

Lab 8. List trong Android (4 tiết)

I. Yêu cầu

- Sinh viên sử dụng công cụ Android Studio để làm bài. Kết quả bài làm cần được chụp lại và giữ lại toàn bộ dự án để sử dụng khi cần thiết.
- Mỗi người làm trên một dự án khác nhau.
- Khi có yêu cầu, sinh viên nộp qua email giáo viên hoặc một kênh khác.

II. Luyện tập

1. ListView

ListView là phần tử View được dùng để hiển thị dữ liệu là một danh sách (mảng) từ một nguồn cấp gọi là Adapter.

Các bước để tạo và sử dụng ListView bao gồm:

Bước 1. Tạo một ListView trong Layout:

```
<?xml version="1.0" encoding="utf-8"?>|
<LinearLayout android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:weightSum="10"
    xmlns:android="http://schemas.android.com/apk/res/android">
    <ListView
        android:id="@+id/lvMain"
        android:layout_weight="8"
        android:layout_width="match_parent"
        android:layout_height="0dp">
    </ListView>
    <TextView
        android:id="@+id/txtDisplay"
        android:layout_weight="2"
        android:layout_width="match_parent"
        android:layout_height="0dp"/>
</LinearLayout>
```

Bước 2. Trong lớp MainActivity.java, khai báo một tập dữ liệu và hai đối tượng như sau:

```
String[] arr = new String[]{"Android", "iOS", "Window Phone"};  
ListView lvMain;  
TextView txtDisplay;
```

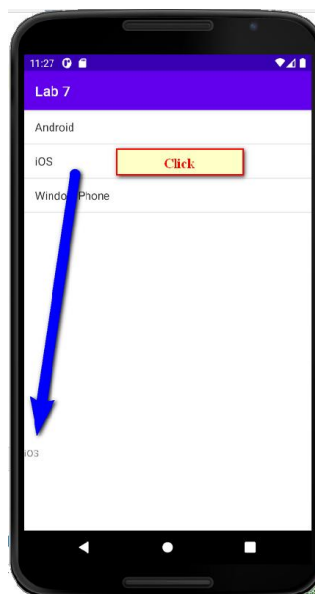
Trong hàm onCreate, Để đưa dữ liệu vào ListView, ta phải khai báo một Adapter kiểu String và thêm vào ListView như sau:

```
lvMain = findViewById(R.id.lvMain);  
txtDisplay = findViewById(R.id.txtDisplay);  
ArrayAdapter<String> adapter = new ArrayAdapter<>( context: this,  
    android.R.layout.simple_list_item_1,arr);  
lvMain.setAdapter(adapter);
```

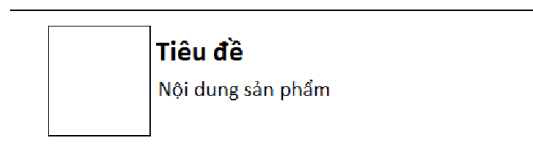
Bước 3: Để người dùng click vào từng dòng, dòng đó sẽ hiển thị thông tin lên TextView, ta thực hiện thêm sự kiện setOnItemClickListener cho ListView như sau:

```
lvMain.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
    @Override  
    public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  
        txtDisplay.setText(arr[position]);  
    }  
});
```

Chạy kết quả, khi bấm vào 1 dòng trong ListView, nội dung sẽ hiện ra ở phần TextView như sau:



Tiếp theo, chúng ta cải tiến để sử dụng Custom ListView. Giả sử ListView sẽ hiện ra các dòng, mỗi dòng bao gồm hình ảnh, tiêu đề và mô tả của sản phẩm:



Thứ tự thực hiện như sau:

Bước 4. Xây dựng một lớp có tên Product như sau:

```
public class Product {
    public int image;
    public String title;
    public String content;

    public Product(int image, String title, String content) {
        this.image = image;
        this.title = title;
        this.content = content;
    }
}
```

Bước 5. Xây dựng một layout để trang trí cho dòng dữ liệu, layout này bao gồm 1 ImageView và 2 TextView như sau:

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    xmlns:android="http://schemas.android.com/apk/res/android">
    <ImageView
        android:id="@+id/imgRow"
        android:layout_width="60dp"
        android:layout_height="match_parent" />
    <LinearLayout
        android:layout_width="wrap_content"
        android:orientation="vertical"
        android:layout_marginLeft="10dp"
        android:layout_marginTop="5dp"
        android:weightSum="3"
        android:layout_height="match_parent">
        <TextView
            android:id="@+id/txtTitle"
            android:layout_width="match_parent"
            android:layout_weight="1"
            android:textStyle="bold"
```

```

        android:textColor="#0942EC"
        android:layout_height="0dp"/>
    <TextView
        android:id="@+id/txtContent"
        android:layout_width="match_parent"
        android:layout_weight="1"
        android:layout_height="0dp"/>
    </LinearLayout>
</LinearLayout>

```

Bước 6. Xây dựng lớp *MyListViewAdapter*, có các phương thức như sau:

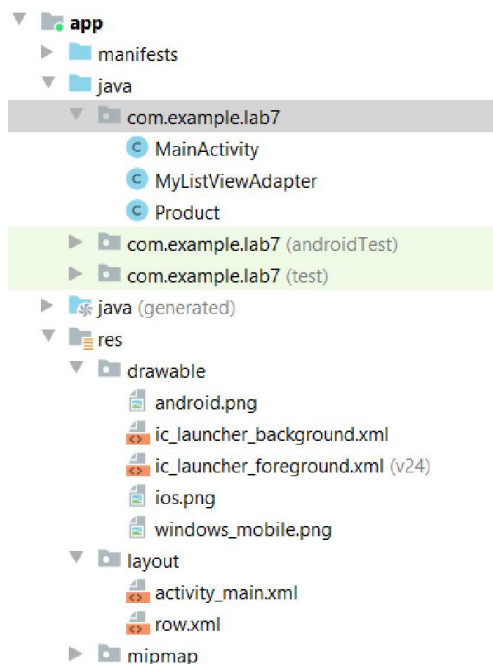
```

public class MyListViewAdapter extends BaseAdapter {
    ArrayList<Product> list;
    public MyListViewAdapter(ArrayList<Product> list){
        this.list = list;
    }
    @Override
    public int getCount() {
        return list.size();
    }
    @Override
    public Object getItem(int position) {
        return list.get(position);
    }
    @Override
    public long getItemId(int position) {
        return 0;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent) {
        View view = View.inflate(parent.getContext(), R.layout.row, root: null);
        Product product = (Product)getItem(position);
        ImageView img = view.findViewById(R.id.imgRow);
        img.setImageResource(product.image);
        TextView txtTitle = view.findViewById(R.id.txtTitle);
        txtTitle.setText(product.title);
        TextView txtContent = view.findViewById(R.id.txtContent);
        txtContent.setText(product.content);
        return view;
    }
}

```

Bước 7. Lên trang icon8, tìm các hình của android, iOS, windows phone, tải về dạng 64px, và đưa vào drawable. Lúc này, cấu trúc dự án như sau:



Bước 8. Trong lớp MainActivity, thay đổi danh sách thiết bị tại Bước 2 bằng 3 danh sách như sau:

```
String[] titles = new String[]{"Android", "iOS", "Window Phone"};
String[] contents = new String[]{"Đây là hệ điều hành Android",
                                "Đây là hệ điều hành iOS",
                                "Đây là hệ điều hành Window Phone"};
int[] imgs = new int[]{ R.drawable.android,
                        R.drawable.ios,
                        R.drawable.windows_mobile};
```

Bước 9. Trong hàm onCreate, khai báo ArrayList của Product và Adapter mới như sau:

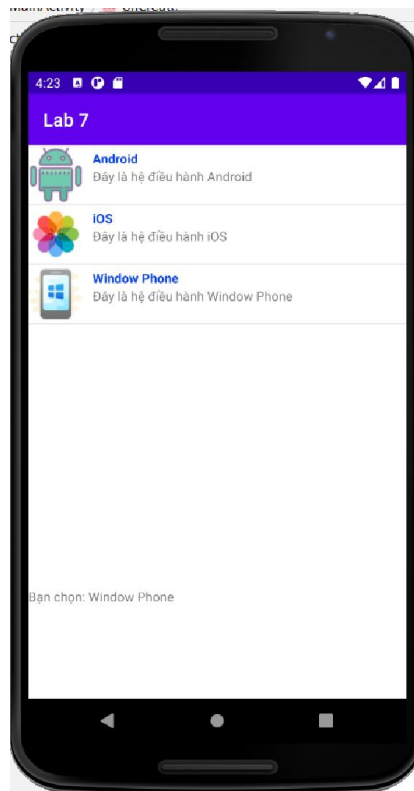
```
//ArrayAdapter<String> adapter = new ArrayAdapter<>(this,
//          android.R.layout.simple_list_item_1,arr);
ArrayList<Product> list = new ArrayList<>();
for(int i = 0; i< titles.length; i++){
    list.add(new Product(imgs[i],titles[i],contents[i]));
}
MyListViewAdapter adapter = new MyListViewAdapter(list);
lvMain.setAdapter(adapter);
```

Lưu ý: Phần code bị che chính là code ở Bước 2.

Bước 10. Thay đổi đoạn code ở Bước 3 thành như sau để khi người dùng click vào từng đối tượng thì hiển thị ở TextView ở phía dưới:

```
lvMain.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
    @Override  
    public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  
        txtDisplay.setText("Bạn chọn: " + titles[position]);  
    }  
});
```

Bước 11. Chạy kết quả trên màn hình giả lập:

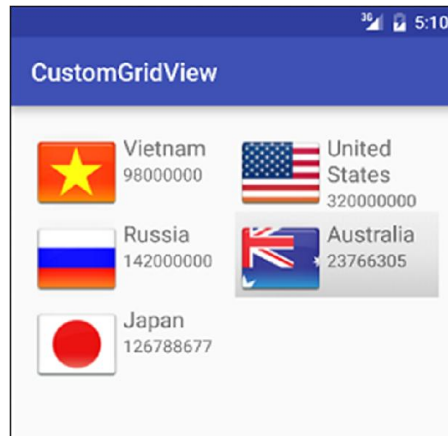


Các bạn có thể xem thêm về ListView tại đây:

<https://openplanning.net/10435/android-listview>

2. GridView

GridView là một viewgroup, nó hiển thị các phần tử con trên một lưới cuộn 2 chiều.



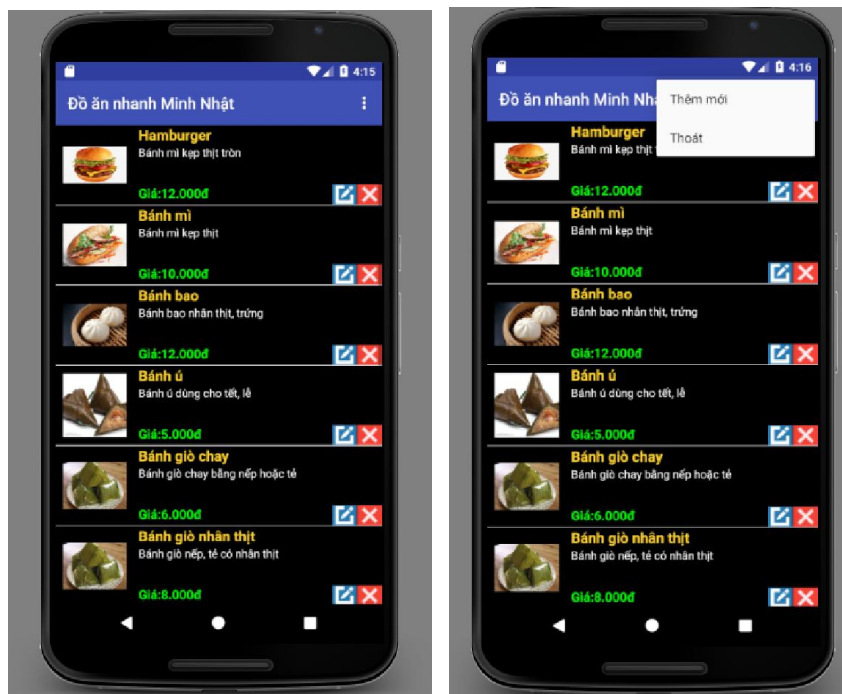
Cách làm việc của GridView cũng giống như ListView, chi tiết, các bạn xem thêm hướng dẫn tại đây:

<https://openplanning.net/10473/android-gridview>

III. Bài tập

Bài tập 1.

Sử dụng Custom Listview, xây dựng chương trình hiển thị và quản lý các món ăn như sau:



Lưu ý: Chỉ hiển thị, chưa thực hiện chức năng; Phần menu chưa cần thực hiện.

Bài tập 2:

Sử dụng Custom GridView, thiết kế chương trình như sau:

**Bài tập 3*.**

Thực hiện bài tập 1 với các chức năng Thêm, xóa, sửa

Lưu ý:

- Khi người dùng nhấn Menu, chức năng cho phép thêm một món ăn
- Người dùng có thể nhấn nút sửa, xóa món ăn trên đó.
- Sử dụng ContextMenu để thay các nút thêm, sửa trên mỗi món ăn.

--Hết--