HUST

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

ONE LOVE. ONE FUTURE.

TRƯỜNG ĐẠI HỌC
BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY
OF SCIENCE AND TECHNOLOGY

LẬP TRÌNH ỨNG DỤNG DI ĐỘNG

Mobile Application Programming

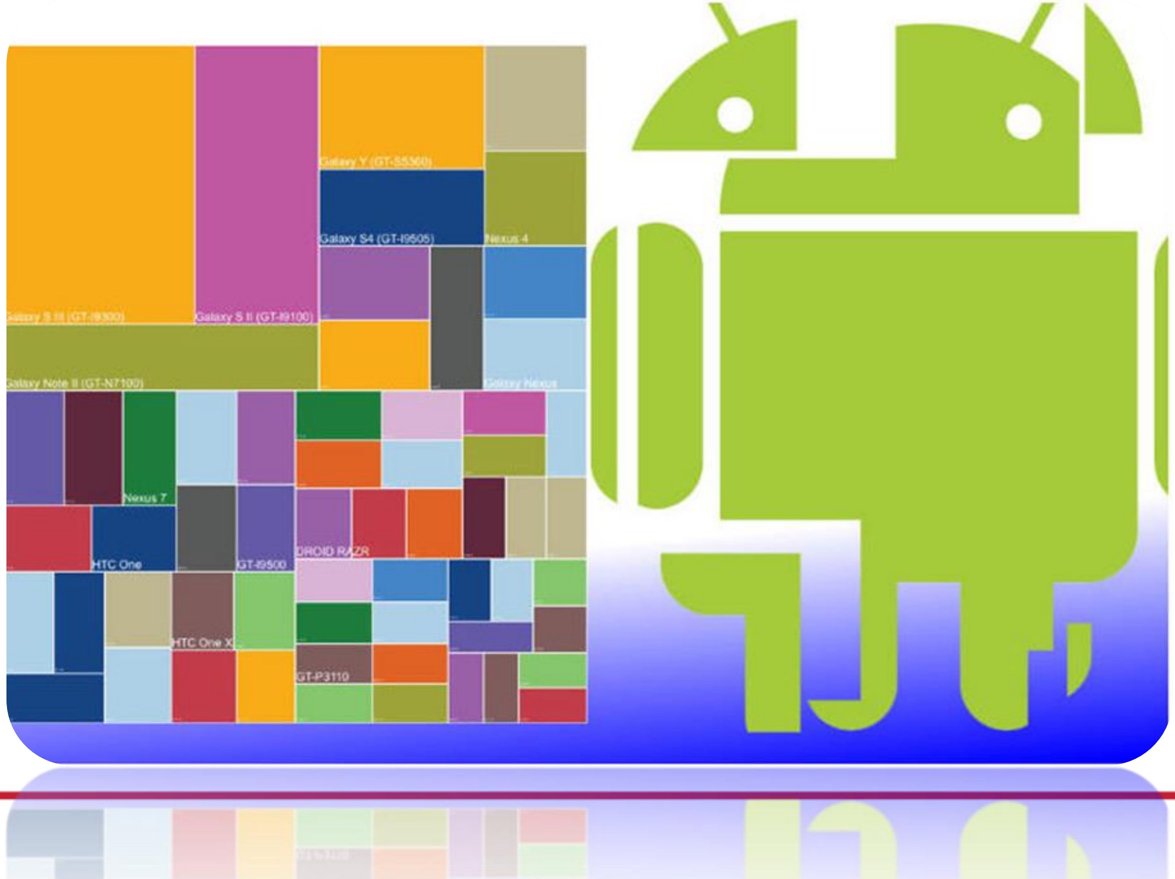
ET4710

PGS. TS. Đỗ Trọng Tuấn
Viện Điện tử Viễn thông * Đại học Bách Khoa Hà Nội

ONE LOVE. ONE FUTURE.

CHƯƠNG 3.

Lập trình giao diện người dùng với UI Fragment (UI Fragment and Fragment Manager)



Chương 3

Lập trình giao diện người dùng với UI Fragment
(UI Fragment and Fragment Manager)

3.1 Giới thiệu về fragment (Introduction to fragment)

3.2 Tạo UI Fragment (Creating UI Fragment)

3.3 Quản lý fragment với Fragment Manager

(Fragment management with FragmentManager)

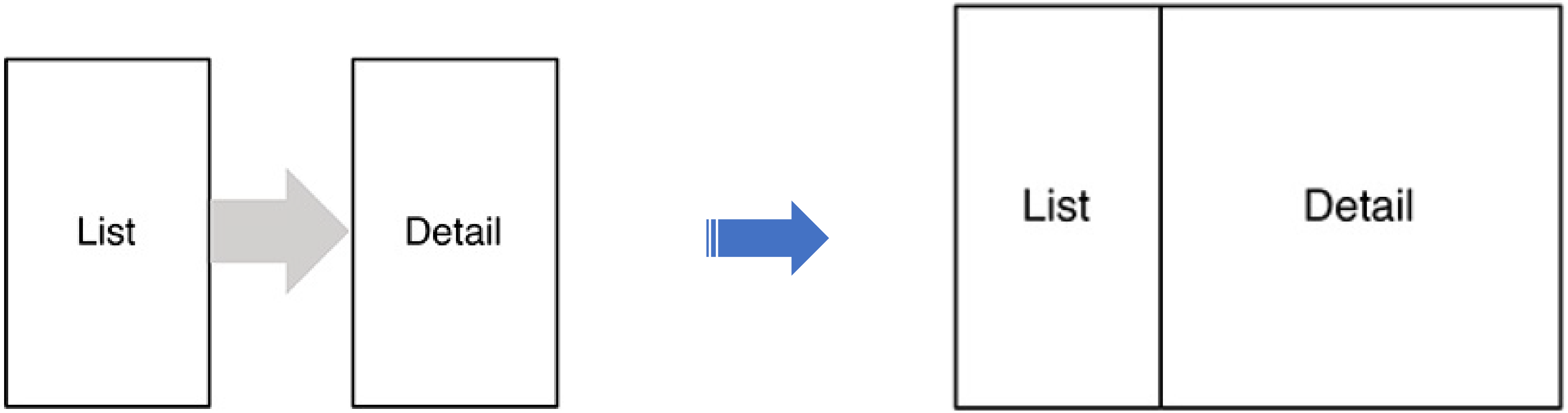
3.4 Thiết kế fragment có tham số (Fragment with arguements)

3.1 Giới thiệu về fragment (Introduction to fragment)

UI flexibility

Phone

Tablet



List

Detail

List

Detail

3.1 Giới thiệu về fragment (Introduction to fragment)

UI flexibility

- The ability to compose and recompose an activity’s view at runtime depending on what the user or the device requires.
- Activities were not built to provide this flexibility.
- An activity’s views may change at runtime, but the code to control those views must live inside the activity.
- Activities are tightly coupled to the particular screen being used.

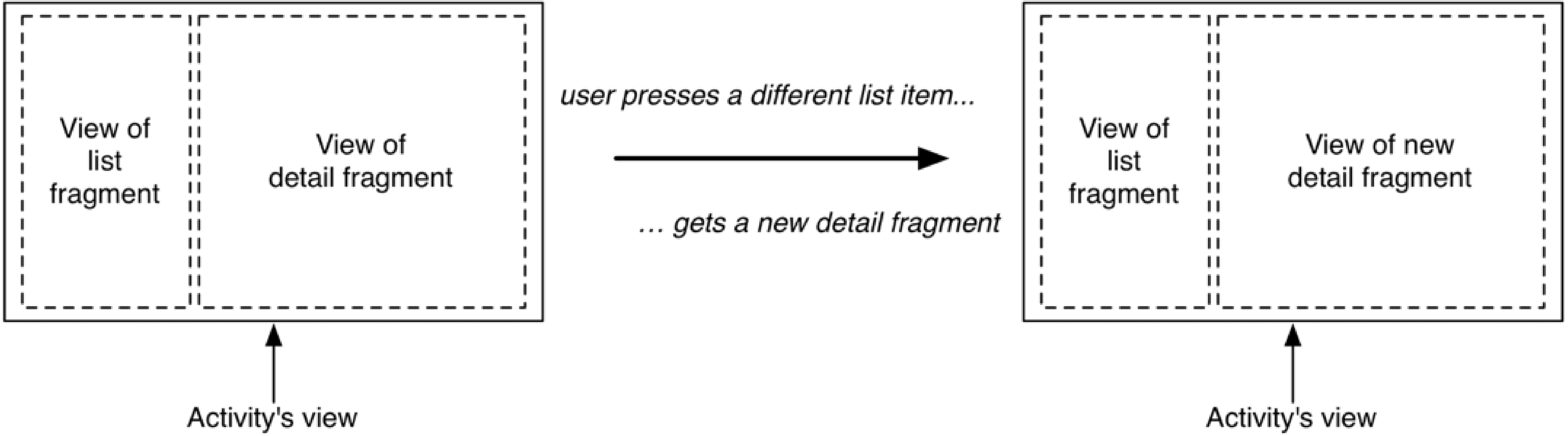
3.1 Giới thiệu về fragment (Introduction to fragment)

What is Android Fragment?

- A fragment is a controller object that an activity can deputize to perform tasks. It is a Graphical User Interface component of Android.
- Resides within the Activities of an Android application.
- Represents a portion of UI (Sub-Activity) that the user sees on the screen.
- Android Fragments cannot exist outside an activity.
- Fragments improve the adaptability & user experience by making the UI flexible for all devices

3.1 Giới thiệu về fragment (Introduction to fragment)

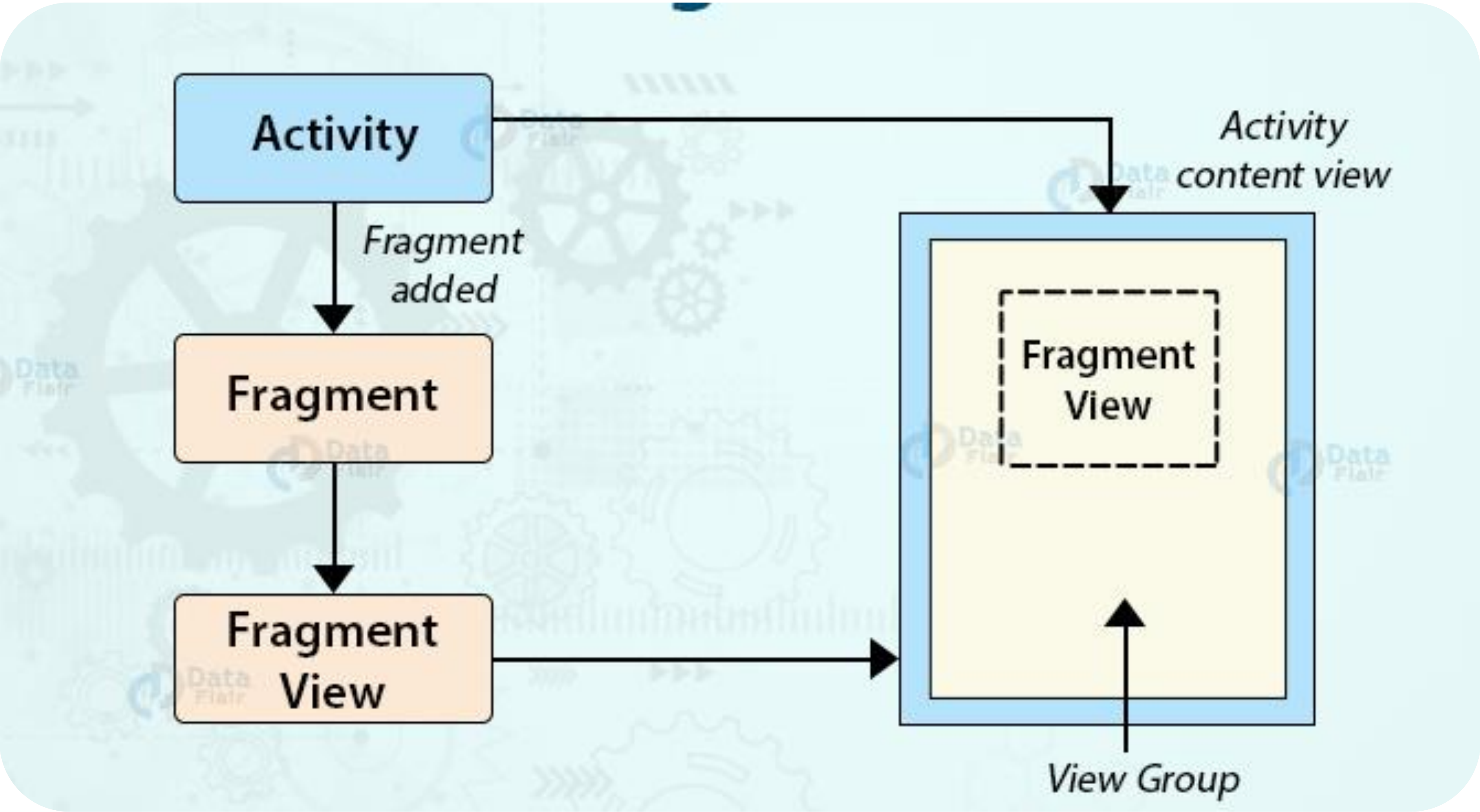
What is Android Fragment?



- Using UI fragments separates the UI of your app into building blocks, which is useful for more than just list-detail applications.
- Working with individual blocks, it is easy to build tab interfaces, tack on animated sidebars, and more.

3.1 Giới thiệu về fragment (Introduction to fragment)

Android Fragment



Types of fragments

- Native fragments and Support fragments
- The native implementation of fragments is built into the device that the user runs your app on.
- The support implementation of fragments is built into a library that you include in your application.
- With support fragments, each device you run your app on will depend on the same implementation of fragments no matter the Android version.

3.1 Giới thiệu về fragment (Introduction to fragment)

Hosting a UI Fragment

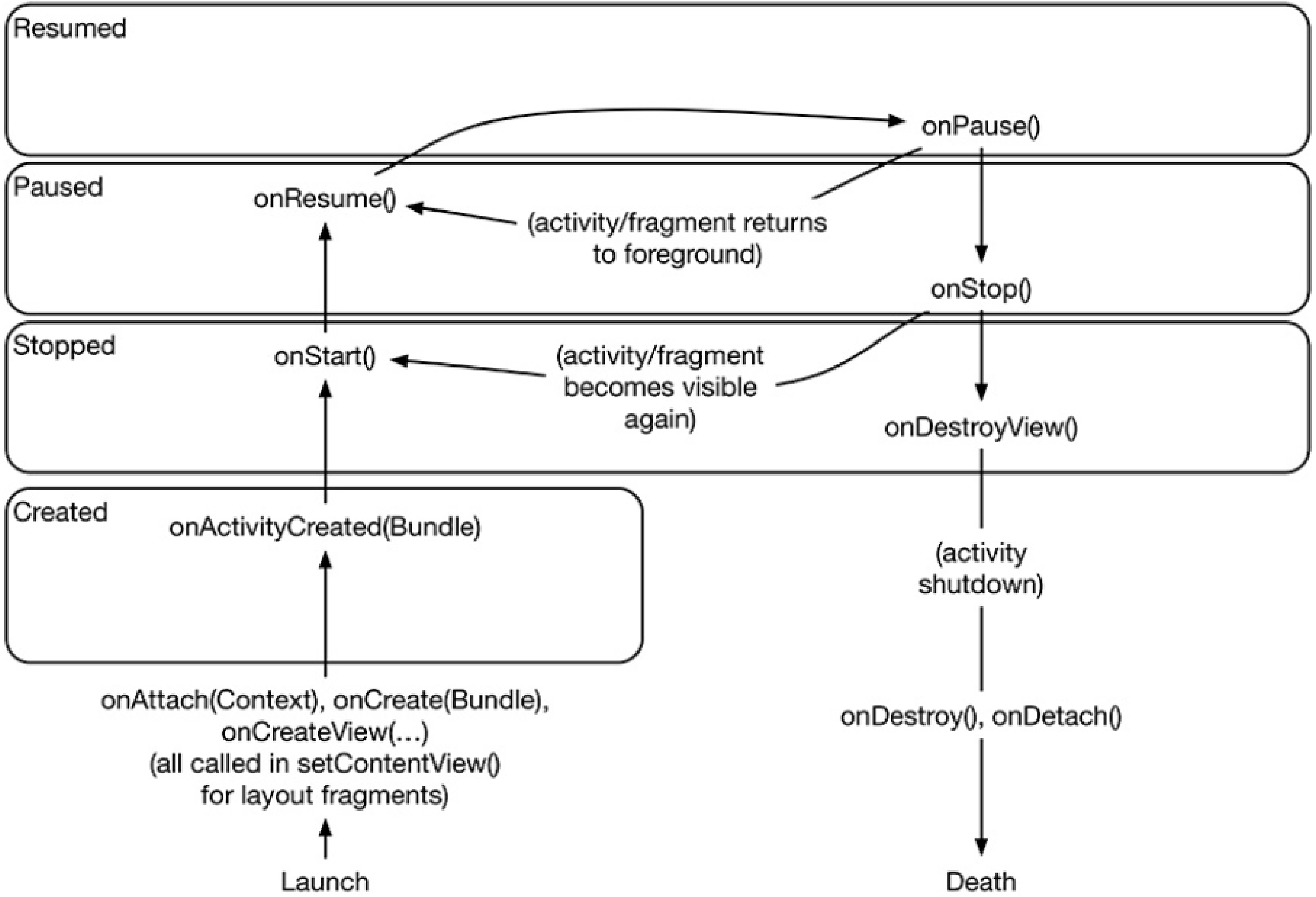
- To host a UI fragment, an activity must:
  - ✓ Define a spot in its layout for the fragment’s view
  - ✓ Manage the lifecycle of the fragment instance
- Two options for hosting a UI fragment in an activity:
  - ✓ Add the fragment to the activity’s layout
  - ✓ Add the fragment in the activity’s code

To achieve real UI flexibility, you must add your fragment in code which is more complex, but it is the only way to have control at runtime over your fragments.

3.1 Giới thiệu về fragment (Introduction to fragment)

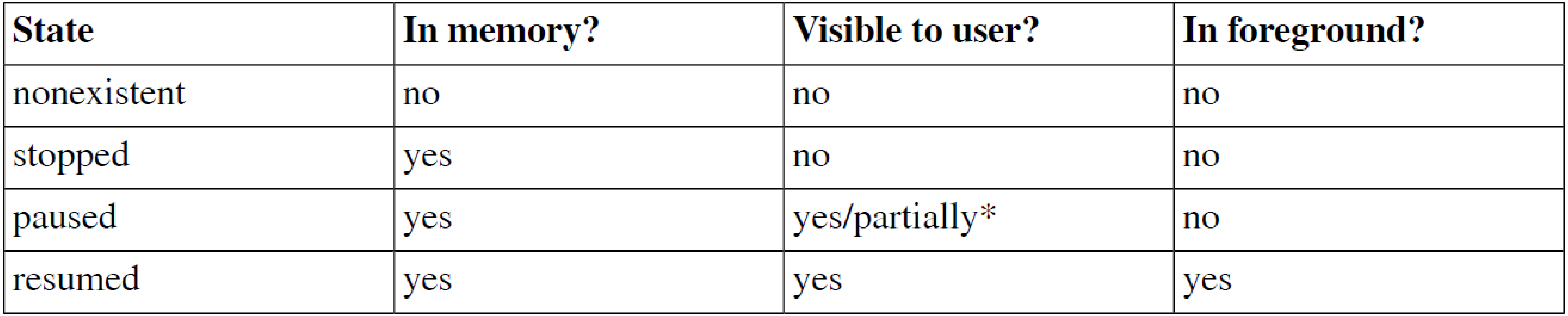
Fragment lifecycle diagram

Fragment lifecycle is similar to the activity lifecycle: It has stopped, paused, and resumed states,


Resumed
onPause()
Paused
onResume()
(activity/fragment returns to foreground)
onStop()
Stopped
onStart()
(activity/fragment becomes visible again)
onDestroyView()
Created
onActivityCreated(Bundle)
(activity shutdown)
onAttach(Context), onCreate(Bundle), onCreateView(...)
(all called in setContentView() for layout fragments)
onDestroy(), onDetach()
Launch
Death

3.1 Giới thiệu về fragment (Introduction to fragment)

Activity States

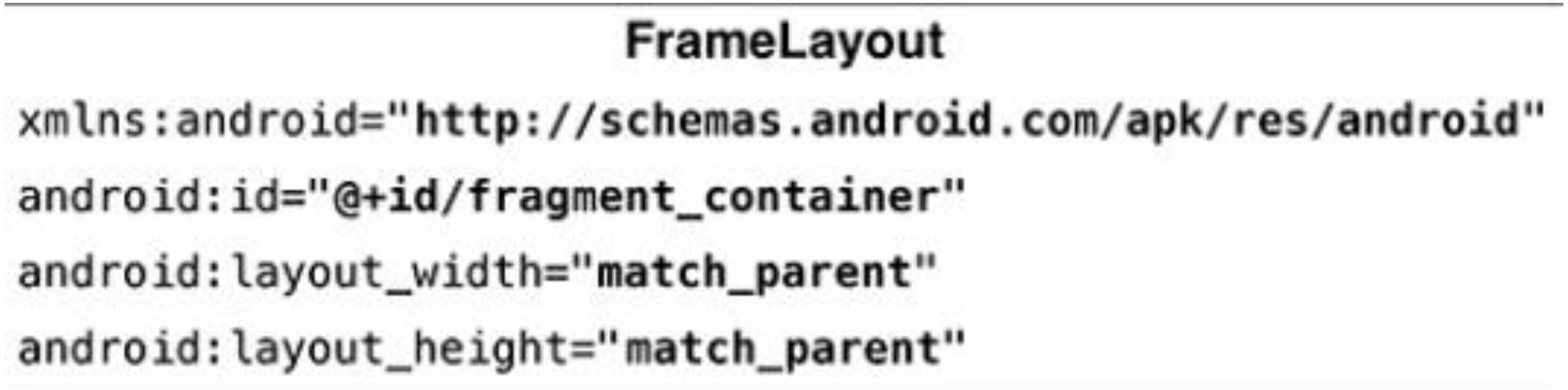


State	In memory?	Visible to user?	In foreground?
nonexistent	no	no	no
stopped	yes	no	no
paused	yes	yes/partially*	no
resumed	yes	yes	yes

3.1 Giới thiệu về fragment (Introduction to fragment)

Defining a container view

- Make a spot for the fragment’s view in the activity’s view hierarchy.
- The spot would be the FrameLayout shown below



```
FrameLayout
xmlns:android="http://schemas.android.com/apk/res/android"
android:id="@+id/fragment_container"
android:layout_width="match_parent"
android:layout_height="match_parent"
```

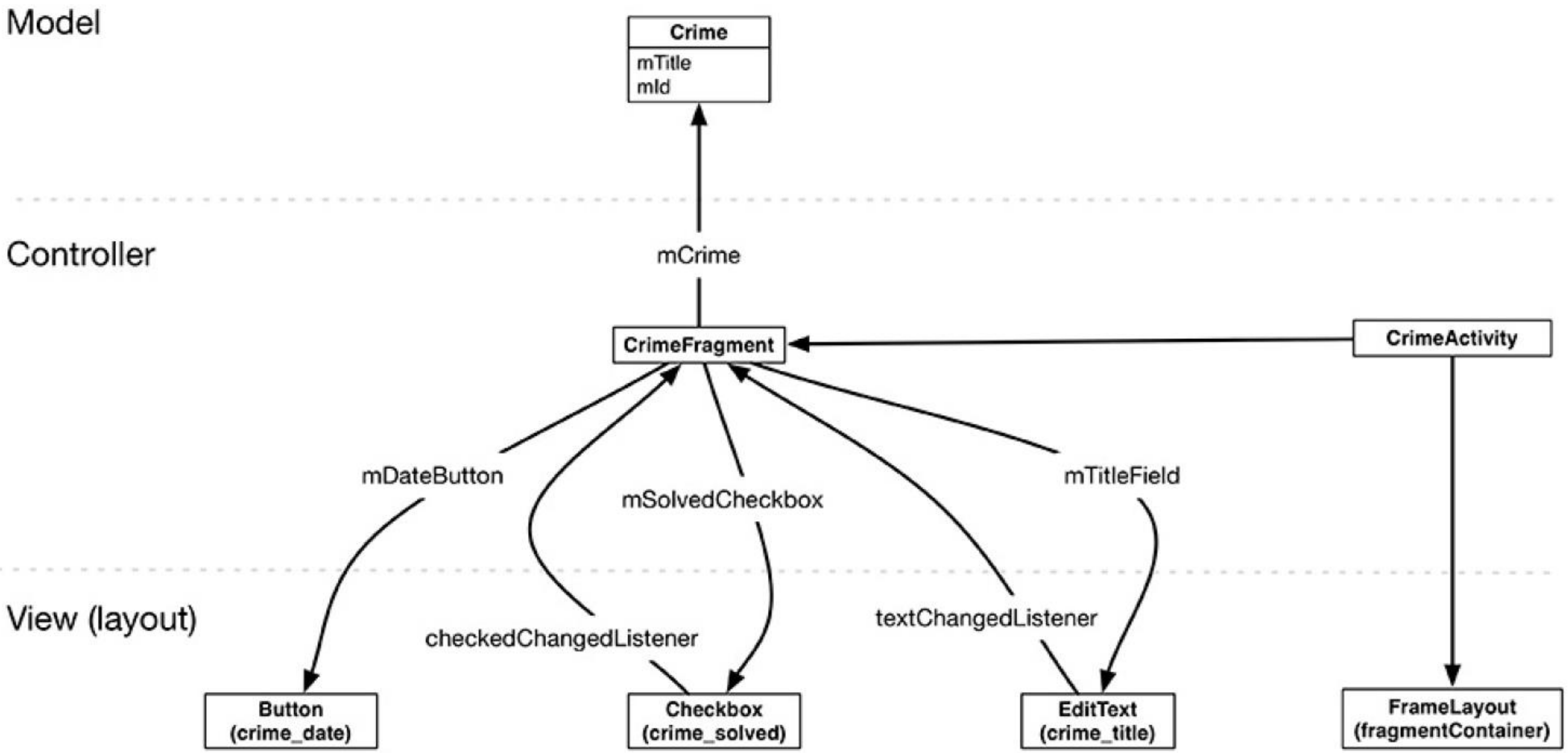
- This FrameLayout will be the container view for a Fragment.
- An activity’s layout can be more complex and define multiple container views as well as widgets of its own.

3.2 Tạo UI Fragment (Creating UI Fragment)

Creating a UI Fragment

- The steps to create a UI fragment are the same as those you followed to create an activity:
  - ✓ Compose a UI by defining widgets in a layout file
  - ✓ Create the class and set its view to be the layout that you defined
  - ✓ Wire up the widgets inflated from the layout in code

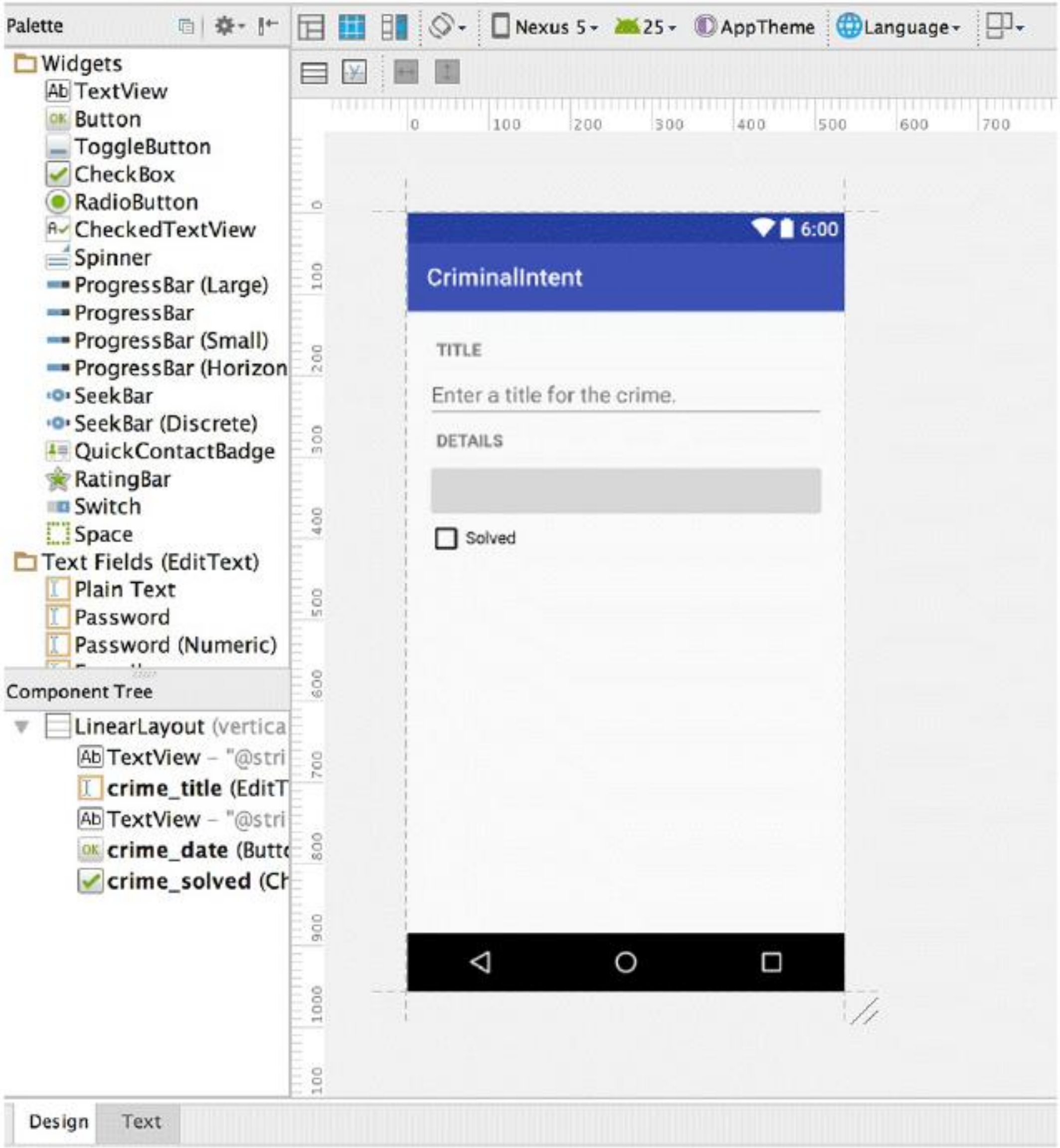
3.2 Tạo UI Fragment (Creating UI Fragment)



3.2 Tạo UI Fragment (Creating UI Fragment)

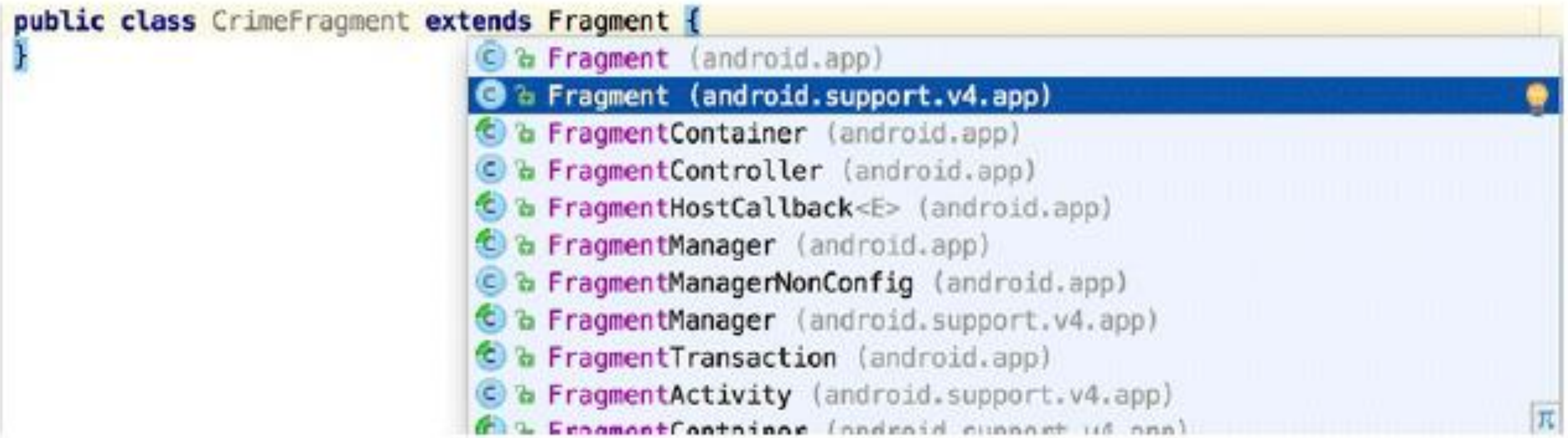
Creating a UI Fragment

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="16dp"
    android:orientation="vertical">

    <TextView
        style="?android:listSeparatorTextViewStyle"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/crime_title_label"/>

    <EditText
        android:id="@+id/crime_title"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/crime_title_hint"/>

    <TextView
        style="?android:listSeparatorTextViewStyle"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/crime_details_label"/>

    <Button
        android:id="@+id/crime_date"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>

    <CheckBox
        android:id="@+id/crime_solved"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/crime_solved_label"/>

</LinearLayout>
```



3.2 Tạo UI Fragment (Creating UI Fragment)

Creating the Fragment class



```
public class CrimeFragment extends Fragment {
}
```

Fragment (android.app)
Fragment (android.support.v4.app)
FragmentContainer (android.app)
FragmentController (android.app)
FragmentHostCallback<E> (android.app)
FragmentManager (android.app)
FragmentManagerNonConfig (android.app)
FragmentManager (android.support.v4.app)
FragmentTransaction (android.app)
FragmentActivity (android.support.v4.app)
FragmentContainer (android.support.v4.app)

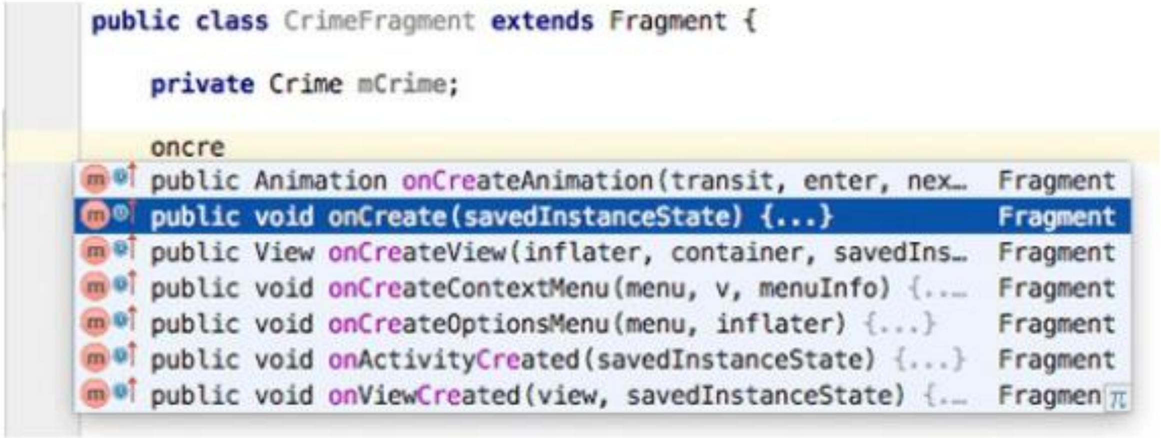
Listing 7.9 Supporting the **Fragment** import (CrimeFragment.java)

```
package com.bignerdranch.android.criminalintent;

import android.support.v4.app.Fragment;

public class CrimeFragment extends Fragment {

}
```

3.2 Tạo UI Fragment (Creating UI Fragment)

Implementing fragment lifecycle methods

Figure 7.16 Overriding the **onCreate(Bundle)** method



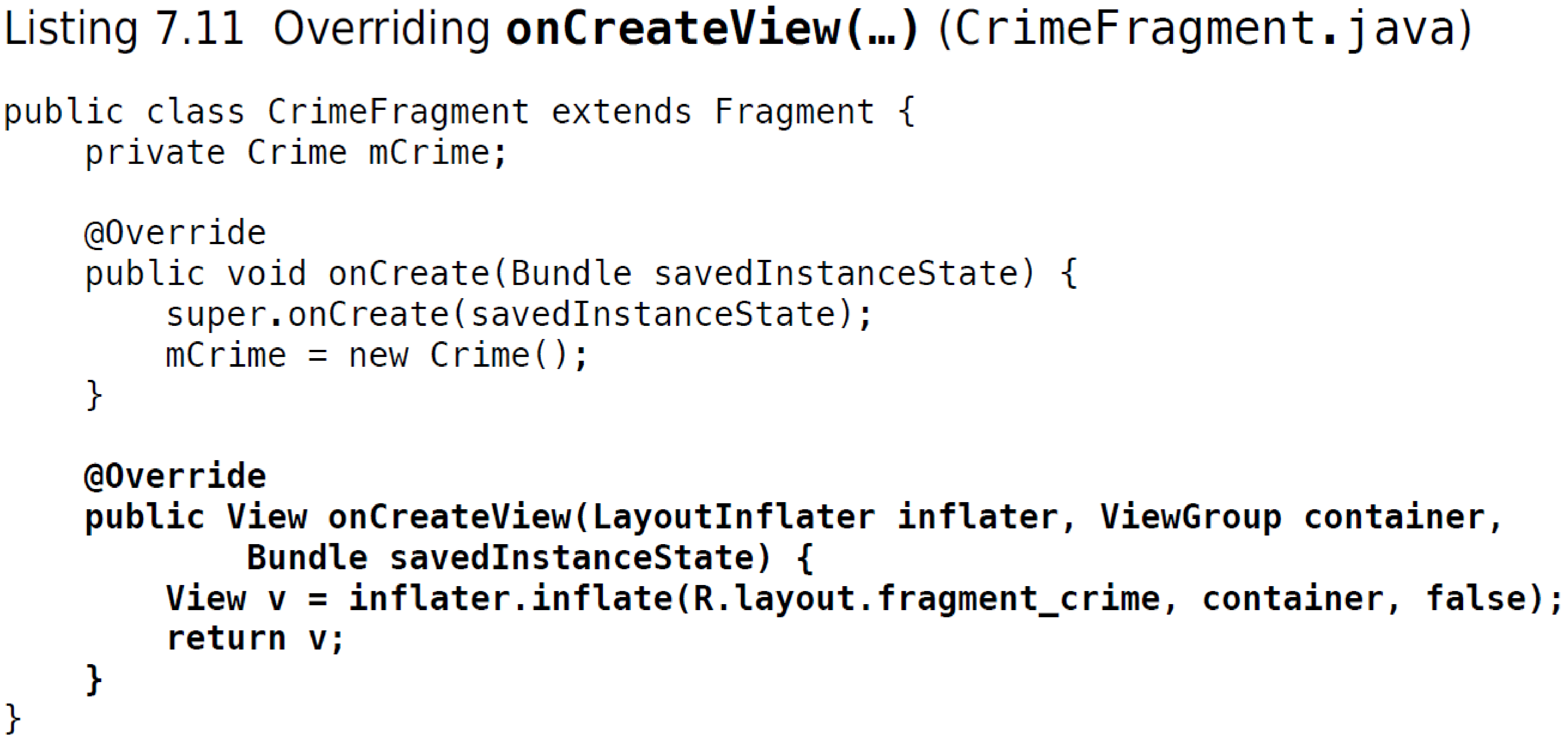
```
public class CrimeFragment extends Fragment {

    private Crime mCrime;

    oncre
```

```
public Animation onCreateAnimation(transit, enter, nex…   Fragment
public void onCreate(savedInstanceState) {...}            Fragment
public View onCreateView(inflater, container, savedIns…   Fragment
public void onCreateContextMenu(menu, v, menuInfo) {...   Fragment
public void onCreateOptionsMenu(menu, inflater) {...}     Fragment
public void onActivityCreated(savedInstanceState) {...}   Fragment
public void onViewCreated(view, savedInstanceState) {...  Fragmen
```

```
public class CrimeFragment extends Fragment {
    private Crime mCrime;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        mCrime = new Crime();
    }
}
```

3.2 Tạo UI Fragment (Creating UI Fragment)

Implementing fragment lifecycle methods

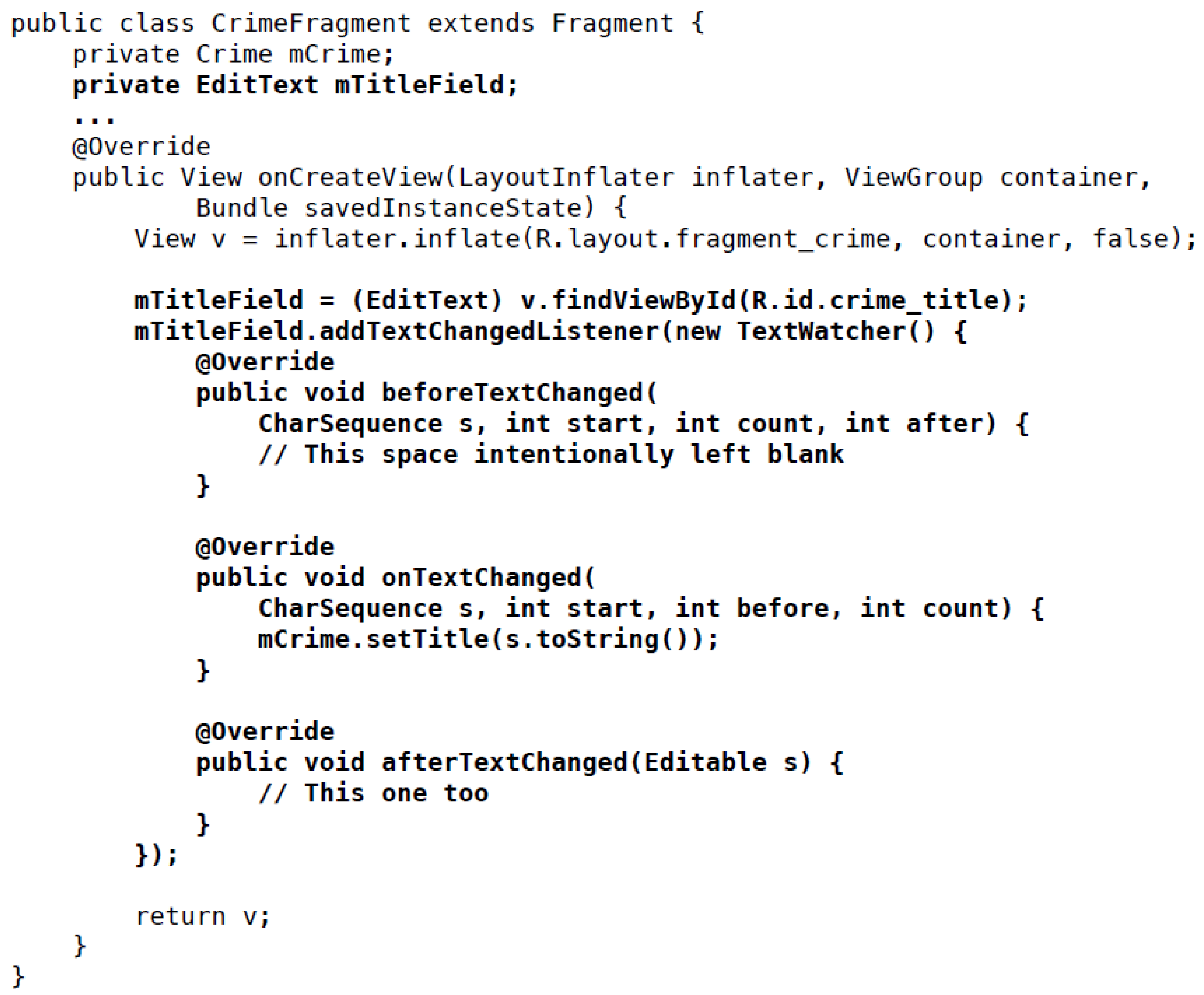


Listing 7.11 Overriding **onCreateView(…)** (CrimeFragment.java)

```
public class CrimeFragment extends Fragment {
    private Crime mCrime;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        mCrime = new Crime();
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
            Bundle savedInstanceState) {
        View v = inflater.inflate(R.layout.fragment_crime, container, false);
        return v;
    }
}
```

3.2 Tạo UI Fragment (Creating UI Fragment)

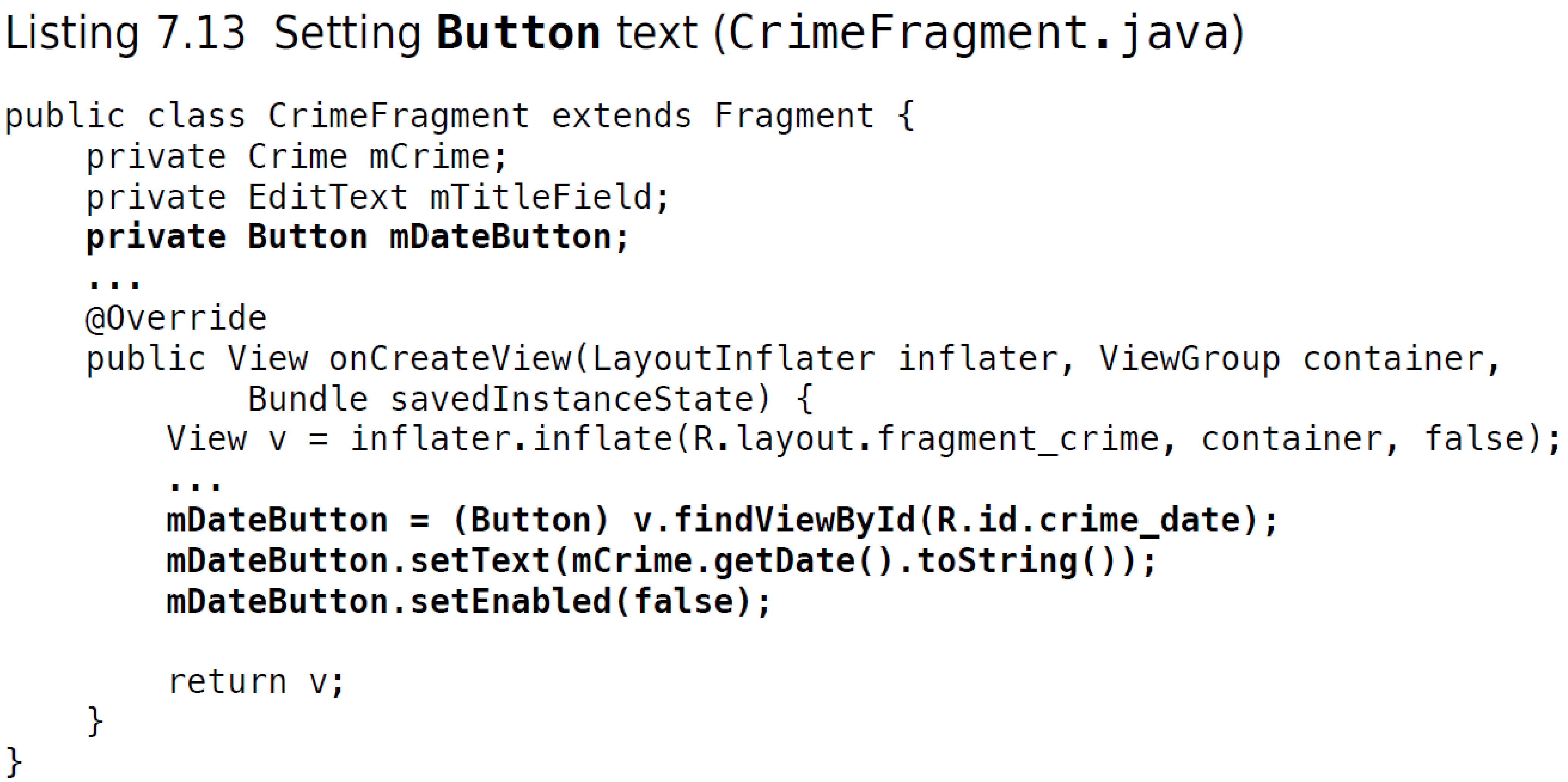
Wiring widgets in a fragment



```
public class CrimeFragment extends Fragment {
    private Crime mCrime;
    private EditText mTitleField;
    ...
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
            Bundle savedInstanceState) {
        View v = inflater.inflate(R.layout.fragment_crime, container, false);

        mTitleField = (EditText) v.findViewById(R.id.crime_title);
        mTitleField.addTextChangedListener(new TextWatcher() {
            @Override
            public void beforeTextChanged(
                CharSequence s, int start, int count, int after) {
                // This space intentionally left blank
            }

            @Override
            public void onTextChanged(
                CharSequence s, int start, int before, int count) {
                mCrime.setTitle(s.toString());
            }

            @Override
            public void afterTextChanged(Editable s) {
                // This one too
            }
        });

        return v;
    }
}
```

3.2 Tạo UI Fragment (Creating UI Fragment)

Wiring widgets in a fragment

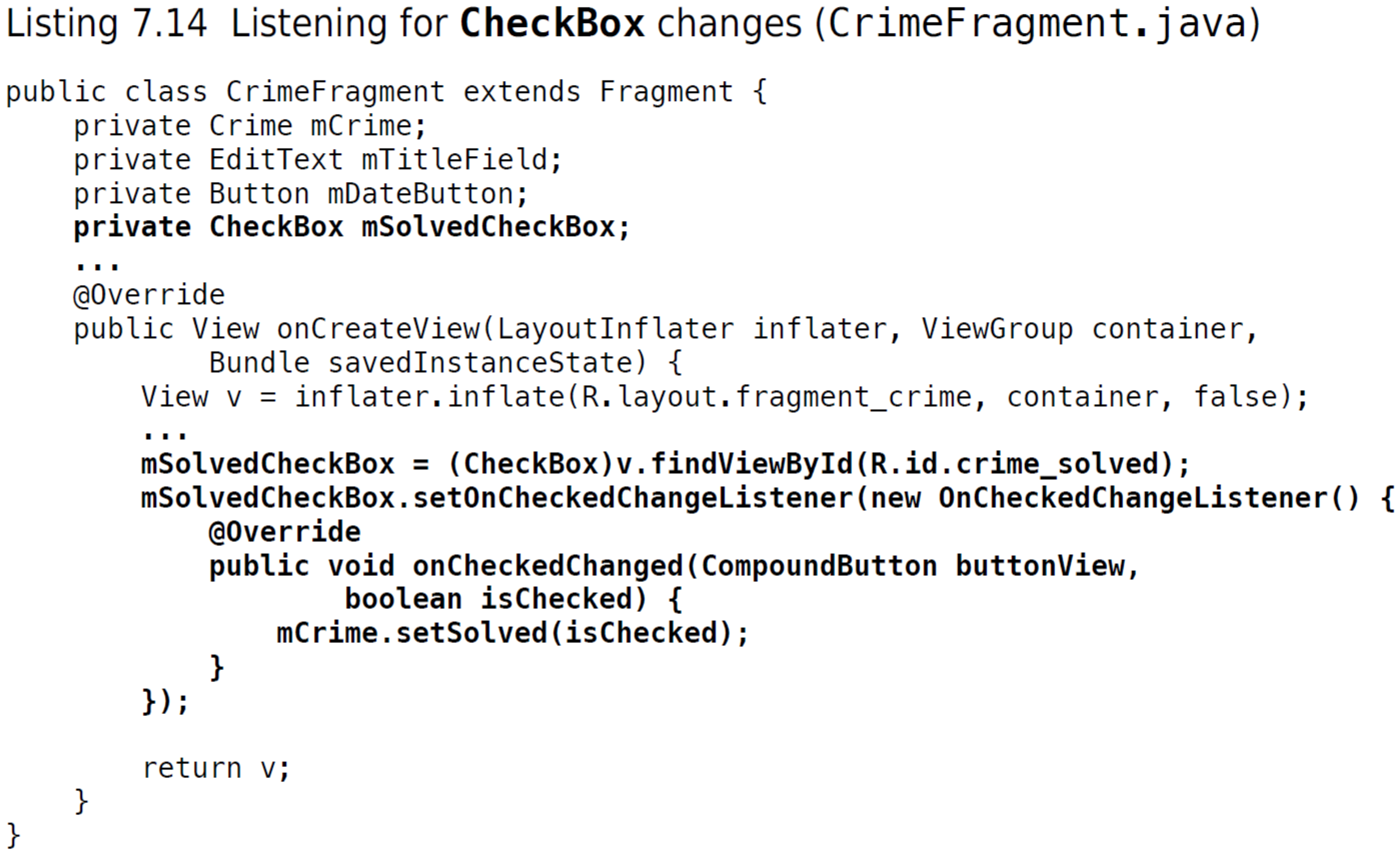


Listing 7.13 Setting **Button** text (CrimeFragment.java)

```
public class CrimeFragment extends Fragment {
    private Crime mCrime;
    private EditText mTitleField;
    private Button mDateButton;
    ...
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
            Bundle savedInstanceState) {
        View v = inflater.inflate(R.layout.fragment_crime, container, false);
        ...
        mDateButton = (Button) v.findViewById(R.id.crime_date);
        mDateButton.setText(mCrime.getDate().toString());
        mDateButton.setEnabled(false);

        return v;
    }
}
```

3.2 Tạo UI Fragment (Creating UI Fragment)

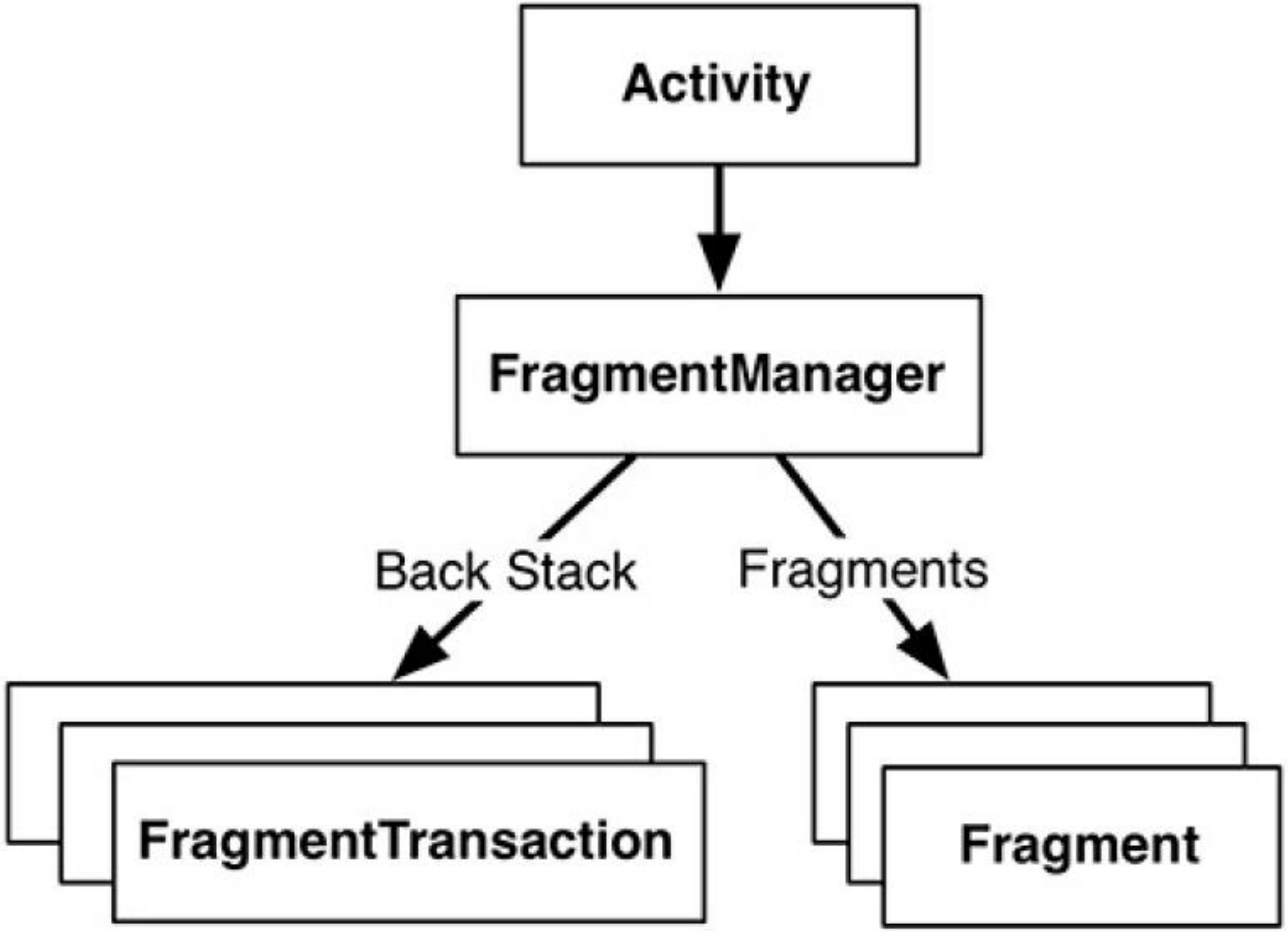
Wiring widgets in a fragment



Listing 7.14 Listening for **CheckBox** changes (CrimeFragment.java)

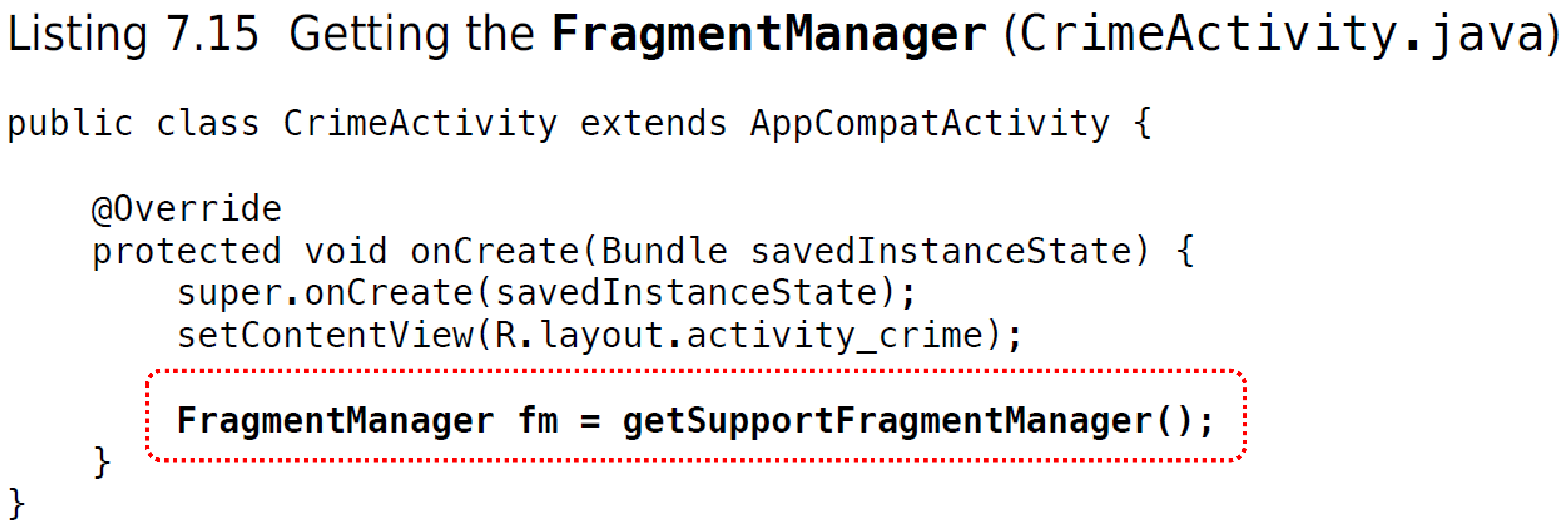
```
public class CrimeFragment extends Fragment {
    private Crime mCrime;
    private EditText mTitleField;
    private Button mDateButton;
    private CheckBox mSolvedCheckBox;
    ...
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
            Bundle savedInstanceState) {
        View v = inflater.inflate(R.layout.fragment_crime, container, false);
        ...
        mSolvedCheckBox = (CheckBox)v.findViewById(R.id.crime_solved);
        mSolvedCheckBox.setOnCheckedChangeListener(new OnCheckedChangeListener() {
            @Override
            public void onCheckedChanged(CompoundButton buttonView,
                    boolean isChecked) {
                mCrime.setSolved(isChecked);
            }
        });

        return v;
    }
}
```

3.3. Quản lý fragment với Fragment Manager



3.3 Quản lý fragment với Fragment Manager

Adding a UI Fragment to the FragmentManager

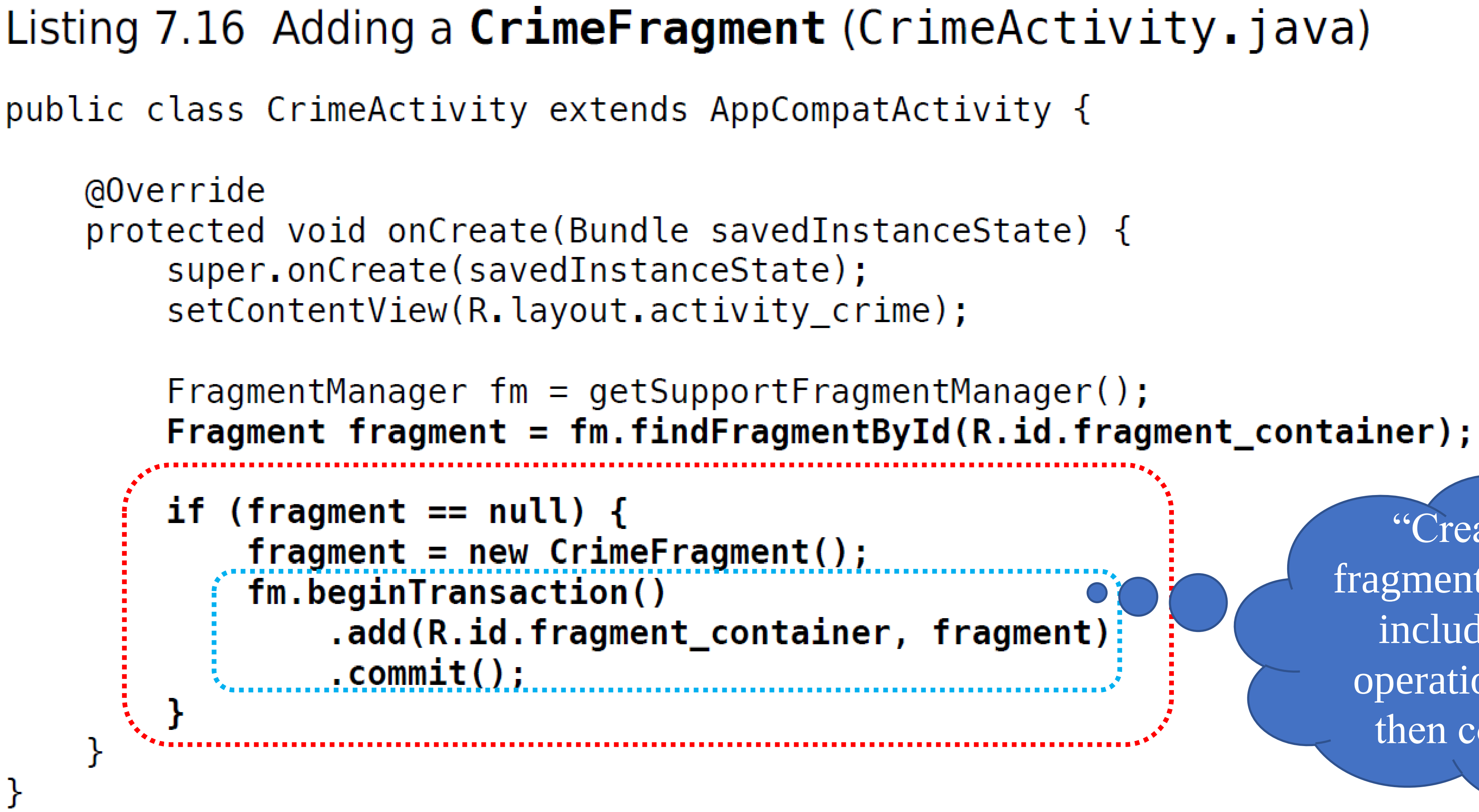


Listing 7.15 Getting the **FragmentManager** (CrimeActivity.java)

```
public class CrimeActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_crime);

        FragmentManager fm = getSupportFragmentManager();
    }
}
```

3.3 Quản lý fragment với Fragment Manager

Fragment transactions



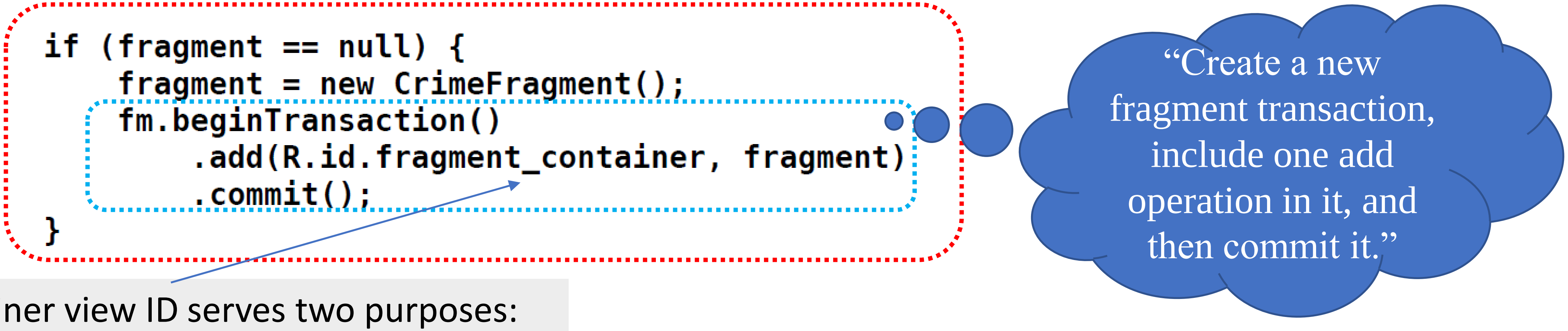
Listing 7.16 Adding a **CrimeFragment** (CrimeActivity.java)

```
public class CrimeActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_crime);

        FragmentManager fm = getSupportFragmentManager();
        Fragment fragment = fm.findFragmentById(R.id.fragment_container);

        if (fragment == null) {
            fragment = new CrimeFragment();
            fm.beginTransaction()
                .add(R.id.fragment_container, fragment)
                .commit();
        }
    }
}
```

"Create a new fragment transaction, include one add operation in it, and then commit it."

3.3 Quản lý fragment với Fragment Manager

Fragment transactions



```
if (fragment == null) {
    fragment = new CrimeFragment();
    fm.beginTransaction()
        .add(R.id.fragment_container, fragment)
        .commit();
}
```

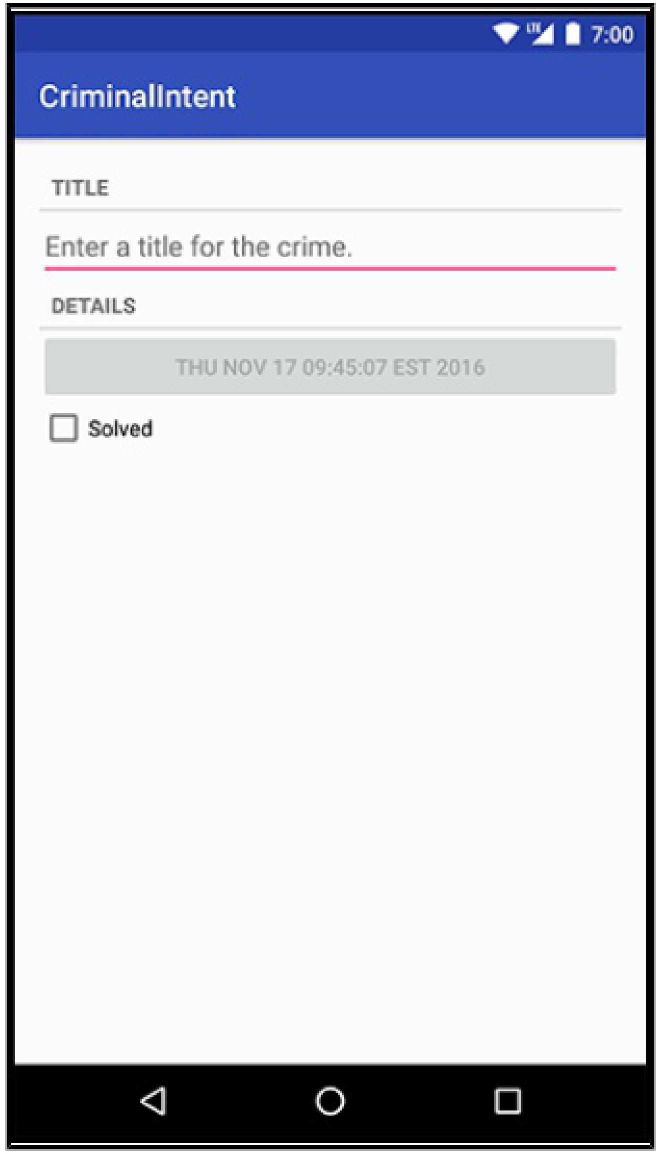
“Create a new fragment transaction, include one add operation in it, and then commit it.”

A container view ID serves two purposes:

- ✓ It tells the FragmentManager where in the activity’s view the fragment’s view should appear.
- ✓ It is used as a unique identifier for a fragment in the FragmentManager’s list.

3.3 Quản lý fragment với Fragment Manager

CrimeFragment’s view hosted by CrimeActivity



7:00

CriminalIntent

TITLE

Enter a title for the crime.

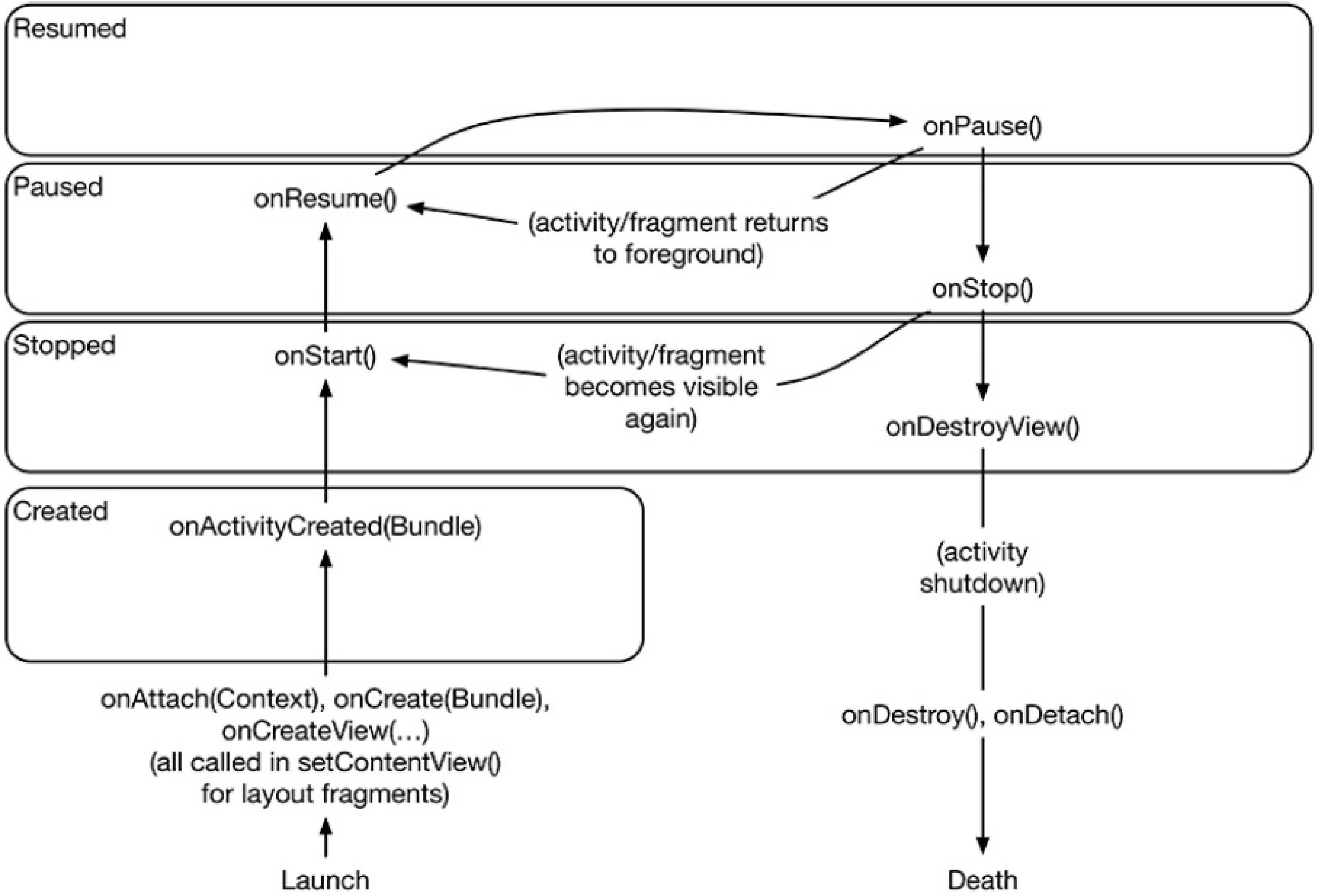
DETAILS

THU NOV 17 09:45:07 EST 2016

□ Solved

3.3 Quản lý fragment với Fragment Manager

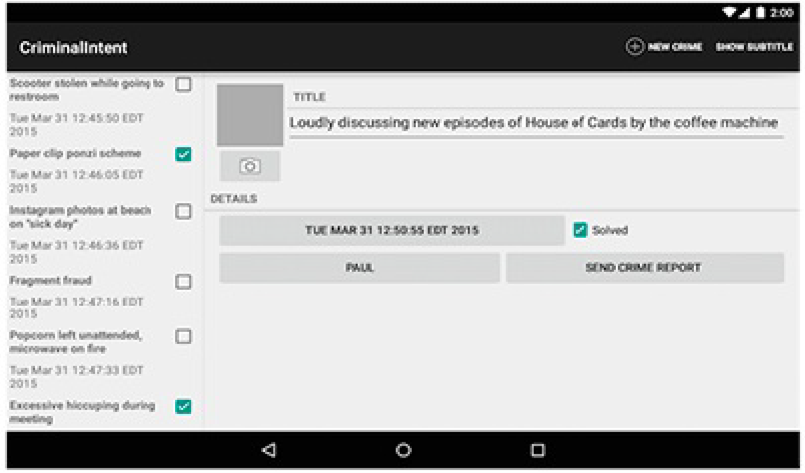
The FragmentManager and the fragment lifecycle


Resumed
onPause()
Paused
onResume()
(activity/fragment returns to foreground)
onStop()
Stopped
onStart()
(activity/fragment becomes visible again)
onDestroyView()
Created
onActivityCreated(Bundle)
(activity shutdown)
onAttach(Context), onCreate(Bundle), onCreateView(...)
(all called in setContentView() for layout fragments)
onDestroy(), onDetach()
Launch
Death

3.3 Quản lý fragment với Fragment Manager

Application Architecture with Fragments

Figure 7.21 Less is more



2:00

CriminalIntent

NEW CRIME SHOW SUBTITLE

Scooter stolen while going to restroom □
Tue Mar 31 12:45:50 EDT 2015

Paper clip ponzi scheme ☑
Tue Mar 31 12:46:05 EDT 2015

Instagram photos at beach on "sick day" □
Tue Mar 31 12:46:36 EDT 2015

Fragment fraud □
Tue Mar 31 12:47:16 EDT 2015

Popcorn left unattended, microwave on fire □
Tue Mar 31 12:47:33 EDT 2015

Excessive hiccuping during meeting ☑

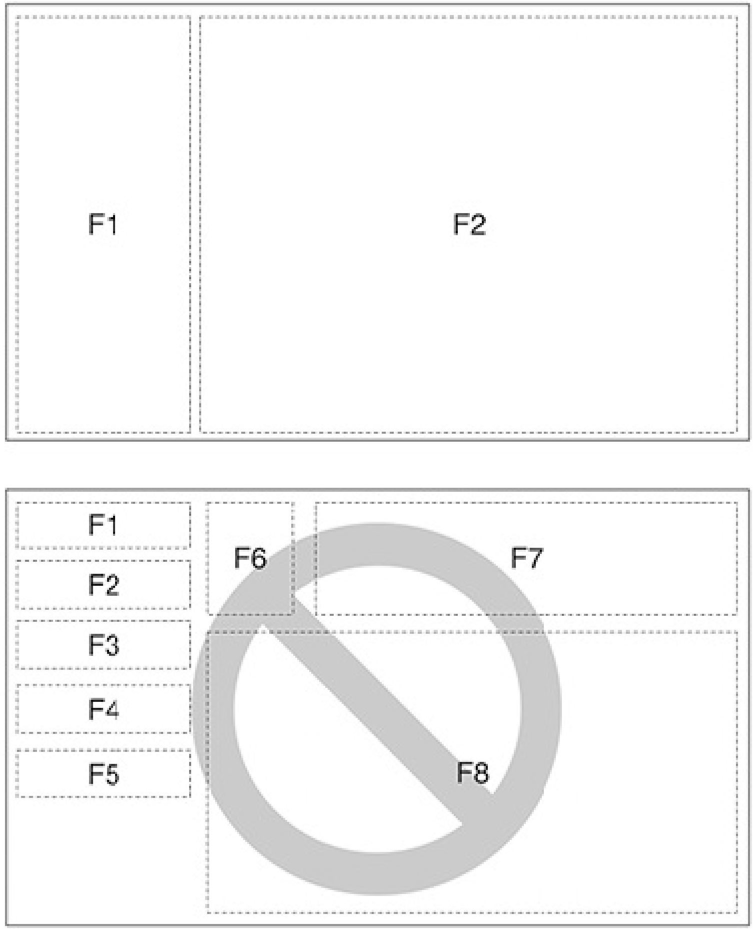
TITLE

Loudly discussing new episodes of House of Cards by the coffee machine

DETAILS

TUE MAR 31 12:50:55 EDT 2015 ☑ Solved

PAUL SEND CRIME REPORT

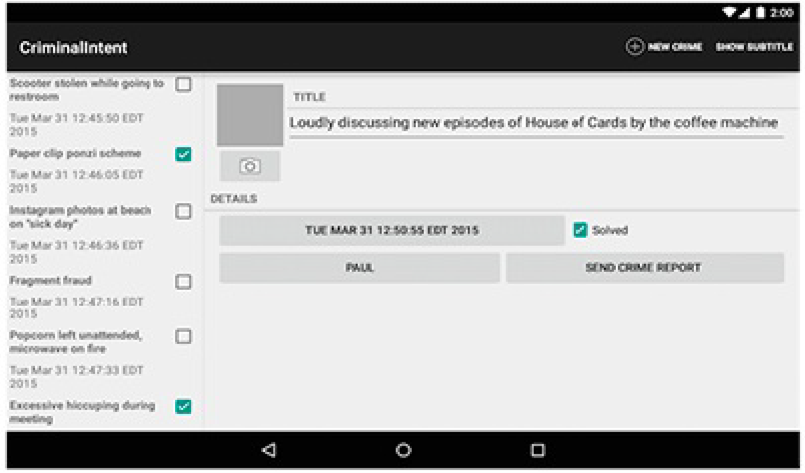


- Use fragments responsibly.
- A good rule of thumb is to have no more than two or three fragments on the screen at a time.

3.3 Quản lý fragment với Fragment Manager

Application Architecture with Fragments

Figure 7.21 Less is more



Scooter stolen while going to restroom ☐
Tue Mar 31 12:45:50 EDT 2015

Paper clip ponzi scheme ☑
Tue Mar 31 12:46:05 EDT 2015

Instagram photos at beach on "sick day" ☐
Tue Mar 31 12:46:36 EDT 2015

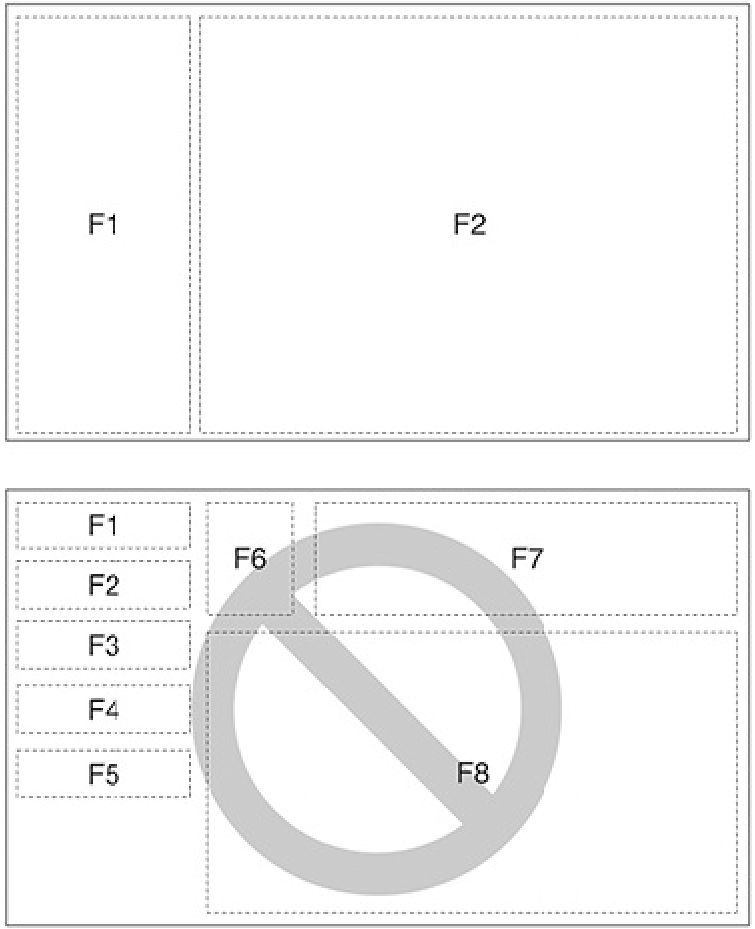
Fragment fraud ☐
Tue Mar 31 12:47:16 EDT 2015

Popcorn left unattended, microwave on fire ☐
Tue Mar 31 12:47:33 EDT 2015

Excessive hiccuping during meeting ☑

TITLE
Loudly discussing new episodes of House of Cards by the coffee machine

DETAILS
TUE MAR 31 12:50:55 EDT 2015
☑ Solved
PAUL
SEND CRIME REPORT

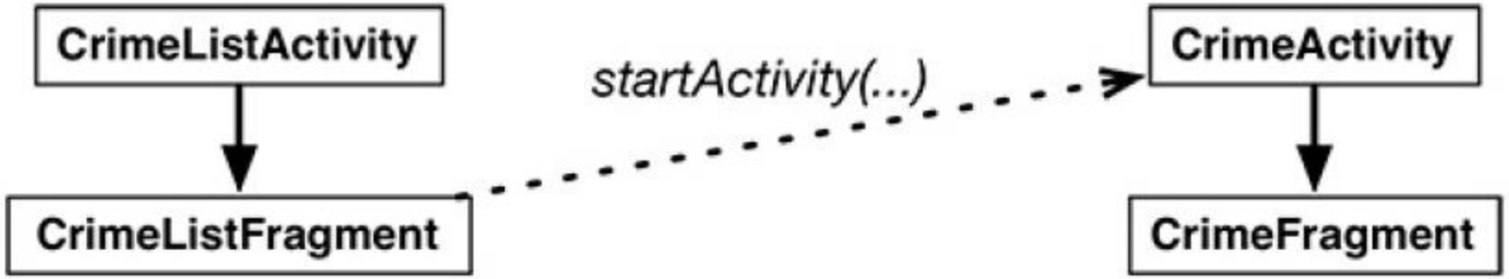


- Use fragments responsibly.
- A good rule of thumb is to have no more than two or three fragments on the screen at a time.

3.4 Thiết kế fragment có tham số

Using Fragment Arguments

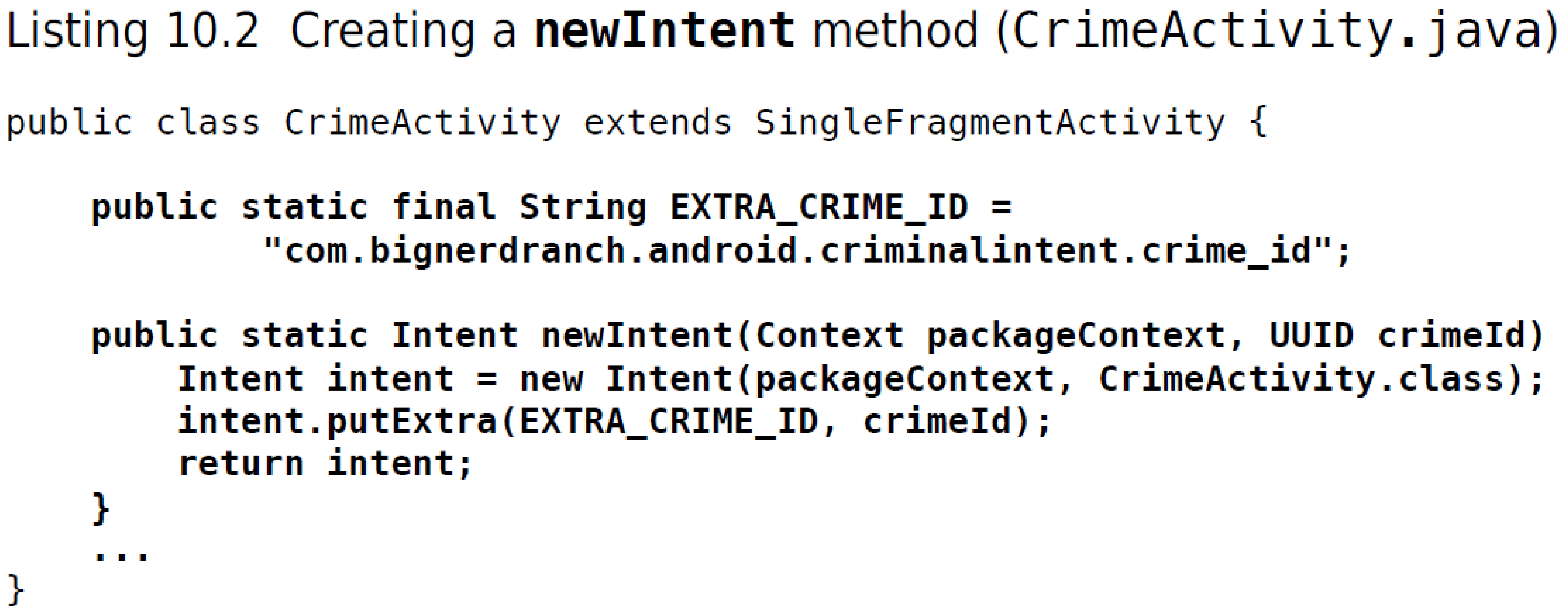
Figure 10.1 Starting **CrimeActivity** from **CrimeListActivity**



```
Intent intent = new Intent(getActivity(), CrimeActivity.class);
startActivity(intent);
```

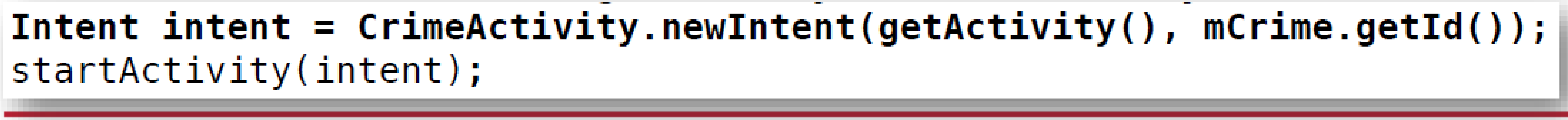
3.4 Thiết kế fragment có tham số

Putting an extra



Listing 10.2 Creating a **newIntent** method (CrimeActivity.java)

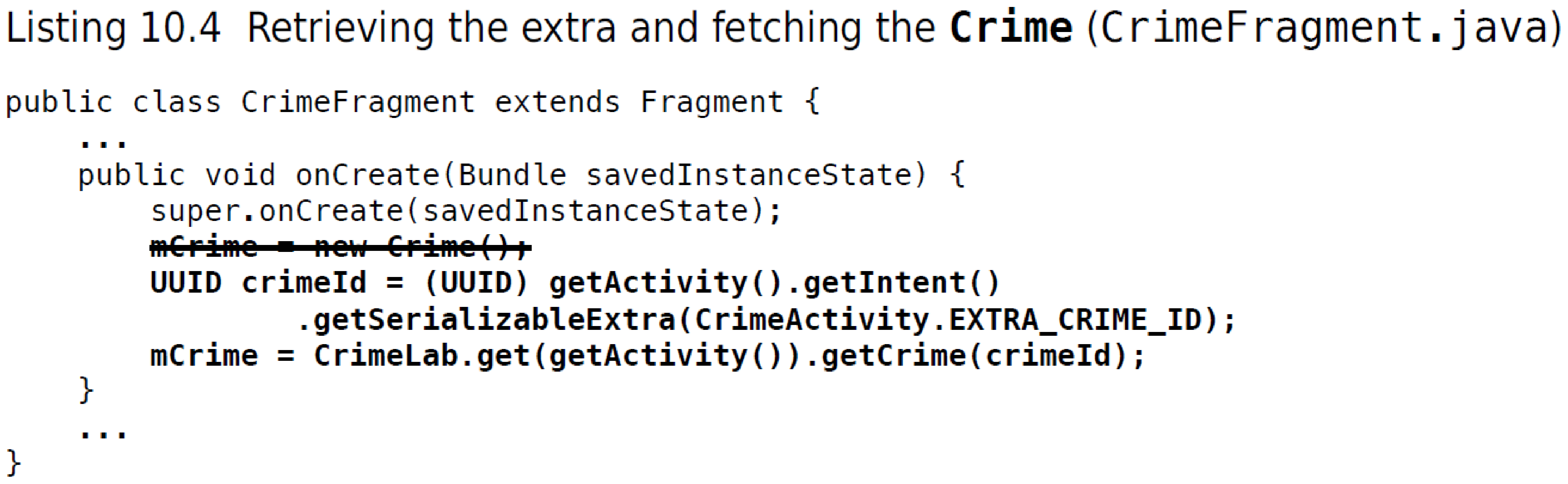
```
public class CrimeActivity extends SingleFragmentActivity {

    public static final String EXTRA_CRIME_ID =
            "com.bignerdranch.android.criminalintent.crime_id";

    public static Intent newIntent(Context packageContext, UUID crimeId) {
        Intent intent = new Intent(packageContext, CrimeActivity.class);
        intent.putExtra(EXTRA_CRIME_ID, crimeId);
        return intent;
    }
    ...
}
```



```
Intent intent = CrimeActivity.newIntent(getActivity(), mCrime.getId());
startActivity(intent);
```

3.4 Thiết kế fragment có tham số

Retrieving an extra

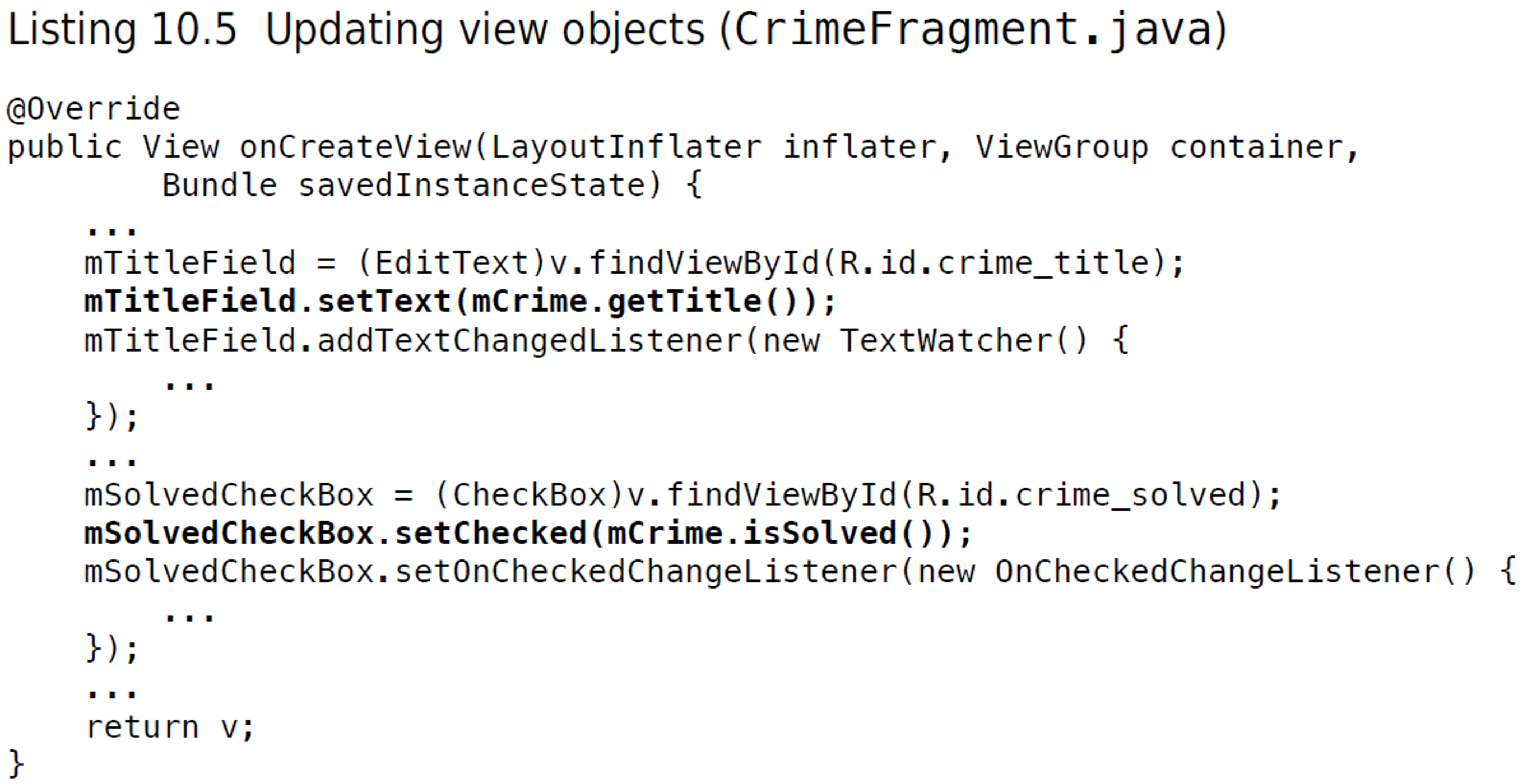


Listing 10.4 Retrieving the extra and fetching the **Crime** (CrimeFragment.java)

```
public class CrimeFragment extends Fragment {
    ...
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        ~~mCrime = new Crime();~~
        UUID crimeId = (UUID) getActivity().getIntent()
                .getSerializableExtra(CrimeActivity.EXTRA_CRIME_ID);
        mCrime = CrimeLab.get(getActivity()).getCrime(crimeId);
    }
    ...
}
```

3.4 Thiết kế fragment có tham số

Updating CrimeFragment’s view with Crime data



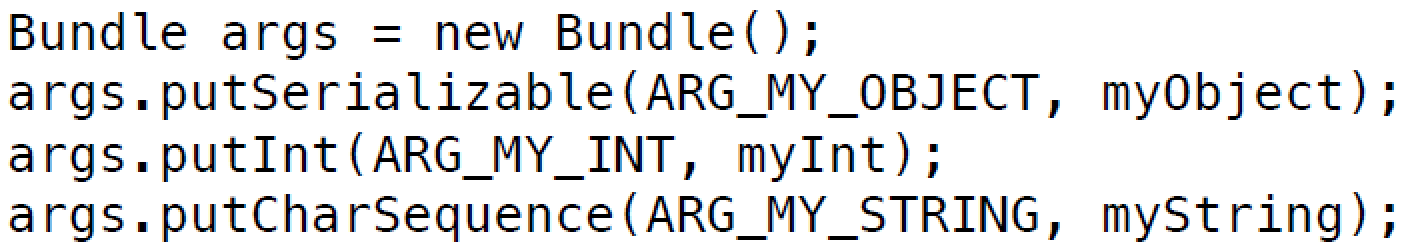
Listing 10.5 Updating view objects (CrimeFragment.java)

```
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {
    ...
    mTitleField = (EditText)v.findViewById(R.id.crime_title);
    mTitleField.setText(mCrime.getTitle());
    mTitleField.addTextChangedListener(new TextWatcher() {
        ...
    });
    ...
    mSolvedCheckBox = (CheckBox)v.findViewById(R.id.crime_solved);
    mSolvedCheckBox.setChecked(mCrime.isSolved());
    mSolvedCheckBox.setOnCheckedChangeListener(new OnCheckedChangeListener() {
        ...
    });
    ...
    return v;
}
```

3.4 Thiết kế fragment có tham số

Fragment Arguments

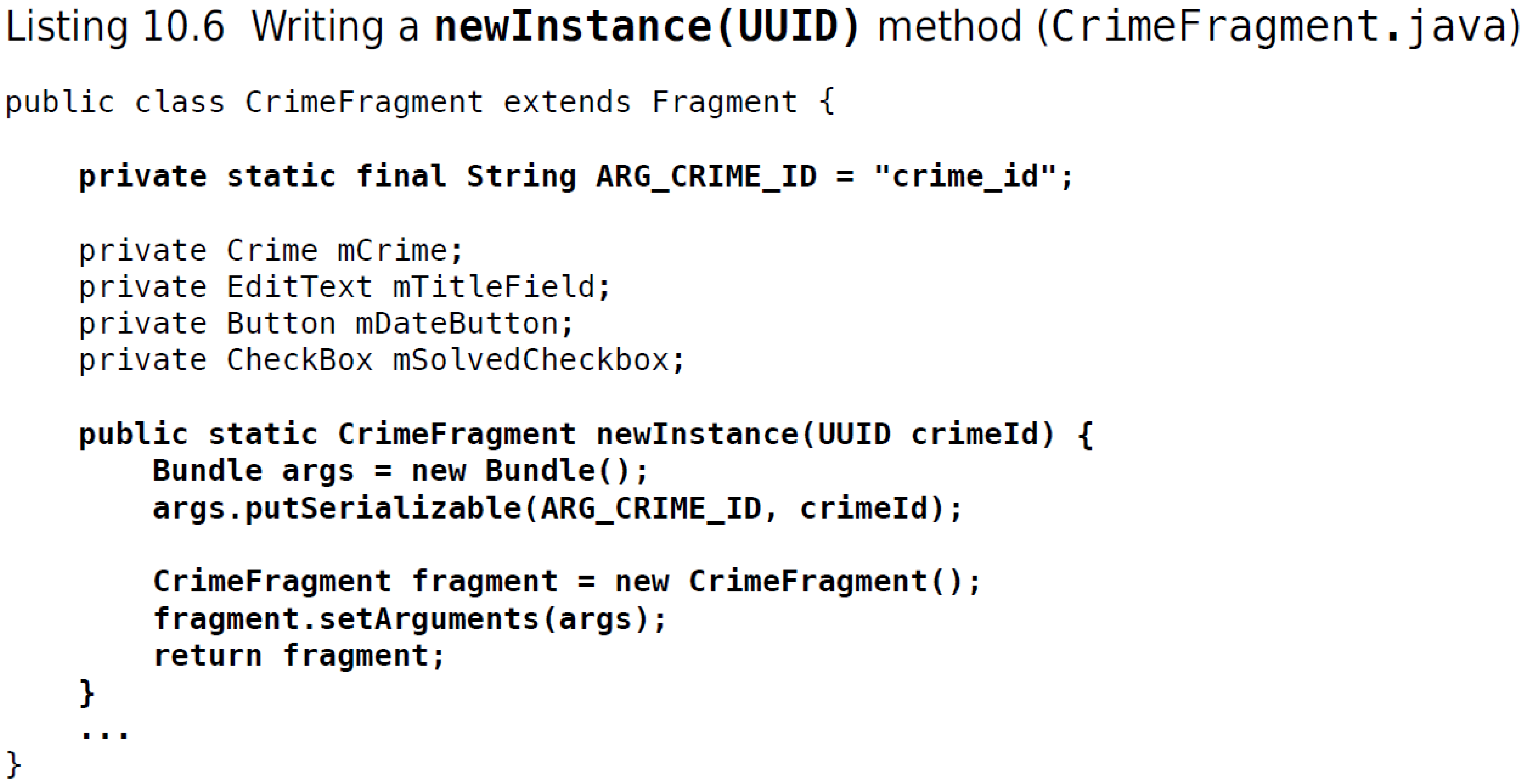
- To create fragment arguments, you first create a **Bundle** object
- Next, you use type-specific “put” methods of Bundle (similar to those of Intent) to add arguments to the bundle



```
Bundle args = new Bundle();
args.putSerializable(ARG_MY_OBJECT, myObject);
args.putInt(ARG_MY_INT, myInt);
args.putCharSequence(ARG_MY_STRING, myString);
```

3.4 Thiết kế fragment có tham số

Attaching arguments to a fragment

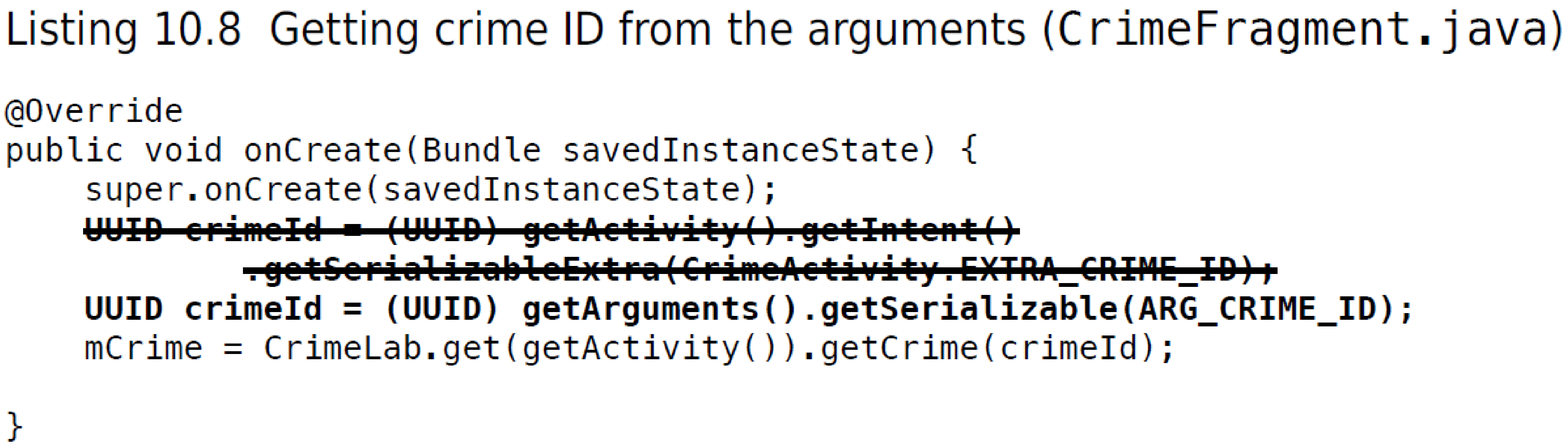


Listing 10.6 Writing a **newInstance(UUID)** method (CrimeFragment.java)

```
public class CrimeFragment extends Fragment {

    private static final String ARG_CRIME_ID = "crime_id";

    private Crime mCrime;
    private EditText mTitleField;
    private Button mDateButton;
    private CheckBox mSolvedCheckbox;

    public static CrimeFragment newInstance(UUID crimeId) {
        Bundle args = new Bundle();
        args.putSerializable(ARG_CRIME_ID, crimeId);

        CrimeFragment fragment = new CrimeFragment();
        fragment.setArguments(args);
        return fragment;
    }
    ...
}
```

3.4 Thiết kế fragment có tham số

Retrieving arguments

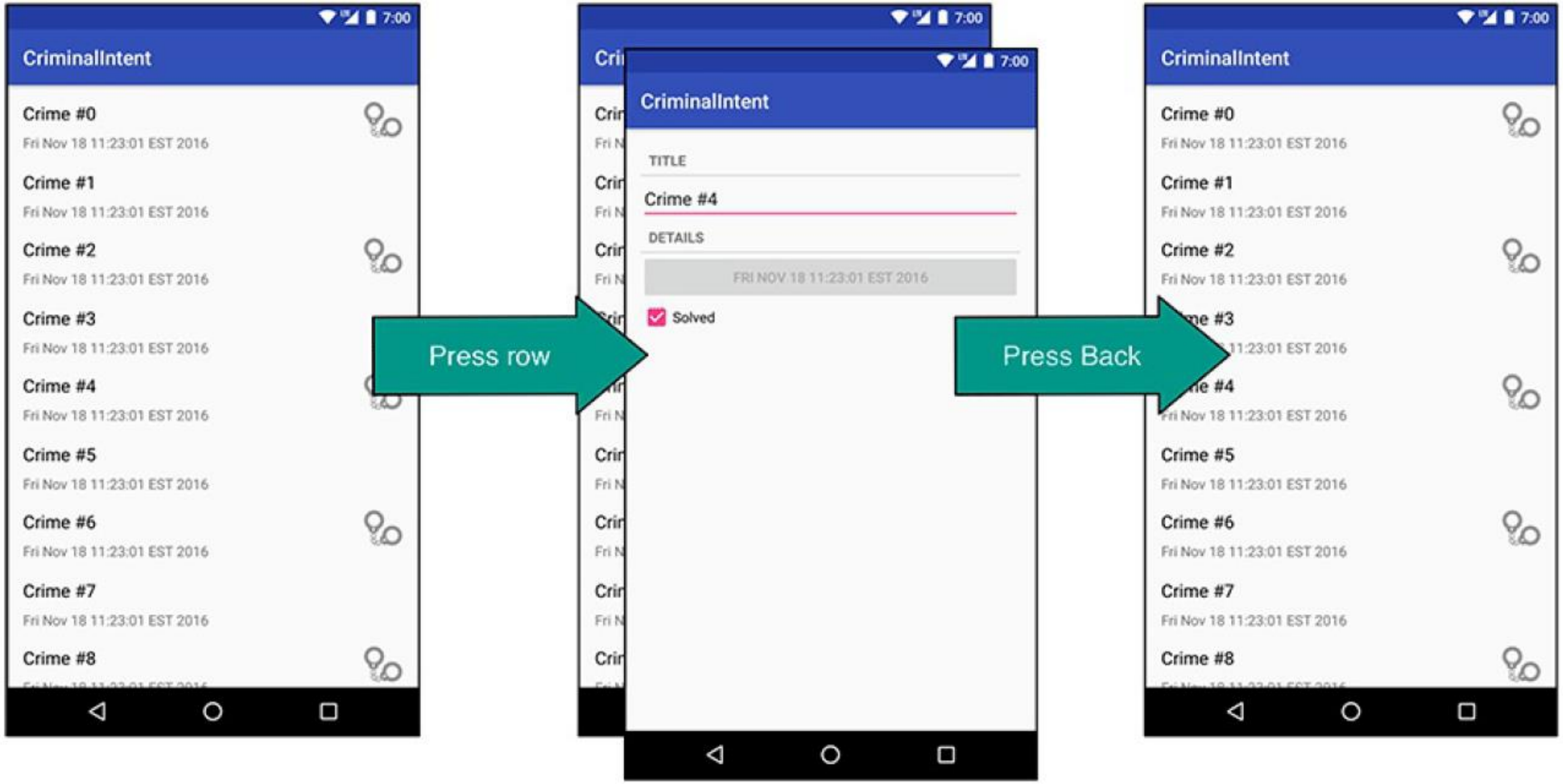


Listing 10.8 Getting crime ID from the arguments (CrimeFragment.java)

```
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    ~~UUID crimeId = (UUID) getActivity().getIntent()~~
            ~~.getSerializableExtra(CrimeActivity.EXTRA_CRIME_ID);~~
    UUID crimeId = (UUID) getArguments().getSerializable(ARG_CRIME_ID);
    mCrime = CrimeLab.get(getActivity()).getCrime(crimeId);

}
```

3.4 Thiết kế fragment có tham số

Run CriminalIntent



7:00

CriminalIntent

Crime #0
Fri Nov 18 11:23:01 EST 2016

Crime #1
Fri Nov 18 11:23:01 EST 2016

Crime #2
Fri Nov 18 11:23:01 EST 2016

Crime #3
Fri Nov 18 11:23:01 EST 2016

Crime #4
Fri Nov 18 11:23:01 EST 2016

Crime #5
Fri Nov 18 11:23:01 EST 2016

Crime #6
Fri Nov 18 11:23:01 EST 2016

Crime #7
Fri Nov 18 11:23:01 EST 2016

Crime #8

7:00

CriminalIntent

TITLE

Crime #4

DETAILS

FRI NOV 18 11:23:01 EST 2016

☑ Solved

7:00

CriminalIntent

Crime #0
Fri Nov 18 11:23:01 EST 2016

Crime #1
Fri Nov 18 11:23:01 EST 2016

Crime #2
Fri Nov 18 11:23:01 EST 2016

me #3
11:23:01 EST 2016

me #4
Fri Nov 18 11:23:01 EST 2016

Crime #5
Fri Nov 18 11:23:01 EST 2016

Crime #6
Fri Nov 18 11:23:01 EST 2016

Crime #7
Fri Nov 18 11:23:01 EST 2016

Crime #8

HUST

hust.edu.vn fb.com/dhbkhn

THANK YOU !

Lập trình ứng dụng di động

Mobile Application Programming

ET4710

PGS. TS. Đỗ Trọng Tuấn

*Viện Điện tử Viễn thông * Đại học Bách Khoa Hà Nội*

ONE LOVE. ONE FUTURE.