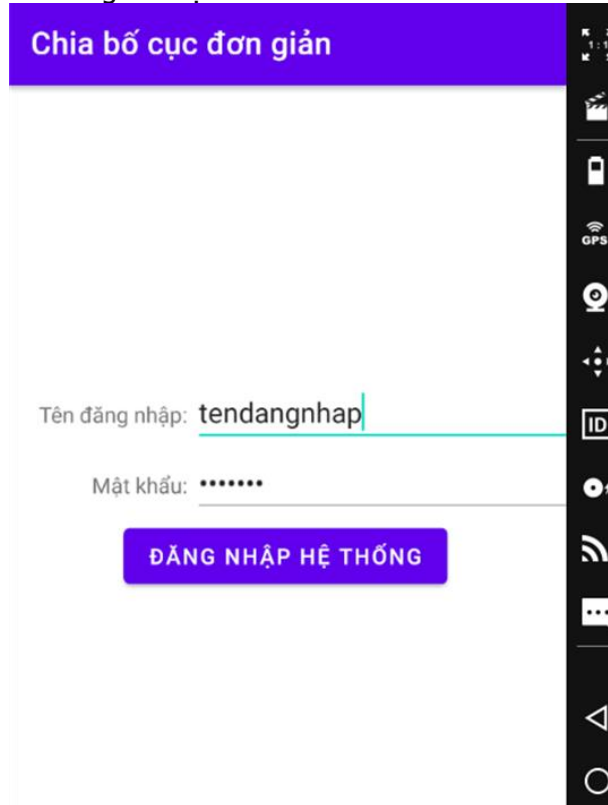
### 1. Mục tiêu

- Biết cách khởi tạo dự án Android
- Biết cách khởi chạy thiết bị ảo
- Biết cách sử dụng các điều khiển cơ bản (textView, button, checkbox, radiobutton, Image,...) kết hợp với các layout (Table Layout, Linear Layout, RelativeLayout ... ) cơ bản để xây dựng các view đơn giản chạy trên nền tảng android
- Biết cách lập trình xử lý sự kiện đơn giản

### 2. Nội dung:

#### Bài 1.

Tạo project bằng Android Studio có giao diện như sau:



Yêu cầu: khi click vào button Đăng nhập hệ thống => hiển thị thông báo **Toast** đăng nhập thành công.

#### Hướng dẫn:

Tạo project, Add thêm activity layout LinearLayout vào phần layout, tạo các linear layout phân chia và thêm các control tương ứng.

Tham khảo mã nguồn

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center">
    // phần nhập username
    <LinearLayout...>
    // phần nhập pass
    <LinearLayout...>
    // phần button đăng nhập
    <LinearLayout...>
</LinearLayout>
```

Phần nhập username (tương tự cho nhập pass)

```
// phần nhập username
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
    <TextView
        android:id="@+id/usn"
        android:layout_width="120dp"
        android:layout_height="wrap_content"
        android:text="Tên đăng nhập: "
        android:textAlignment="textEnd" />
    <EditText
        android:id="@+id/edituser"
        android:layout_width="260dp"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName" />
</LinearLayout>
```

Phần button Đăng nhập hệ thống:

```
// phần button đăng nhập
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:gravity="center" >
    <Button
        android:onClick="onClickLogin"
        android:id="@+id/button22"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Đăng nhập hệ thống" />
</LinearLayout>
```

Phần thông báo Toast

```
public void onClickLogin(View v){
    Toast.makeText(context: MainActivity.this,
        text: "Đăng nhập thành công", Toast.LENGTH_LONG).show();
}
```

## Bài 2

Tạo project bằng Android Studio có giao diện như hình 2.2

**Yêu cầu:** Dùng tableview để thiết kế giao diện, thực hiện các các phép tính cộng trừ nhân chia khi đã nhập đầy đủ các số vào 2 ô text, hiển thị kết quả trên textview Kết quả, Bấm vào button Thoát sẽ tắt chương trình.

## Hướng dẫn:

Tạo project mới, vận dụng layoutTableview để thiết kế giao diện, đưa các điều kiện vào form (có thể vận dụng các layout để trình bày). Bắt sự kiện click của các button

**Tham khảo mã nguồn:**

Activity\_tablelayout có các phần như sau:

```
<?xml version="1.0" encoding="utf-8"?>
<TableLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:stretchColumns="*"
    android:shrinkColumns="*">
    // phần nhập số A
    <TableRow...>
    // phần nhập số B
    <TableRow...>
    // phần các nút bấm cộng trừ nhân chia
    <TableRow...>
    // phần kết quả phép tính
    <TableRow...>
    // phần nút thoát
    <TableRow...>
</TableLayout>
```

Phần nhập số A (tương tự cho phần nhập số B)

Phần nút bấm cộng (tương tự cho trừ, nhân, chia)

```
// phần nhập số A
<TableRow
    android:layout_margin="3dp" >
    <TextView
        android:id="@+id/txtSoA"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Nhập số A"
        android:layout_span="1"
        android:textAlignment="textEnd"/>

    <EditText
        android:id="@+id/editSoA"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="numberDecimal"
        android:layout_span="2"/>
</TableRow>
```

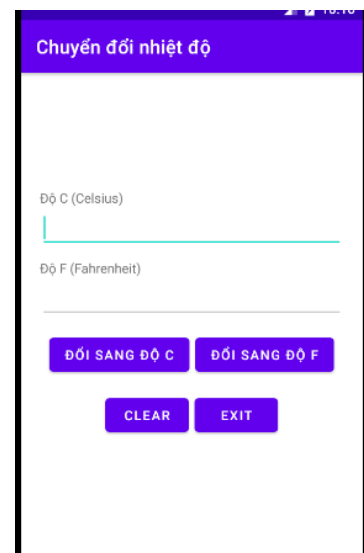
### Bài 3.

Viết chương trình chuyển đổi nhiệt độ C sang nhiệt độ F hoặc ngược lại bằng Android Studio Thiết kế giao diện như hình 2.3:

#### Hướng dẫn:

- Kết hợp LinearLayout-vertical và LinearLayout-horizontal để thiết kế thiết kế giao diện.
- Công thức chuyển đổi nhiệt độ:
 
$$^{\circ}\text{F} = ^{\circ}\text{C} \times \frac{9}{5} + 32$$

$$^{\circ}\text{C} = (^{\circ}\text{F} - 32) \times \frac{5}{9}$$
- Vận dụng công thức tính nhiệt độ để viết code cho các button.
- button exit Sử dụng hàm **finish()**;



Hình 2.3

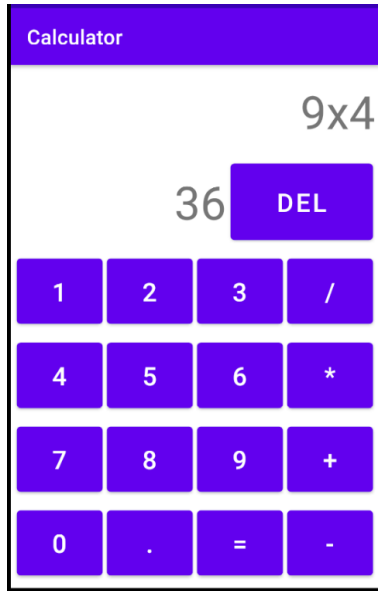
**Bài 4.**

Viết chương trình máy tính bỏ túi đơn giản, kết hợp một số kiểu Layout để thiết kế giao diện như hình bên (hình 2.4)

**Yêu cầu:**

Chương trình chỉ cần thực hiện phép toán đơn (chưa cần lồng phép toán – như ví dụ hình)

- Bấm Del: xóa dữ liệu hiện hữu trên EditText trên cùng và text phần kết quả.
- Bấm vào các con số → hiển thị lên EditText
- Bấm vào phép toán: +, -, \*, / → hiển thị lên EditText
- Bấm tiếp các con số → hiển thị lên EditText
- Bấm dấu "=" để thực hiện phép toán

**Hình 2.4****Hướng dẫn:**

Tham khảo xây dựng bố cục giao diện

```
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:weightSum="6"
    android:layout_margin="3dp">
    // phần hiển thị phép toán
    <LinearLayout...>
    //phần hiển thị kết quả và nút bấm del
    <LinearLayout...>
    //phần hiển thị các số bấm và phép toán
    <LinearLayout...>
</LinearLayout>
```

Chi tiết phần các số

```
//phần hiển thị các số bấm và phép toán
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:layout_weight="4">
    // phần giao diện các số 1,2,3 và phép chia
    <LinearLayout
        android:layout_width="match_parent" android:layout_height="match_parent"
        android:orientation="horizontal" android:layout_weight="1"
        android:layout_margin="1dp">
        <Button
            android:id="@+id/button14" android:layout_width="wrap_content"
            android:layout_height="match_parent" android:layout_margin="2dp"
            android:textSize="25dp" android:layout_weight="1"
            android:text="1" />
        <Button...>
        <Button...>
        <Button...>
    </LinearLayout>
    <LinearLayout...>
    <LinearLayout...>
    <LinearLayout...>
</LinearLayout>
```

Tham khảo mã nguồn các thực hiện các phép tính

## Bài 5.

Tạo project bằng Android Studio dựng 1 phần chương trình nhập thông tin cá nhân có giao diện như hình 2.5.

**Thực hiện yêu cầu:** Khi nhập đủ thông tin vào các mục, click vào button gửi sẽ hiển thị Alert Dialog có nội dung sau

Hình 2.5

## Bài 6.

Tạo project bằng Android Studio dựng 1 phần chương trình nhập thông tin cá nhân có giao diện như hình 2.6.

**Thực hiện yêu cầu:**

- Khi khởi chạy switch cựu sinh viên khởi tạo = **false**, nếu chọn **Switch = true** sẽ hiển thị phần cho phép nhập mã sinh viên:

- Bấm vào đồng ý sẽ hiển thị Alert Dialog chứa thông tin đã nhập.  
- Bấm vào Clear sẽ quay về khởi tạo ban đầu.

Hình 2.6



## Bài 7.

Tạo project bằng Android Studio có giao diện như hình 2.7

### Thực hiện các yêu cầu sau:

- Hiển thị thông báo (**dạng Toast**) nếu chưa nhập email hoặc email không hợp lệ (tương tự cho số điện thoại và ngày sinh) khi click vào button **Đồng ý**

Ví dụ:

Nhập số điện thoại

- Nếu thông tin đã nhập đầy đủ và đúng thì hiển thị Alert Dialog các thông tin đã nhập

Hình 2.7

**Hướng dẫn:** Tham khảo mã nguồn:

### Xây dựng giao diện

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="10dp"
    android:orientation="vertical"
    android:weightSum="12">
    <!--Phần chứa Email-->
    <LinearLayout...>
    <!--Phần chứa số điện thoại-->
    <LinearLayout...>
    <!--Phần nhập ngày tháng năm sinh-->
    <LinearLayout...>
    <!--Phần button Đồng ý-->
    <LinearLayout...>
    <!--...-->
    <LinearLayout...>
</LinearLayout>
```

Tham khảo phần chứa Email (hình bên), tương tự cho các phần khác của bài tập

```
<!--Phần chứa Email-->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="2"
    android:orientation="vertical"
    android:weightSum="2">
    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:gravity="center"
        android:text="Email"
        android:textAlignment="textStart"
        tools:ignore="RtlCompat" />
    <EditText
        android:id="@+id/editemail"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:ems="10"
        android:inputType="textEmailAddress" />
</LinearLayout>
```

## Tham khảo mã nguồn phần xử lý kiểm tra email và điện thoại

```
Button button;
EditText emailId;
// check email
String emailPattern = "[a-zA-Z0-9._-]+@[a-z]+\\.+[a-z]+";

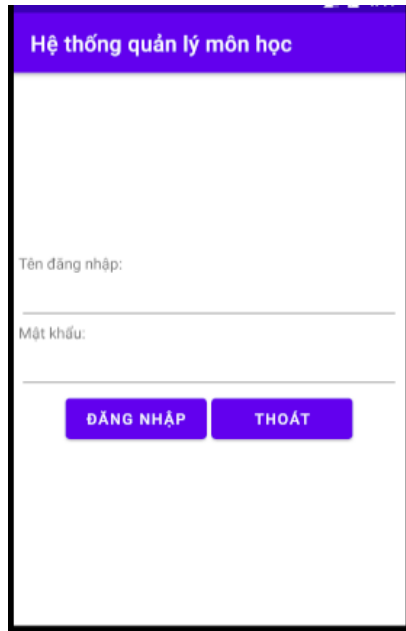
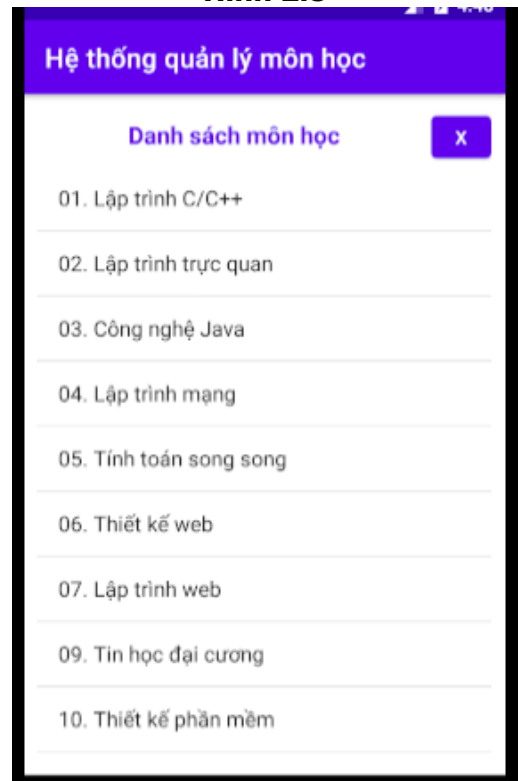
EditText editphoneNumber;
// check số điện thoại
String pattern = "^\\s*(?:\\+?(\\d{1,3}))?[-. (]*\\d{3}[-. (]*\\d{4}(?: *x(\\d+))?\\s*$";
Matcher m;
```

## Tham khảo mã nguồn phần xử lý cho button Đồng ý

```
// cách khác cho việc khai báo onclick
public void onClickDongY(View v) {
    button = findViewById(R.id.btDongY);
    emailId = findViewById(R.id.editemail);
    //Kiểm tra email
    if (emailId.getText().toString().isEmpty()) {
        // kiểm tra nếu chưa nhập địa chỉ
        Toast.makeText(getApplicationContext(), text: "Nhập địa chỉ email", Toast.LENGTH_SHORT).show();
    } else {
        if (emailId.getText().toString().trim().matches(emailPattern) == false) {
            // Kiểm tra không phải địa chỉ email
            Toast.makeText(getApplicationContext(), text: "Email không đúng", Toast.LENGTH_SHORT).show();
        } else {
            // Kiểm tra đúng là địa chỉ email => thực hiện các hành động đúng tại đây
            Toast.makeText(getApplicationContext(), text: "Địa chỉ email hợp lệ", Toast.LENGTH_SHORT).show();
        }
    }
    // kiểm tra số điện thoại
    editphoneNumber = findViewById(R.id.editphone);
    Pattern r = Pattern.compile(pattern);
    if (editphoneNumber.getText().toString().trim().equals("") == true) {
        Toast.makeText(context: MainActivity.this, text: "Nhập số điện thoại", Toast.LENGTH_SHORT).show();
    }
    else {
        m = r.matcher(editphoneNumber.getText().toString().trim());
        if (m.find()) {
            Toast.makeText(context: MainActivity.this, text: "SĐT hợp lệ", Toast.LENGTH_SHORT).show();
        } else {
            Toast.makeText(context: MainActivity.this, text: "SĐT không hợp lệ", Toast.LENGTH_SHORT).show();
        }
    }
    // đặt focus
    if (emailId.getText().toString().isEmpty() ||
        emailId.getText().toString().trim().matches(emailPattern) == false) {
        emailId.requestFocus();
    }
    else if (editphoneNumber.getText().toString().trim().equals("") == true || m.find() == false) {
        editphoneNumber.requestFocus();
    }
}
```

**Bài 8.**

Tạo project bằng Android Studio có giao diện ban đầu như hình sau:

**Hình 2.8****Thực hiện các yêu cầu:**

- click vào Thoát => thoát chương trình
- Để đăng nhập hệ thống nhập vào:  
tên đăng nhập: admin  
mật khẩu: 1

\* Nếu nhập sai sẽ đưa ra thông báo lỗi (**Toast**) ví dụ:

Tên đăng nhập hoặc mật khẩu sai

\* Nếu nhập đúng sẽ chuyển sang form danh sách môn học như **hình 2.8**

Hướng dẫn:

Tham khảo mã nguồn:

```
<!--Thiết lập bố cục tổng thể-->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    android:layout_margin="10dp">
    <!--Phần tên đăng nhập-->
    <LinearLayout...>
    <!--Phần mật khẩu-->
    <LinearLayout...>
    <!--Phần button-->
    <LinearLayout...>
</LinearLayout>
```

- Trong form danh sách môn học sẽ hiển thị các môn học được đưa list.

- Khi click vào button **X** sẽ thoát khỏi form và quay lại giao diện đăng nhập

**Tương tự cho các phần còn lại**

```
<!--Phần tên đăng nhập-->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">
    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Tên đăng nhập:" />
    <EditText
        android:id="@+id/editTenDangNhap"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="" />
</LinearLayout>
```

**Phần list môn học**

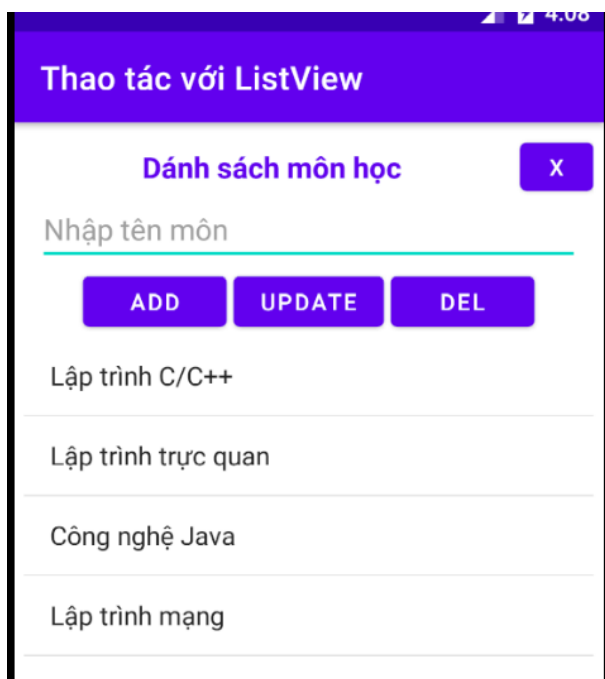
```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:weightSum="12"
    android:layout_margin="10dp">
    <!--Phần tiêu đề và button exit-->
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:layout_weight="1"
        android:weightSum="4">
        <TextView...>
        <Button...>
    </LinearLayout>
    <!--Phần listview-->
    <LinearLayout
        android:layout_weight="11"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">
        <ListView
            android:id="@+id/listDSMonHoc"
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="1"/>
    </LinearLayout>
</LinearLayout>


```

## Bài 9.

Tạo project bằng Android Studio có giao diện ban đầu như hình sau:



### Thực hiện các yêu cầu:

- Khi chương trình chạy hiển thị danh sách các môn học trong listview.
- Khi click vào button  sẽ khởi tạo lại các giá trị mặc định ban đầu
- Sau khi nhập dữ liệu vào editText, click vào add sẽ thêm dữ liệu vào list và đưa ra thông báo thêm thành công. Trường hợp chưa nhập dữ liệu sẽ đưa ra thông báo yêu cầu nhập dữ liệu.

Vui lòng nhập tên môn

- Nếu chọn 1 dòng dữ liệu nào đó trong Listview sẽ hiển thị text lên editText; có thể sửa hoặc xóa dữ liệu này bằng cách click vào button Update, button Delete, Trường hợp chưa chọn dữ liệu sẽ đưa ra thông báo chưa chọn dữ liệu

Vui lòng chọn dữ liệu!

**Hướng dẫn:**

Có thể sử dụng LinearLayout để thiết kế, xây dựng bố cục giao diện cho hợp lý (Tham khảo mã nguồn).

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="6dp"
    android:orientation="vertical"
    android:weightSum="12">
    <!--Phần tiêu đề và button reload-->
    <LinearLayout...>
    <!--Phần thêm, xóa, sửa dữ liệu-->
    <LinearLayout...>
    <!--Phần listview-->
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:orientation="vertical"
        android:layout_weight="9">
        <ListView
            android:id="@+id/listDSMonHoc"
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="1"
            android:choiceMode="singleChoice" />
    </LinearLayout>
</LinearLayout>
```

```
<!--Phần thêm, xóa, sửa dữ liệu-->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="2"
    android:gravity="center"
    android:orientation="vertical"
    android:textAlignment="gravity">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:layout_marginLeft="8dp"
        android:layout_marginRight="8dp">
        <EditText...>
    </LinearLayout>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:gravity="center"
        android:textAlignment="gravity">
        <Button...>
        <Button...>
        <Button...>
    </LinearLayout>
</LinearLayout>
```

**Phần mã nguồn xử lý dữ liệu:**

```
public class MainActivity extends AppCompatActivity {
    List<String> data; // = new ArrayList<>();
    private List setListView() {
        data = new ArrayList<>();
        data.add("Lập trình C/C++");
        data.add("Lập trình trực quan");
        data.add("Công nghệ Java");
        return data;
    }
    private List addItemList() {...}
    private List updateItemList(int index) {...}
    private List removeItemList(int index) {...}
    private int indexSelected;
    private ListView listItemDSMonHoc;
    @Override
    protected void onCreate(Bundle savedInstanceState) {...}
    public void onClickReload(View v) {...}
    public void onClickAdd(View v) {...}
    public void onClickUpdate(View v) {...}
    public void onClickDelete(View v) {...}
}
```

**Chi tiết các hàm thêm, sửa, xóa**

```
private List addItemList() {
    TextView tenMon = findViewById(R.id.editTenMon);
    data.add(tenMon.getText().toString());
    return data;
}
private List updateItemList(int index) {
    TextView tenMon = findViewById(R.id.editTenMon);
    data.set(index, tenMon.getText().toString());
    return data;
}
private List removeItemList(int index) {
    TextView tenMon = findViewById(R.id.editTenMon);
    data.remove(index);
    return data;
}
private int indexSelected;
private ListView listItemDSMonHoc;
```

## Chi tiết phần xử lý các button Add, Update, Del

```

public void onClickReload(View v) {
    listItemDSMonHoc = findViewById(R.id.listDSMonHoc);
    ArrayAdapter<String> adapter = new ArrayAdapter<String>(
        context: this, android.R.layout.simple_list_item_1, setListView());
    listItemDSMonHoc.setAdapter(adapter);
}

public void onClickAdd(View v) {
    TextView tenMon = findViewById(R.id.editTenMon);
    if (tenMon.getText().toString().trim().equals("") == true) {
        Toast.makeText(context: MainActivity.this,
            text: "Vui lòng nhập tên môn", Toast.LENGTH_SHORT).show();
        tenMon.setText(""); tenMon.requestFocus();
    } else {
        listItemDSMonHoc = findViewById(R.id.listDSMonHoc);
        ArrayAdapter<String> adapter = new ArrayAdapter<String>(
            context: this, android.R.layout.simple_list_item_1, addItemList());
        listItemDSMonHoc.setAdapter(adapter);
        Toast.makeText(context: MainActivity.this,
            text: "Thêm thành công!", Toast.LENGTH_SHORT).show();
        tenMon.setText(""); tenMon.requestFocus();
    }
}

public void onClickUpdate(View v) {...}
public void onClickDelete(View v) {...}

```

Lưu ý: Hàm Update và Delete sử dụng phương thức Set() và Remove() của List để thực hiện

**Phần xử lý hàm onCreate:**

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    listItemDSMonHoc = findViewById(R.id.listDSMonHoc);
    ArrayAdapter<String> adapter = new ArrayAdapter<String>(context: this,
        android.R.layout.simple_list_item_1, setListView());
    listItemDSMonHoc.setAdapter(adapter);
    // xử lý sự kiện chọn 1 item trong listview
    listItemDSMonHoc.setOnItemClickListener(new AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> adapterView, View view, int indexItem, long l) {
            EditText tenMonUpdate = findViewById(R.id.editTenMon);
            tenMonUpdate.setText(data.get(indexItem));
            indexSelected = indexItem; // lấy chỉ số phần tử
            int FocusEnd = tenMonUpdate.getText().length(); // lấy độ dài text
            tenMonUpdate.setSelection(FocusEnd); // gán focus xuống cuối
        }
    });
}

```

## Bài 10

Tạo project bằng Android Studio có giao diện ban đầu như hình sau:

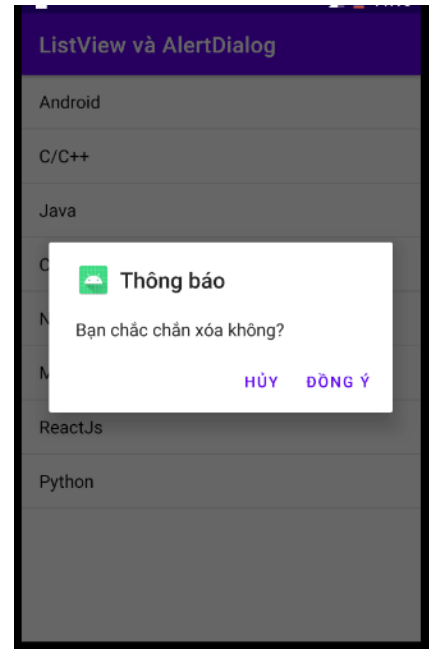
### Thực hiện các yêu cầu:

- Khi chương trình chạy hiển thị danh sách các môn học trong ListView (hình 10.1)
- Khi click vào 1 dòng nào đó sẽ hiển thị thông báo xóa dữ liệu (hình 10.2)
- + Đồng ý để xóa dữ liệu
- + Hủy bỏ để trở lại màn hình 10.1

Hình 10.1



Hình 10.2



### Hướng dẫn:

Tham khảo mã nguồn xây dựng giao diện

```
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="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=".MainActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <ListView
            android:id="@+id/listviewName"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:choiceMode="singleChoice" />

    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

Tham khảo mã nguồn xử lý dữ liệu:

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

    lvName = (ListView) findViewById(R.id.listviewName);
    arrayName = new ArrayList<>();
    AddArrayName();
    adapter = new ArrayAdapter<>(context, this,
        android.R.layout.simple_list_item_1, arrayName);
    lvName.setAdapter(adapter);

    lvName.setOnItemClickListener(new AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<> adapterView, View v, int i, long l) {
            ShowDialogDelete(i);
            return;
        }
    });
}
```

### Mã nguồn dialog Confirm Delete

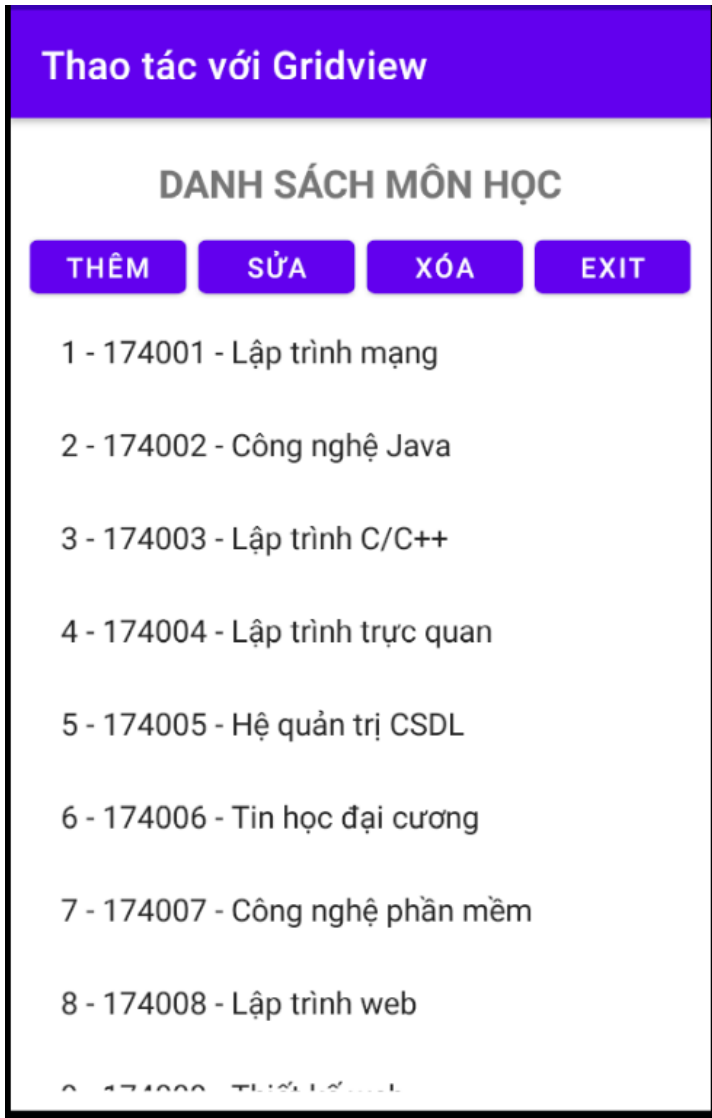
```
private void ShowDialogDelete(int position) {
    AlertDialog.Builder alertDialog = new AlertDialog.Builder(context, this);
    alertDialog.setTitle("Thông báo");
    alertDialog.setIcon(R.mipmap.ic_launcher);
    alertDialog.setMessage("Bạn chắc chắn xóa không?");

    alertDialog.setNegativeButton("Hủy",
        new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialogInterface, int i) {
            }
        });
    alertDialog.setPositiveButton("Đồng ý",
        new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialogInterface, int i) {
                arrayName.remove(position);
                adapter.notifyDataSetChanged();
            }
        });
    alertDialog.show();
}
```



**Bài 11.**

Tạo project bằng Android Studio có giao diện ban đầu như hình sau:

**Thực hiện các yêu cầu:**

- Khi chương trình chạy hiển thị danh sách các môn học trong Gridview.
- Khi click vào button Exit sẽ thoát chương trình.
- Khi click vào button Thêm => chuyển form thêm mới:
- Khi click vào button Sửa => chuyển form sửa có chứa dữ liệu được chọn, trường hợp chưa chọn dữ liệu sẽ hiển thị thông báo yêu cầu chọn dữ liệu.
- Khi click vào button Xóa sẽ hiển thị thông báo đồng ý xóa hay không, nếu đồng ý => sẽ xóa dữ liệu được chọn. trường hợp chưa chọn dữ liệu sẽ hiển thị thông báo có nội dung **"Chưa chọn dữ liệu để xóa"**



## BÀI THỰC HÀNH SỐ 3

Xây dựng ứng dụng android với các điều khiển nâng cao

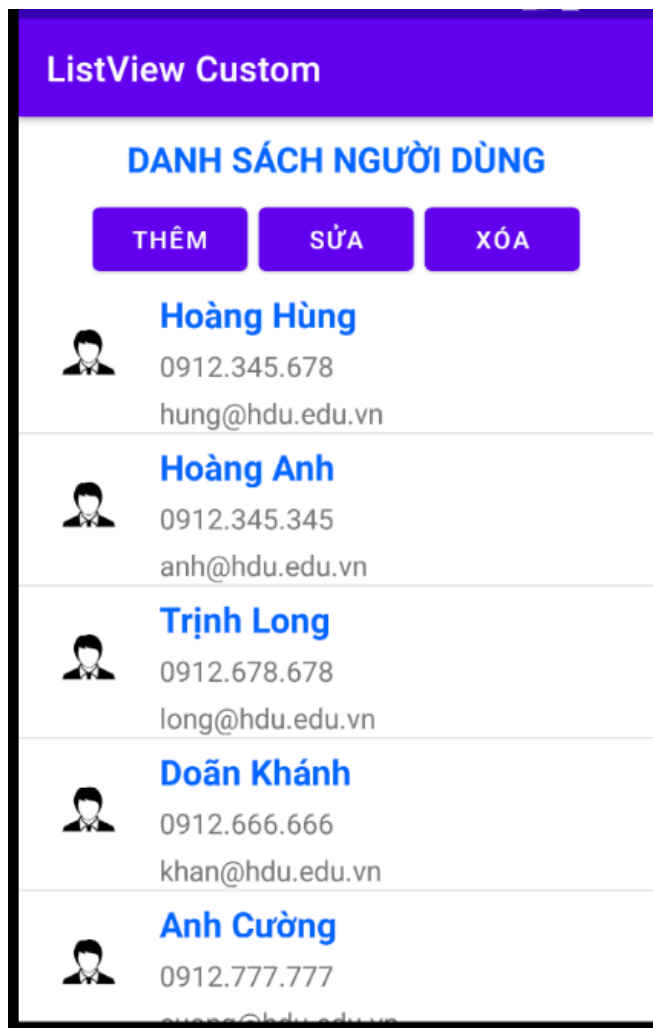
### 1. Mục tiêu

- Biết cách khởi tạo dự án Android
- Biết cách khởi chạy thiết bị ảo
- Biết tạo giao diện một số view phức tạp,
- Biết cách lập trình di chuyển giữa các màn hình (Intent)
- Biết cách sử dụng các loại điều khiển nâng cao (MenuView, TabView WebView, ...) để xây dựng các ứng dụng chạy trên nền tảng android

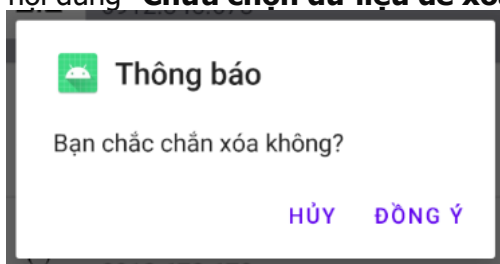
### 2. Nội dung:

#### Bài 1.

Tạo project bằng Android Studio có giao diện ban đầu như hình sau:



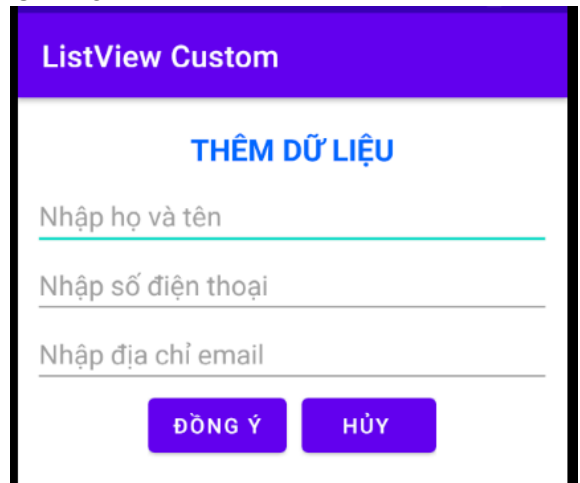
- Khi click vào button Xóa sẽ hiển thị thông báo đồng ý xóa hay không, nếu đồng ý => sẽ xóa dữ liệu được chọn. trường hợp chưa chọn dữ liệu sẽ hiển thị thông báo có nội dung **"Chưa chọn dữ liệu để xóa"**



#### Thực hiện các yêu cầu:

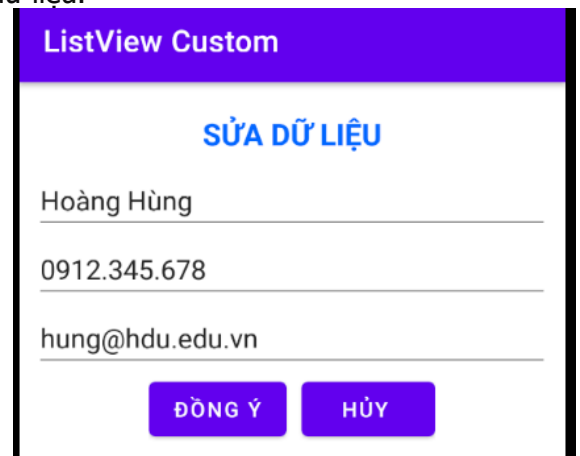
- Khi chương trình chạy hiển thị danh sách người dùng có: họ tên, điện thoại, email trong list view như hình.

- Khi click vào button Thêm => chuyển form thêm mới:



+ Click vào hủy để quay lại form danh sách  
+ Click vào đồng ý để thêm dữ liệu (Lưu ý bắt lỗi khi thêm dữ liệu)

- Khi click vào button Sửa => chuyển form sửa có chứa dữ liệu được chọn, trường hợp chưa chọn dữ liệu sẽ hiển thị thông báo yêu cầu chọn dữ liệu.



+ Click vào hủy để quay lại form danh sách  
+ Click vào đồng ý để thêm dữ liệu (Lưu ý bắt lỗi khi thêm dữ liệu)

Hướng dẫn:  
Tham khảo mã nguồn

## Bài 2.

Tạo project bằng Android Studio có giao diện ban đầu như hình sau:

The screenshot shows an Android app interface titled "Gridview Custom". Below the title is a section titled "DANH SÁCH CẦU THỦ" (List of Players). There are three buttons at the top: "THÊM" (Add), "SỬA" (Edit), and "XÓA" (Delete). Below these buttons is a list of ten players, each with a profile icon, name, phone number, and email address. The players are arranged in two columns.

THÊM	SỬA	XÓA
<b>Nguyễn Công Phụng</b> 0912.345.678 phuongnc@hdu.edu.vn	<b>Hồ Tấn Tài</b> 0912.345.345 tantai@hdu.edu.vn	
<b>Bùi Tiến Dũng</b> 0912.678.678 tiendung@hdu.edu.vn	<b>Bùi Tiến Dũng</b> 0912.666.666 dungbt@hdu.edu.vn	
<b>Nguyễn Duy Mạnh</b> 0912.777.774 manhnd@hdu.edu.vn	<b>Nguyễn Quang Hải</b> 0912.777.775 quanghai@hdu.edu.vn	
<b>Đoàn Văn Hậu</b> 0912.777.776 haudv@hdu.edu.vn	<b>Quế Ngọc Hải</b> 0912.777.777 haiqn@hdu.edu.vn	
<b>Lương Xuân Trường</b> 0912.777.778 xuantruong@hdu.edu.vn	<b>Nguyễn Tuấn Anh</b> 0912.777.779 tuananh@hdu.edu.vn	

- Khi click vào button Xóa sẽ hiển thị thông báo đồng ý xóa hay không, nếu đồng ý => sẽ xóa dữ liệu được chọn. trường hợp chưa chọn dữ liệu sẽ hiển thị thông báo có nội dung "**Chưa chọn dữ liệu để xóa**"

The screenshot shows a confirmation dialog box titled "Thông báo" (Notification). The text inside says "Bạn chắc chắn xóa không?" (Are you sure you want to delete?). There are two buttons at the bottom: "HỦY" (Cancel) and "ĐỒNG Ý" (Yes).

### Thực hiện các yêu cầu:

- Khi chương trình chạy hiển thị danh sách người dùng có: họ tên, điện thoại, email trong list view như hình.

- Khi click vào button Thêm => chuyển form thêm mới:

The screenshot shows a form titled "Gridview Custom" with a section titled "THÊM DỮ LIỆU" (Add data). There are three input fields: "Nhập họ và tên" (Enter name and surname), "Nhập số điện thoại" (Enter phone number), and "Nhập địa chỉ email" (Enter email address). At the bottom, there are two buttons: "ĐỒNG Ý" (Yes) and "HỦY" (Cancel).

+ Click vào hủy để quay lại form danh sách  
+ Click vào đồng ý để thêm dữ liệu (Lưu ý bắt lỗi khi thêm dữ liệu)

- Khi click vào button Sửa => chuyển form sửa có chứa dữ liệu được chọn, trường hợp chưa chọn dữ liệu sẽ hiển thị thông báo yêu cầu chọn dữ liệu.

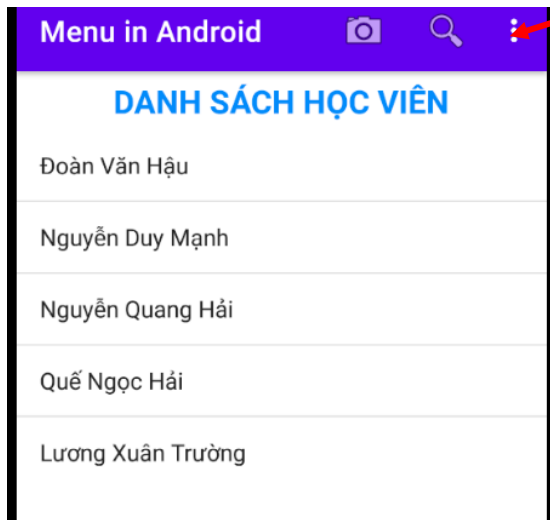
The screenshot shows a form titled "Gridview Custom" with a section titled "SỬA DỮ LIỆU" (Edit data). The form contains three input fields with pre-filled data: "Nguyễn Công Phụng", "0912.345.678", and "phuongnc@hdu.edu.vn". At the bottom, there are two buttons: "ĐỒNG Ý" (Yes) and "HỦY" (Cancel).

+ Click vào hủy để quay lại form danh sách  
+ Click vào đồng ý để thêm dữ liệu (Lưu ý bắt lỗi khi thêm dữ liệu)

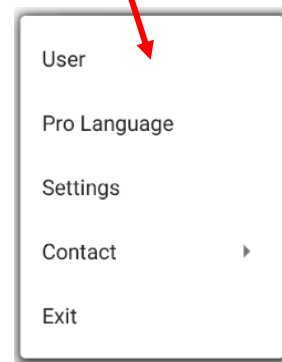
**Hướng dẫn:** Làm tương tự bài 1, đối với gridview thêm thuộc tính chia cột

**Bài 3.**

Tạo project bằng Android Studio có giao diện ban đầu chứa menu như hình sau:

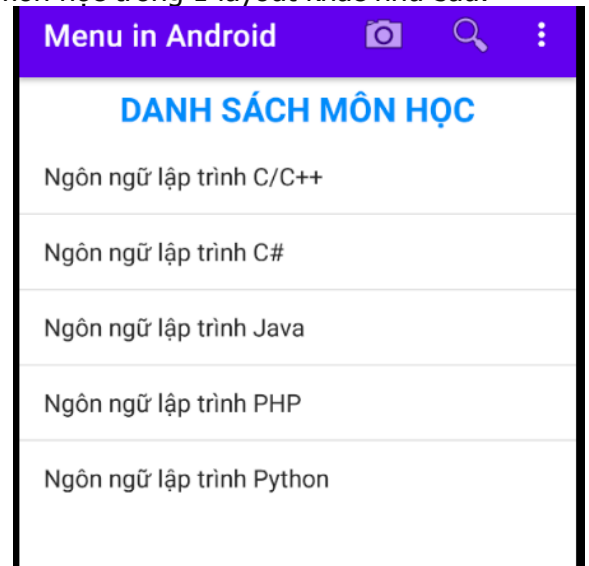
**Thực hiện các yêu cầu:**

- Menu có các item như sau:

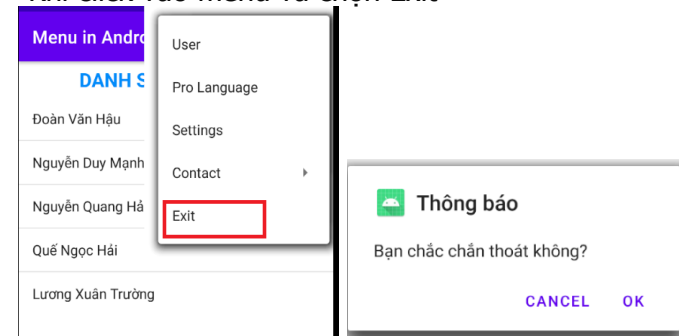


- Khi chương trình chạy hiển thị danh sách họ tên các học viên Listview User như hình.

- Chọn Pro Language để hiển thị danh sách các môn học trong 1 layout khác như sau:

**Thực hiện các yêu cầu:**

- Khi click vào menu và chọn Exit

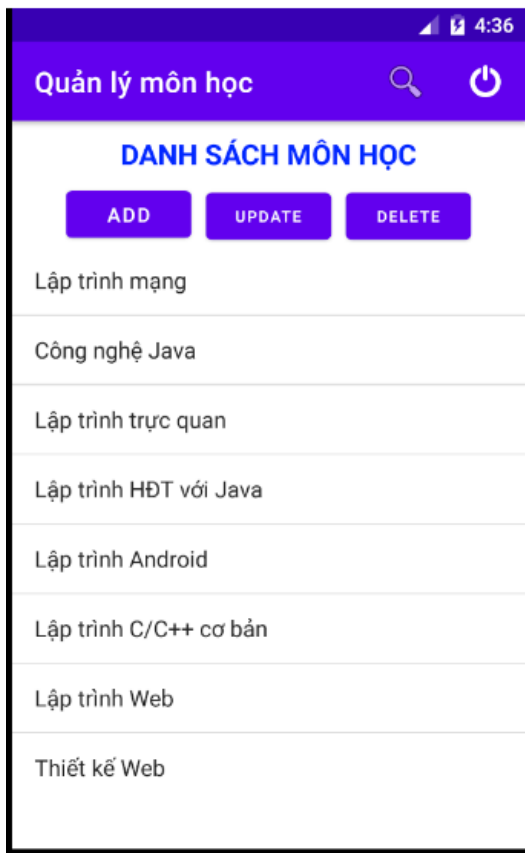


sẽ hiển thị thông báo đồng ý thoát hay không, Bấm OK để thoát chương trình

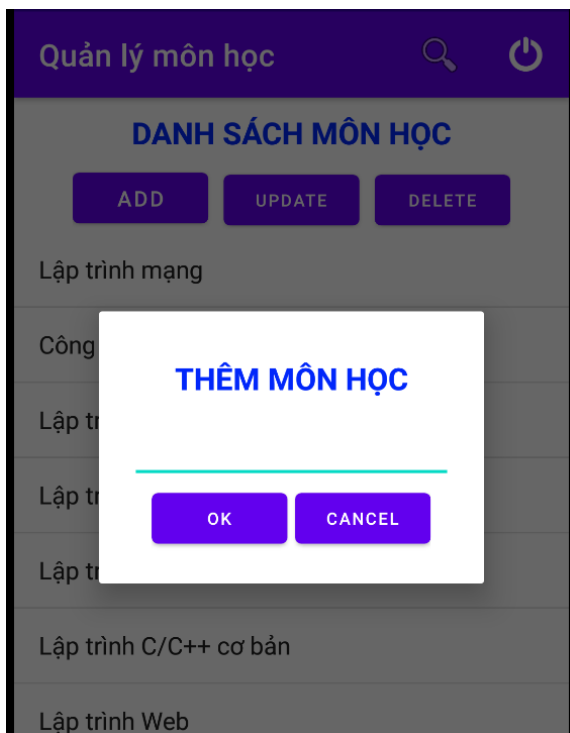
**Hướng dẫn:**

#### Bài 4.

Xây dựng App bằng Android Studio thực hiện quản lý các học phần chuyên ngành CNTT có giao diện như hình sau:

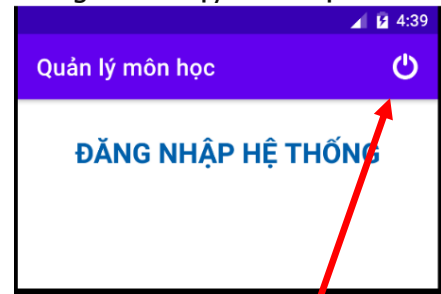


+ Click vào **button ADD** => hiển thị dialog thêm mới môn học (Tương tự cho **button Update**)

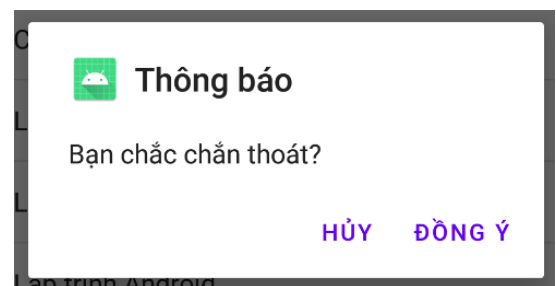


#### Thực hiện các yêu cầu:

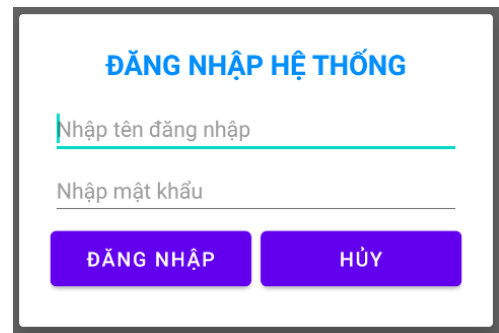
- Khi chương trình chạy hiển thị form sau:



+ Click vào biểu tượng **Thoát** hiển thị dialog thông báo hỏi tắt chương trình hay không? Bấm đồng ý để tắt chương trình

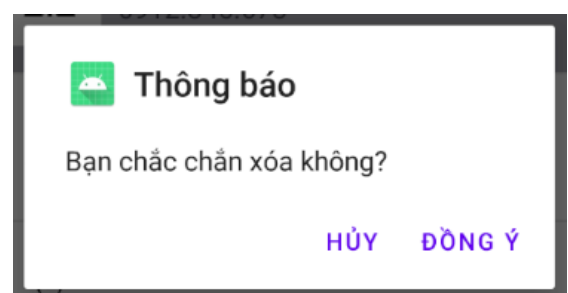


+ Click vào **ĐĂNG NHẬP HỆ THỐNG** để => hiển thị dialog để đăng nhập hệ thống:

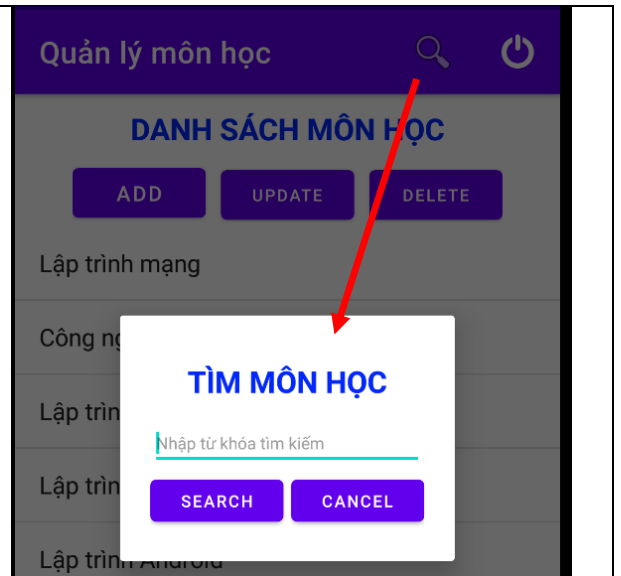


=> Đăng nhập thành công sẽ vào giao diện như yêu cầu.

- Khi click vào **button Delete** sẽ hiển thị thông báo đồng ý xóa hay không, nếu đồng ý => sẽ xóa dữ liệu được chọn. (trường hợp chưa chọn dữ liệu sẽ hiển thị thông báo có nội dung "**Chưa chọn dữ liệu để xóa**")



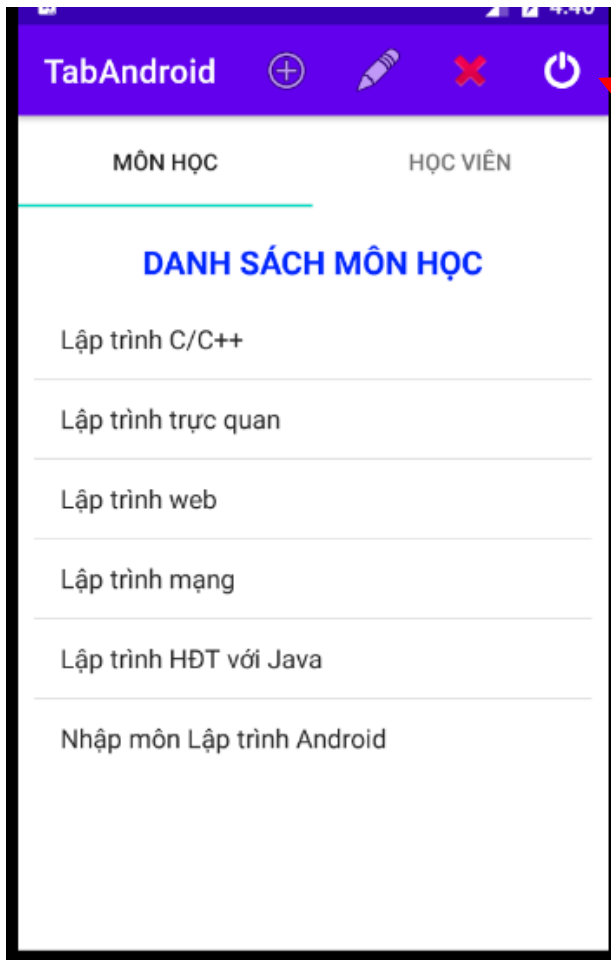
Click vào biểu tượng tìm kiếm sẽ hiển thị dialog nhập tên môn học để tìm, Click vào search sẽ thực hiện tìm dữ liệu; dữ liệu sau khi tìm được sẽ hiển thị vào ds ; (Sinh viên thiết kế và xây dựng thêm phần reload data)



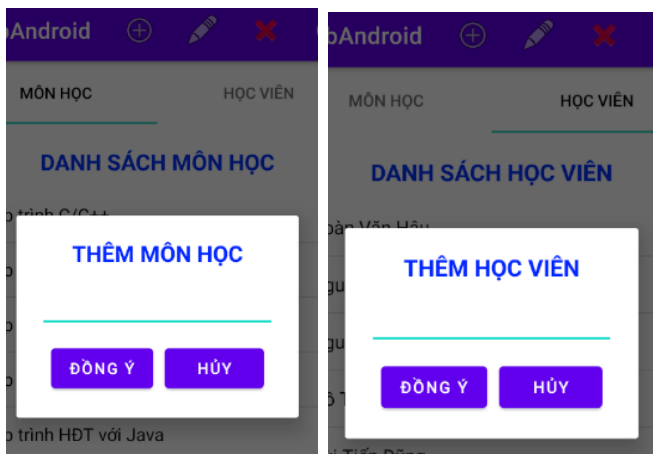
**Hướng dẫn:**

## Bài 5.

Xây dựng App bằng Android Studio sử dụng controller **tabhost** có giao diện như hình sau:



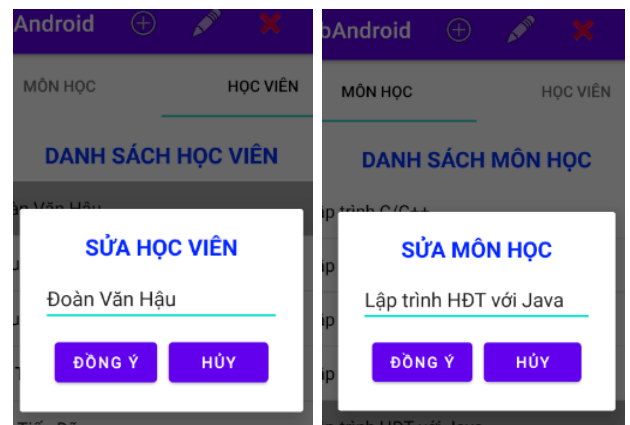
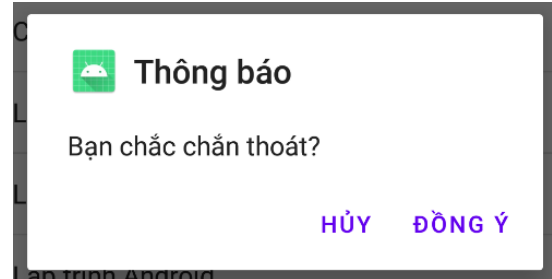
+ Khi chọn tab nào đó thì có thể Thực hiện **thêm, sửa, xóa** dữ liệu tương ứng bằng cách click vào biểu tượng trên menu.



### Thực hiện các yêu cầu:

- Menu có các chức năng **Thêm, Sửa, Xóa** và **Tắt** chương trình.
- Tabhost có 2 tab nhỏ có thông tin các môn học và thông tin các học viên.

- Thoát chương trình được thực hiện khi click vào menu Thoát



## Bài 6.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu sau

## BÀI THỰC HÀNH SỐ 4

Đọc ghi dữ liệu, giao tiếp hình ảnh, âm thanh, Gửi email, gửi SMS, Gọi điện thoại.

### 1. Mục tiêu

- Biết cách lập trình xử lý các giao tiếp với camera
- Biết cách lập trình đọc ghi file
- Biết cách lập trình ứng dụng hoàn chỉnh

### 2. Nội dung:

#### Bài 1.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu Chụp ảnh

#### Bài 2.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu Quay video

Chụp ảnh và lưu vào ổ đĩa có sẵn

```
<manifest ...>
```

```
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
```

```
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
```

```
    ...
```

```
</manifest>
```

#### Bài 3.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu Chuyển văn bản thành giọng nói

#### Bài 4.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu Gọi điện thoại

#### Bài 5.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu Gửi tin nhắn SMS

#### Bài 6.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu Gửi email

## BÀI THỰC HÀNH SỐ 5, 6

Xây dựng giao diện người dùng, biên dịch và chạy ứng dụng trên thiết bị thật

### 1. Mục tiêu

- Biết cách khởi tạo dự án Android
- Biết cách lập trình xử lý qua lại giữa các Activity
- Biết tạo giao diện người dùng, biết cách lập trình đa điểm,
- Biết cách lập trình ứng dụng hoàn chỉnh, biên dịch và chạy ứng dụng trên thiết bị ảo, thiết bị thật

### 2. Nội dung:

Bài 1.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu sau

Bài 2.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu sau

Bài 3.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu sau

Bài 4.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu sau

Bài 5.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu sau

Bài 6.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu sau

## BÀI THỰC HÀNH SỐ 6

Xây dựng ứng dụng android kết nối cơ sở dữ liệu, lấy và hiển thị dữ liệu

### 1. Mục tiêu

- Biết cách khởi tạo dự án Android
- Biết tạo giao diện người dùng
- Biết cách lập trình xử lý với tệp (đọc/ghi)
- Biết cách lập trình kết nối cơ sở dữ liệu
- Biết cách lập trình ứng dụng hoàn chỉnh

### 2. Nội dung:

Bài 1.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu sau

Bài 2.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu sau

Bài 3.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu sau

Bài 4.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu sau

Bài 5.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu sau

Bài 6.

Tạo project bằng Android Studio có giao diện và thực hiện các yêu cầu sau



