

# Xây dựng giao diện với Fragment

---

GV: TRƯƠNG BÁ THÁI

Email: [truongbathai@tdc.edu.vn](mailto:truongbathai@tdc.edu.vn)

ĐT: 0932.577.765

# | Mục Tiêu

- Trình bày được các khái niệm cơ bản của fragment
- Xây dựng hệ thống điều hướng cho ứng dụng Android xử dụng fragment



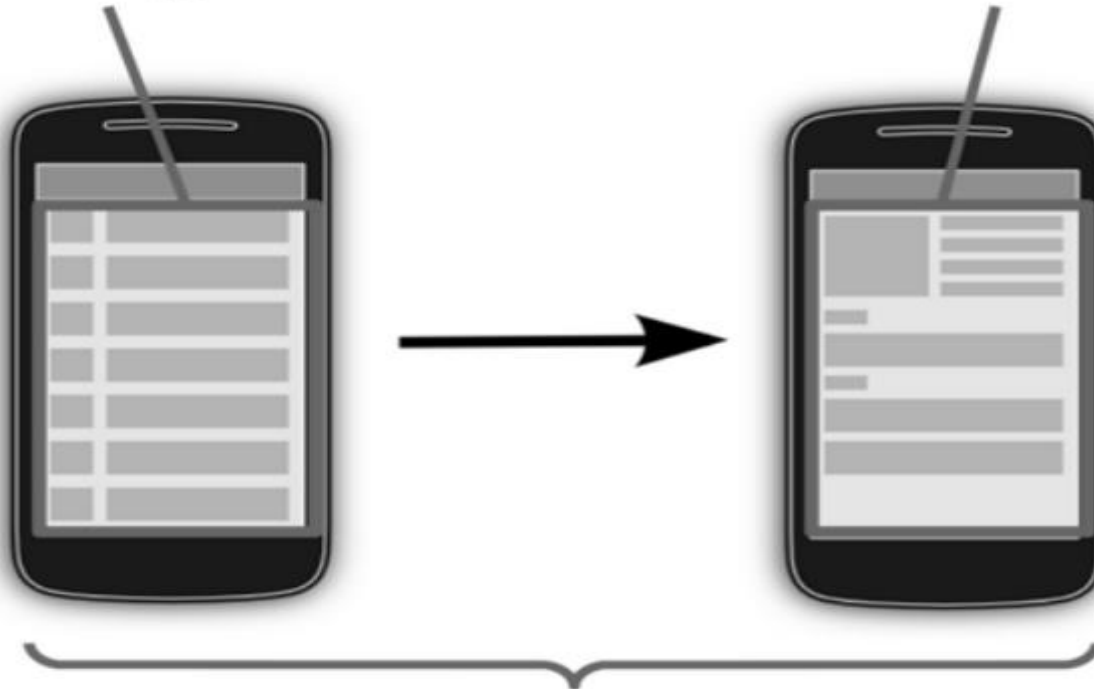
# Các khái niệm cơ bản

# | Fragment và phiên bản hỗ trợ

- Fragment được giới thiệu trong phiên bản Android 3.0, tuy nhiên do được hỗ trợ trong gói Android Support Library nên có thể sử dụng trong các phiên bản từ Android 1.6 trở lên.
- Fragment là đối tượng được nhúng trong Activity, cho phép thực hiện nhận tương tác có vòng đời riêng và thực hiện trao đổi thông tin với Activity và các Fragment khác.

ListFragment

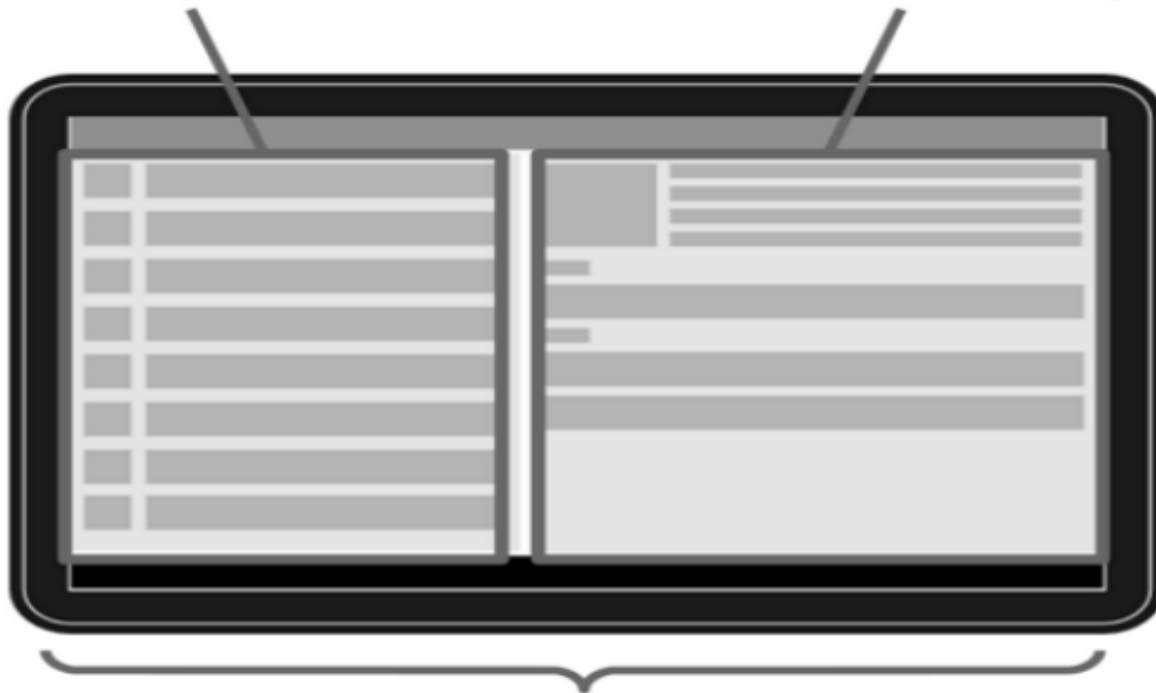
DetailsFragment



MainActivity

ListFragment

DetailsFragment

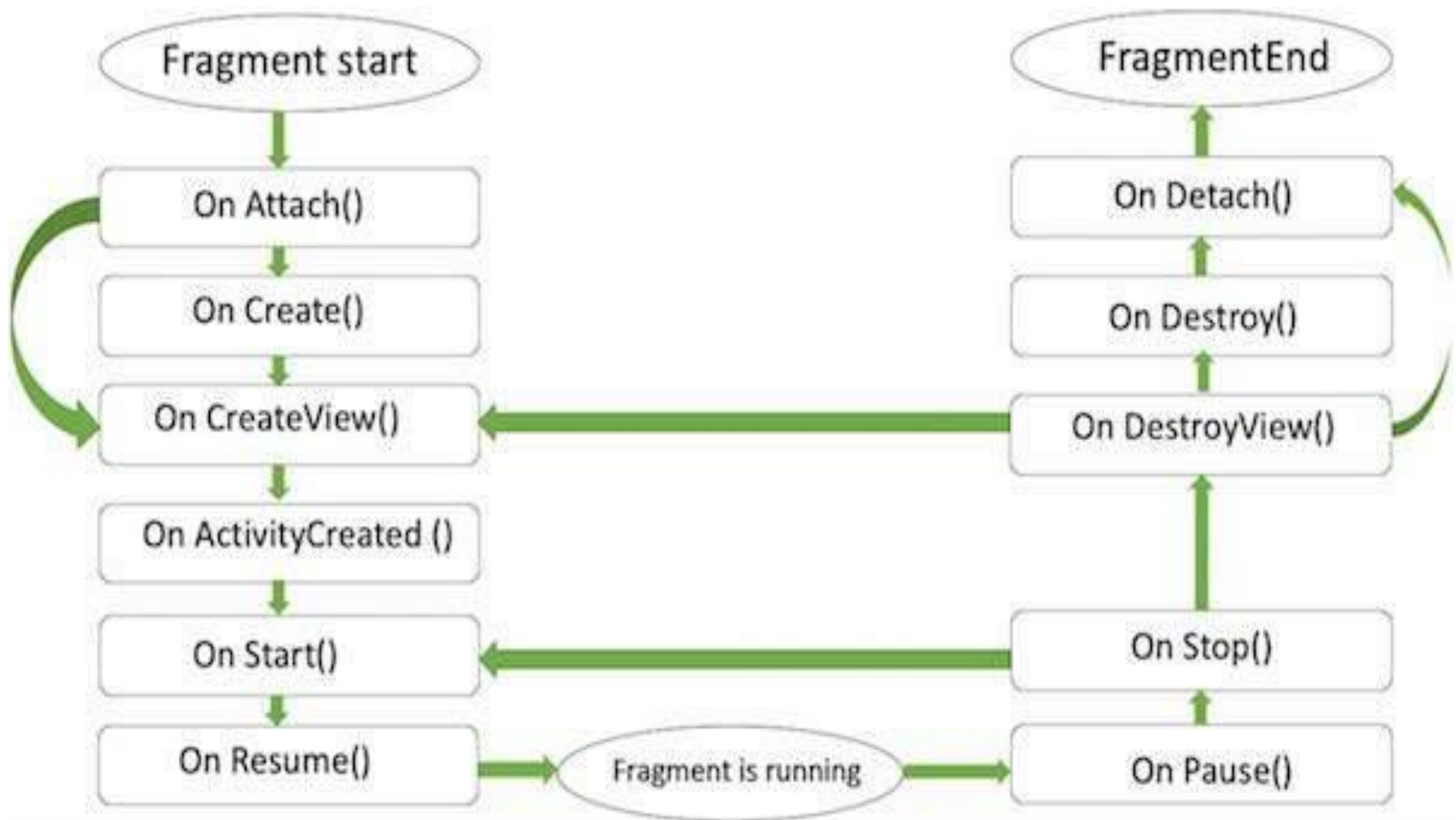


MainActivity

# | Giao diện Fragment

- Để tạo mới đối tượng Fragment ta cần tạo lớp kế thừa từ lớp Fragment, khai báo các điều khiển và thực thi các chức năng.
- Fragment sử dụng phương thức ***getActivity()*** để lấy ra Activity cha
- Fragment được định nghĩa trong file xml của activity (static definition) hoặc có thể sửa đổi fragment khi đang chạy (dynamic definition)

# Vòng đời của một Fragment





- ***onAttach()***: Sự thể hiện (instance) của Fragment được gắn kết với một sự thể hiện của activity. Fragment và Activity không hoàn toàn được khởi tạo. Đặc biệt khi lấy trong phương thức này một tham chiếu tới activity mà sử dụng Fragment cho công việc khởi tạo xa hơn.
- ***onCreate()***: Hệ thống gọi phương thức này khi tạo Fragment. nên khởi tạo các thành phần cơ bản của Fragment mà muốn duy trì khi Fragment bị dừng hoặc tạm dừng, sau đó được phục hồi lại.

- ***onCreateView()***: Hệ thống gọi phương này khi cần Fragment đó để vẽ giao diện UI lần đầu tiên. Để vẽ một UI cho Fragment của, phải trả về một thành phần View từ phương thức này. Đó là root của layout. có thể trả về null nếu Fragment không cung cấp một giao diện UI.
- ***onActivityCreated()***: Được gọi sau phương thức *onCreateView()* khi host activity được tạo. Sự thể hiện của Activity và Fragment đã được tạo cùng với cấu trúc view của activity đó. Tại điểm này, View có thể được truy cập với phương thức ***findViewById()***.

- ***onStart()***
- ***onResume()***: Fragment hoạt động.
- ***onPause()***: Hệ thống gọi phương thức này khi có dấu hiệu chỉ rằng người dùng đang rời khỏi Fragment này.
- ***onStop()***: Fragment đang bị dừng bằng cách gọi phương thức này.
- ***onDestroyView()***: Fragment view sẽ hủy sau khi gọi phương thức này.
- ***onDestroy()***: Được gọi để xóa trạng thái của Fragment.



# Xây dựng và sử dụng Fragment

# | Thực hiện xây dựng Fragment

- Khai báo lớp kế thừa từ lớp Fragment, gọi phương thức ***onCreateView*** thực hiện tạo giao diện cho Fragment.

# Ví dụ:

```
public class MyFragment1 extends Fragment {  
    @Override  
    public View onCreateView(LayoutInflater inflater, ViewGroup container,  
        Bundle savedInstanceState) {  
        super.onCreateView(inflater, container, savedInstanceState);  
        View view = inflater.inflate(R.layout.fragment1, container, false);  
        EditText txtFlagment = getActivity().findViewById(R.id.txtFlagment);  
        TextView tvFlagment = view.findViewById(R.id.tvFlagment);  
        tvFlagment.setText(txtFlagment.getText().toString());  
        return view;  
    }  
}
```

# | Sử dụng Fragment

- Thực hiện tham chiếu Fragment từ giao diện XML của Activity (***Static Fragment***).
- Khai báo đối tượng fragmentManager, cho phép nhúng Fragment vào Activity từ JavaCode.(**dynamic definition**)

# | Static Fragment

- Static Fragment là kiểu fragment được khai báo (định nghĩa) trực tiếp trong file **activity\_main.xml**



```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <fragment
        android:id="@+id/listFragment"
        class="vn.edu.tdc.chuong3vidu.MyFragment1"
        android:layout_width="match_parent"
        android:layout_height="match_parent"></fragment>
</LinearLayout>
```

```
public class MyFragment1 extends Fragment {  
    @Override  
    public View onCreateView(LayoutInflater inflater, ViewGroup container,  
        Bundle savedInstanceState) {  
        super.onCreateView(inflater, container, savedInstanceState);  
        View view = inflater.inflate(R.layout.fragment1, container, false);  
        return view;  
    }  
}
```

# Ví dụ:

- Xây dựng ứng dụng hiển thị 2 Fragment



# | Bước 1:

- Tạo một project tên là Flagment: **File → New → Android Application Project** điền các thông tin → **Next → Finish.**

## | Bước 2:

- Mở **res** → **layout** → **xml**

# | fragment1.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:background="#FF0000"
    android:layout_height="match_parent">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/tvFlagment"
        android:text="fragment 1"
        android:textSize="30dp"
        android:layout_gravity="center"
    />
</LinearLayout>
```

# | fragment2.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:background="#AA0099"
    android:layout_height="match_parent">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/tvFlagment"
        android:text="fragment 2"
        android:textSize="30dp"
        android:layout_gravity="center"
    />
</LinearLayout>
```

# | activity\_main.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="horizontal"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <fragment
        android:id="@+id/listFragment"
        class="vn.edu.tdc.chuong3vidu.MyFragment1"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        tools:layout="@layout/fragment1">
    </fragment>
```



*<fragment*

*android:id="@+id/detailFragment"*

*class="vn.edu.tdc.chuong3vidu.MyFragment2"*

*android:layout\_width="0dp"*

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

*android:layout\_weight="1"*

*tools:layout="@layout/fragment2">*

*</fragment>*

*</LinearLayout>*

## | Bước 3:

- Mở **app** → **src** -> thêm code.

# | MyFragment1.java

```
public class MyFragment1 extends Fragment {  
    @Override  
    public View onCreateView(LayoutInflater inflater, ViewGroup container,  
        Bundle savedInstanceState) {  
        super.onCreateView(inflater, container, savedInstanceState);  
        View view = inflater.inflate(R.layout.fragment1, container, false);  
        EditText  
        return view;  
    }  
}
```

# | MyFragment2.java

```
public class MyFragment1 extends Fragment {  
    @Override  
    public View onCreateView(LayoutInflater inflater, ViewGroup container,  
        Bundle savedInstanceState) {  
        super.onCreateView(inflater, container, savedInstanceState);  
        View view = inflater.inflate(R.layout.fragment1, container, false);  
        EditText  
        return view;  
    }  
}
```

# | MainActivity.java

```
public class MainActivity extends AppCompatActivity {  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
    }  
}
```



# Dynamics Fragment

# | Dynamics Fragment

- Lớp **FragmentManager** cho phép thêm, xóa, thay thế fragment trong layout của activity. Sử dụng phương thức **getFragmentManager()** hoặc **getSupportFragmentManager()** để lấy ra một đối tượng **FragmentManager**.
- Việc sửa đổi phải được thực hiện trong một giao dịch thông qua lớp **FragmentTransaction**

- Mỗi Activity sẽ có một Fragment Manager dùng để quản lý các Fragment chứa trong nó. Để sử dụng ta cần gọi phương thức `getFragmentManager`
- ***FragmentManager fm = getSupportFragmentManager();***



- Trong trường hợp ta cần thay đổi giao diện dựa trên trạng thái của ứng dụng thì cách tốt nhất ta nên sử dụng các giao diện động dựa trên **FragmentManager**.

- Mỗi phiên làm việc của FragmentTransaction được khởi tạo bằng phương thức `beginTransaction`. Các hoạt động của Fragment sẽ được định nghĩa bằng các phương thức *add*, *remove* hoặc *replace*. Sau đó chúng ta có thể tùy chỉnh các chuyển hoạt hoặc cách thức hoạt động trong stack trước khi thực hiện phương thức *commit* để thêm *Transaction* vào hàng đợi.

```
android.support.v4.app.FragmentTransaction ft_add = fm.beginTransaction();  
ft_add.add(R.id.frame_layout, new MyFragment1());  
ft_add.commit();
```

- Để thực hiện các chuyển hoạt cho các Fragment trong một Transaction ta có 2 cách, một là dùng phương thức ***setTransaction*** và truyền vào các tham số có sẵn trong lớp ***FragmentTransaction***.
- ***ft\_add.setCustomAnimations(android.R.anim.fade\_in, android.R.anim.fade\_out);***

```
public class MyFragment1 extends Fragment {
```

```
    @Override
```

```
    public View onCreateView(LayoutInflater inflater, ViewGroup container,  
    Bundle savedInstanceState) {
```

```
        super.onCreateView(inflater, container, savedInstanceState);
```

```
        View view = inflater.inflate(R.layout.fragment1, container, false);
```

```
        EditText txtFlagment=getActivity().findViewById(R.id.txtFlagment);
```

```
        TextView tvFlagment = view.findViewById(R.id.tvFlagment);
```

```
        tvFlagment.setText(txtFlagment.getText().toString());
```

```
        return view;
```

```
    }
```

```
}
```

# Ví dụ: Xây dựng ứng dụng hiển thị Fragment động

The screenshot shows an Android application window with a teal header bar containing the text "Chuong3ViDu". The main content area is a large white rectangle. At the bottom, there is a light gray bar containing a text input field with the placeholder text "nhập fragment name" and two radio buttons labeled "MyFlagment1" and "MyFlagment2".

# | Bước 1:

- Tạo một project tên là Flagment: **File → New → Android Application Project** điền các thông tin → **Next → Finish.**

# Bước 2: Mở res → layout → xml

*fragment1.xml*

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:background="#FF0000"
    android:layout_height="match_parent">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/tvFlagment"
        android:text="fragment 1"
        android:textSize="30dp"
        android:layout_gravity="center"
    />
</LinearLayout>
```



*fragment2.xml*

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:background="#AA0099"
    android:layout_height="match_parent">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/tvFlagment"
        android:text="fragment 2"
        android:textSize="30dp"
        android:layout_gravity="center"
    />
</LinearLayout>
```

activity\_main.xml

```
<LinearLayout 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:orientation="vertical">
    <FrameLayout
        android:id="@+id/frame_layout"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"></FrameLayout>
    <EditText
        android:id="@+id/txtFlagment"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="nhập flagment name" />
```

**<RadioGroup**

**android:id="@+id/radGroup"**  
**android:layout\_width="match\_parent"**  
**android:layout\_height="wrap\_content"**  
**android:gravity="center"**  
**android:orientation="horizontal">**

**<RadioButton**

**android:id="@+id/radMyFlagment1"**  
**android:layout\_width="wrap\_content"**  
**android:layout\_height="wrap\_content"**  
**android:text="MyFlagment1" />**

**<RadioButton**

**android:id="@+id/radMyFlagment2"**  
**android:layout\_width="wrap\_content"**  
**android:layout\_height="wrap\_content"**  
**android:text="MyFlagment2" />**

**</RadioGroup>**

**</LinearLayout>**

| Bước 3: Mở app → src -> thêm code.

# | MyFragment1.java

```
public class MyFragment1 extends Fragment {  
    @Override  
    public View onCreateView(LayoutInflater inflater, ViewGroup container,  
        Bundle savedInstanceState) {  
        super.onCreateView(inflater, container, savedInstanceState);  
        View view = inflater.inflate(R.layout.fragment1, container, false);  
        EditText txtFlagment=getActivity().findViewById(R.id.txtFlagment);  
        TextView tvFlagment = view.findViewById(R.id.tvFlagment);  
        tvFlagment.setText(txtFlagment.getText().toString());  
  
        return view;  
    }  
}
```

# | MyFragment2.java

```
public class MyFragment1 extends Fragment {  
    @Override  
    public View onCreateView(LayoutInflater inflater, ViewGroup container,  
        Bundle savedInstanceState) {  
        super.onCreateView(inflater, container, savedInstanceState);  
        View view = inflater.inflate(R.layout.fragment1, container, false);  
        EditText txtFlagment=getActivity().findViewById(R.id.txtFlagment);  
        TextView tvFlagment = view.findViewById(R.id.tvFlagment);  
        tvFlagment.setText(txtFlagment.getText().toString());  
        return view;  
    }  
}
```

# MainActivity.java

*MainActivity.java*

```
public class MainActivity extends AppCompatActivity {  
    //Khai báo các widget  
    RadioGroup radioGroup;  
    FragmentManager fm;  
    FragmentTransaction ft_add;  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        setConTrol();  
        setEvent();  
    }  
    private void setConTrol() {  
        radioGroup = findViewById(R.id.radGroup);  
    }
```

```

private void setEvent() {
    fm = getSupportFragmentManager();
    radioGroup.setOnCheckedChangeListener(new
RadioGroup.OnCheckedChangeListener() {
        @Override
        public void onCheckedChanged(RadioGroup group, int checkedId) {
            if (checkedId == R.id.radMyFlagment1) {
                ft_add = fm.beginTransaction();
                ft_add.setCustomAnimations(android.R.animator.fade_in,
android.R.animator.fade_out);
                ft_add.replace(R.id.frame_layout, new MyFragment1());
                ft_add.addToBackStack(null);
                ft_add.commit();
            }
        }
    }
}

```



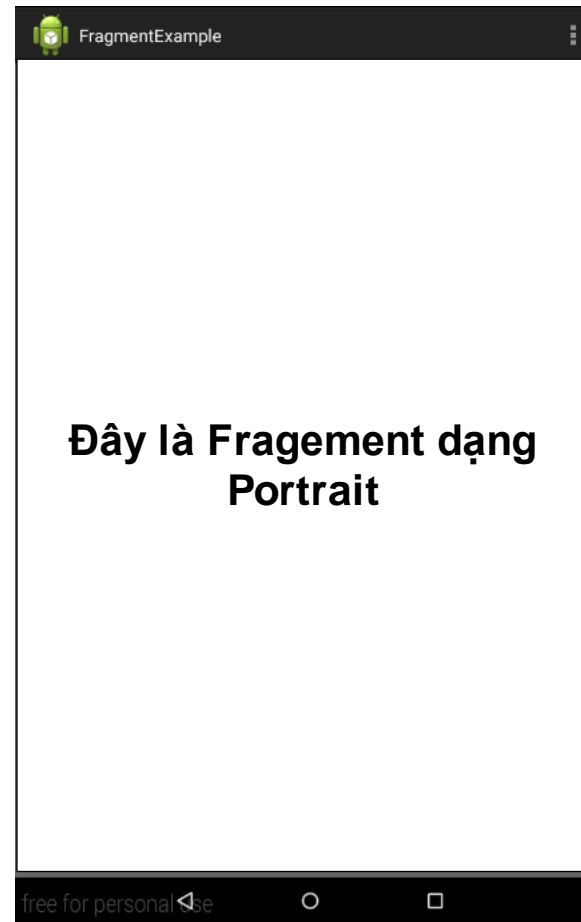
```
if (checkedId == R.id.radMyFlagment2) {  
    ft_add = fm.beginTransaction();  
    ft_add.replace(R.id.frame_layout, new MyFragment2());  
    ft_add.addToBackStack(null);  
    ft_add.commit();  
}  
}  
});  
}  
}
```



# Bài tập Chương 3

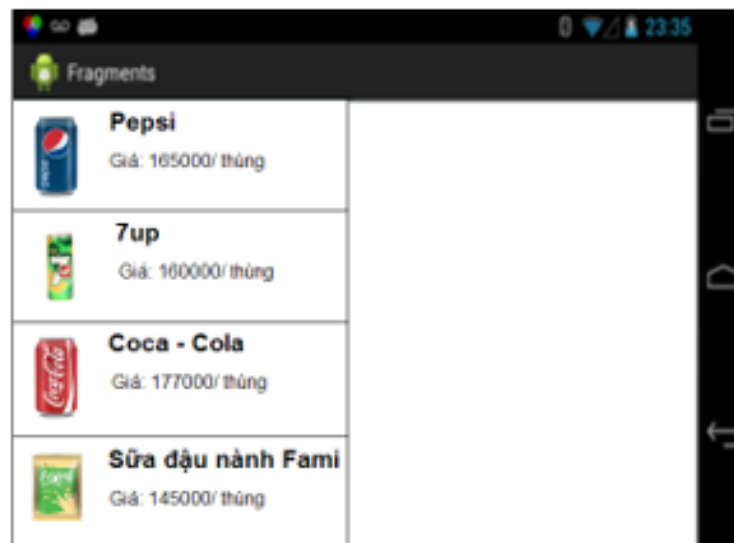
# Bài 1: Tạo và sử dụng Fragment

- Tạo 2 Fragment:
  - Fragment 1 được dùng khi màn hình thiết bị nằm ngang.
  - Fragment 2 được dùng khi màn hình thiết bị nằm dọc.
- Yêu cầu:
  - Khi xoay màn hình thiết bị nằm ngang sẽ hiển thị giao diện như hình 1, còn khi xoay màn hình thiết bị nằm dọc sẽ hiển thị giao diện như hình 2.

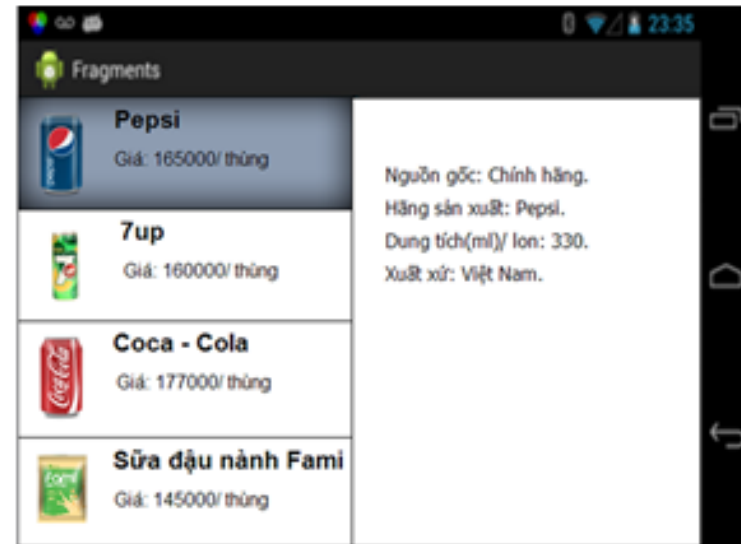


# Bài 2: Tạo và sử dụng Fragment

- Sử dụng Fragment khi người dùng nhấn chọn vào 1 dòng trên ListView sẽ hiển thị nội dung của dòng đó



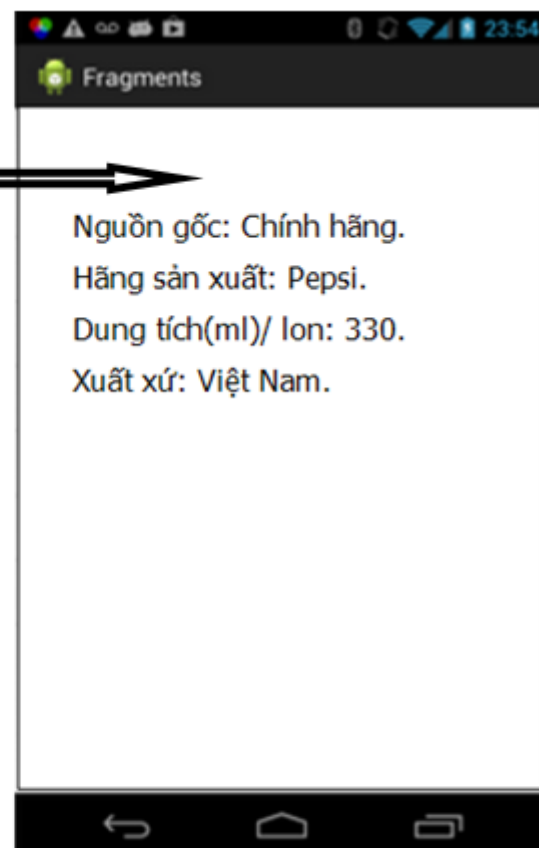
*Giao diện khi thiết bị nằm ngang*



*Giao diện khi người dùng chọn 1 dòng trên ListView trên thiết bị nằm ngang.*



*Giao diện khi thiết bị nằm dọc.*



*Giao diện khi người dùng chọn 1 dòng trên ListView khi thiết bị nằm dọc.*





**CẢM ƠN TẤT CẢ  
ĐÃ LẮNG NGHE**