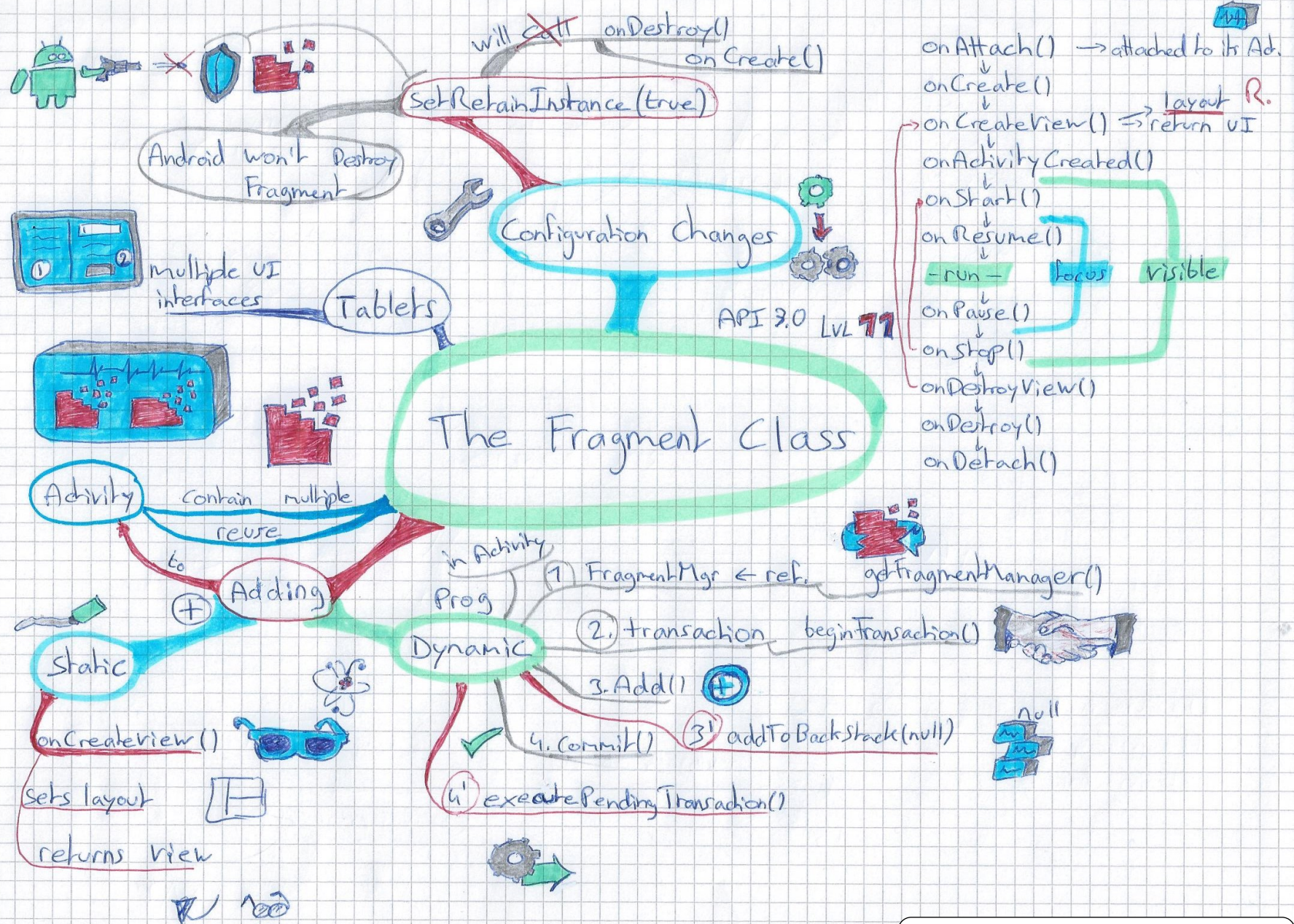


RL Chiapp

Android 3

05.02.14



Romain Chiappinelli

# PROGRAMMING HANDHELD SYSTEMS

ADAM PORTER



# THE FRAGMENT CLASS

# TABLET UIs

TABLETS HAVE LARGER DISPLAYS THAN  
PHONES DO

THEY CAN SUPPORT MULTIPLE UI PANES /  
USER BEHAVIORS AT THE SAME TIME

THE 1 ACTIVITY – 1 THING THE USER CAN DO  
HEURISTIC MAY NOT WORK FOR LARGER DEVICES

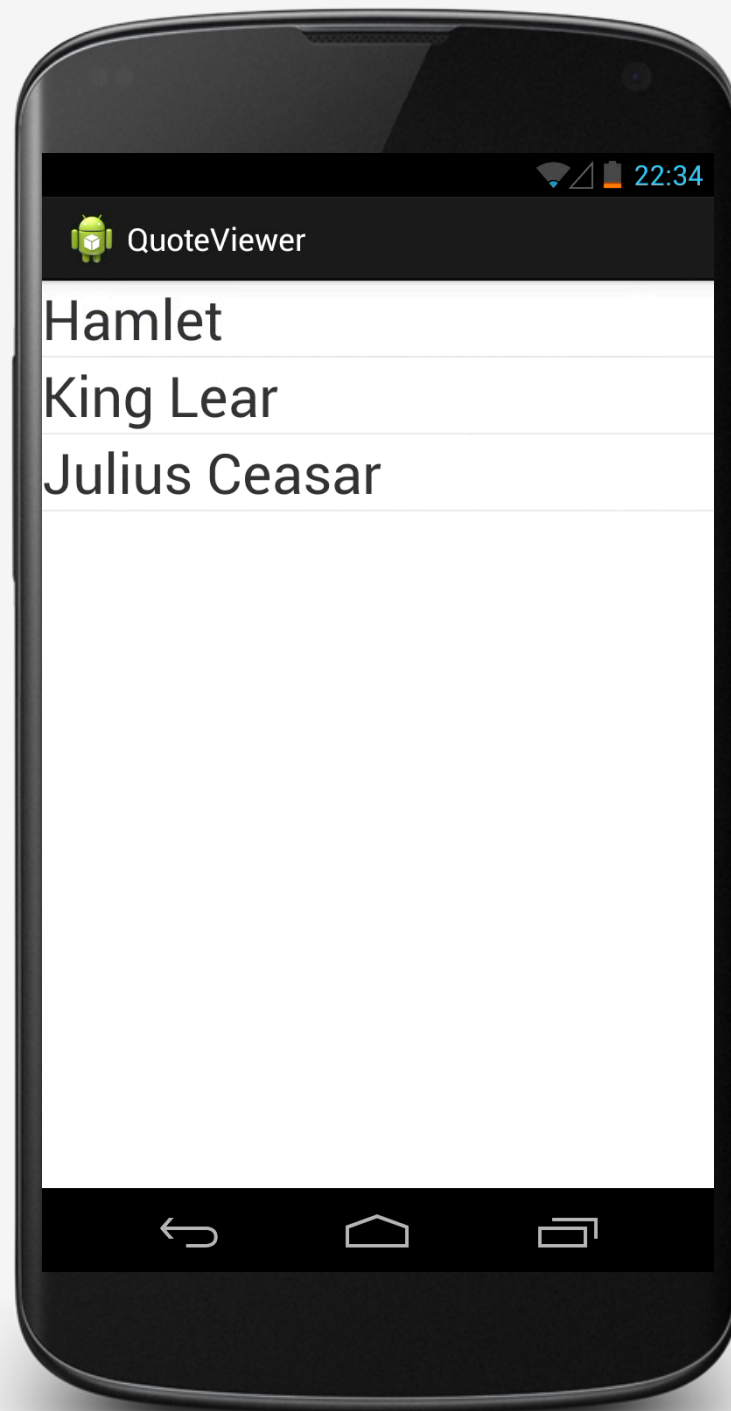
# QUOTEVIEWER

APPLICATION USES TWO ACTIVITIES

ONES SHOWS TITLES OF SHAKESPEARE PLAYS &  
ALLOWS USER TO SELECT ONE TITLE

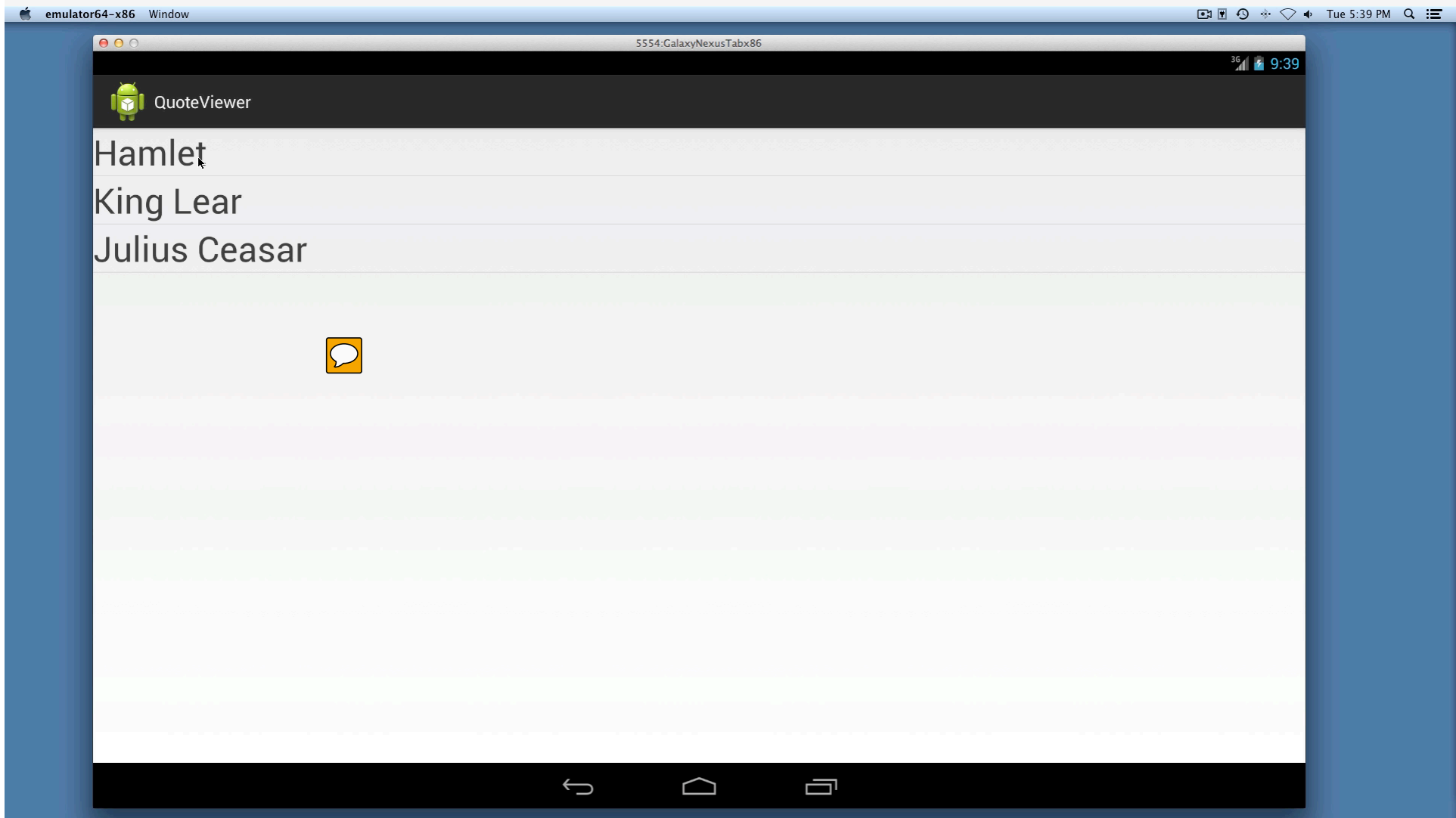
THE OTHER SHOWS A QUOTE FROM SELECTED PLAY





# THE QUOTEVIEWER UI

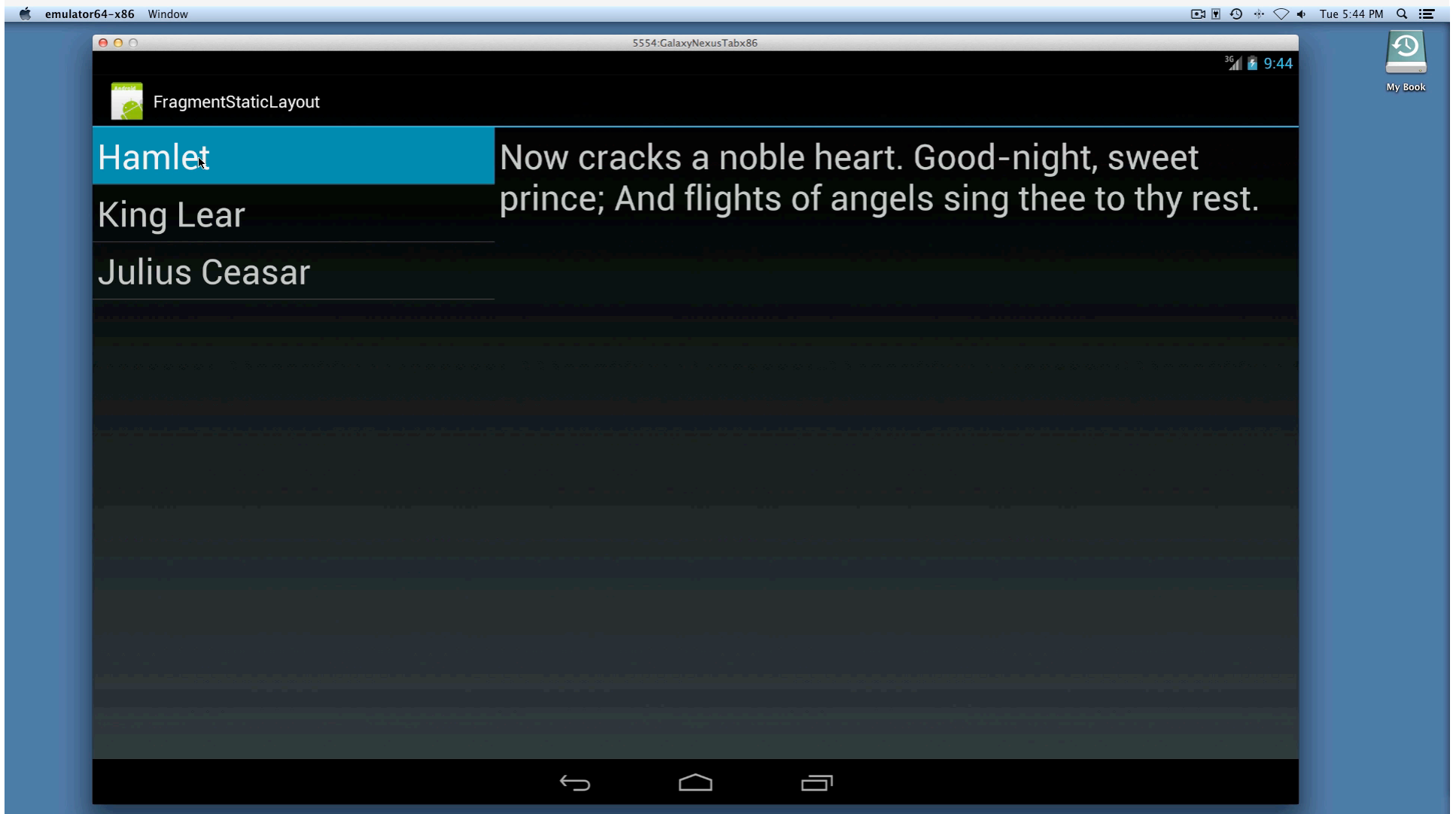
THIS LAYOUT IS REASONABLE ON A PHONE  
BUT INEFFICIENT ON A LARGER DEVICE





# BETTER LAYOUT

USE TWO COOPERATING LAYOUT UNITS ON ONE  
SCREEN



# FRAGMENT

REPRESENTS A BEHAVIOR / PORTION OF  
UI WITHIN AN ACTIVITY

MULTIPLE FRAGMENTS CAN BE  
EMBEDDED IN AN ACTIVITY TO CREATE  
A MULTI-PANE UI

A SINGLE FRAGMENT CAN BE REUSED  
ACROSS MULTIPLE ACTIVITIES

# FRAGMENT LIFECYCLE

FRAGMENT LIFECYCLE IS COORDINATED  
WITH THE LIFECYCLE OF ITS CONTAINING  
ACTIVITY

FRAGMENTS HAVE THEIR OWN  
LIFECYCLES AND RECEIVE THEIR OWN  
CALLBACKS

# FRAGMENT LIFECYCLE STATES

## RESUMED

FRAGMENT IS VISIBLE IN THE RUNNING  
ACTIVITY

## PAUSED

ANOTHER ACTIVITY IS IN THE FOREGROUND  
AND HAS FOCUS, CONTAINING ACTIVITY IS  
VISIBLE

## STOPPED

THE FRAGMENT IS NOT VISIBLE

# LIFECYCLE CALLBACK METHODS

# ACTIVITY CREATED: ONATTACH()

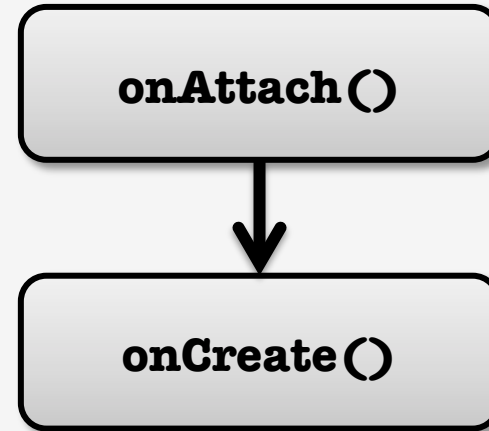
FRAGMENT IS FIRST  
ATTACHED TO ITS  
ACTIVITY

**onAttach()**



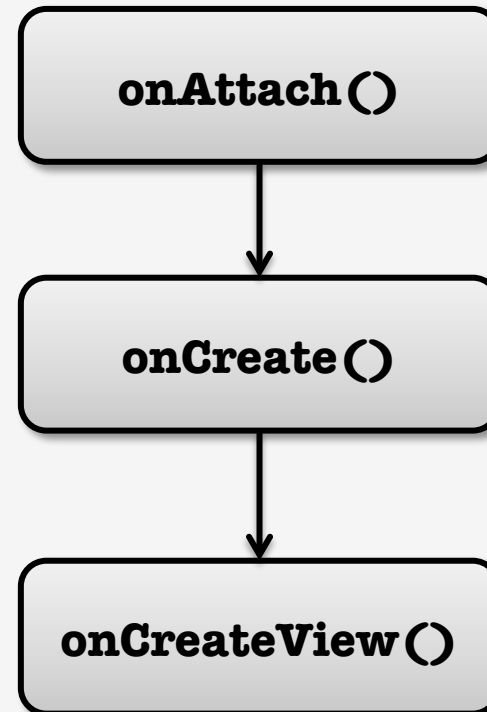
# ONCREATE()

INITIALIZE THE  
FRAGMENT



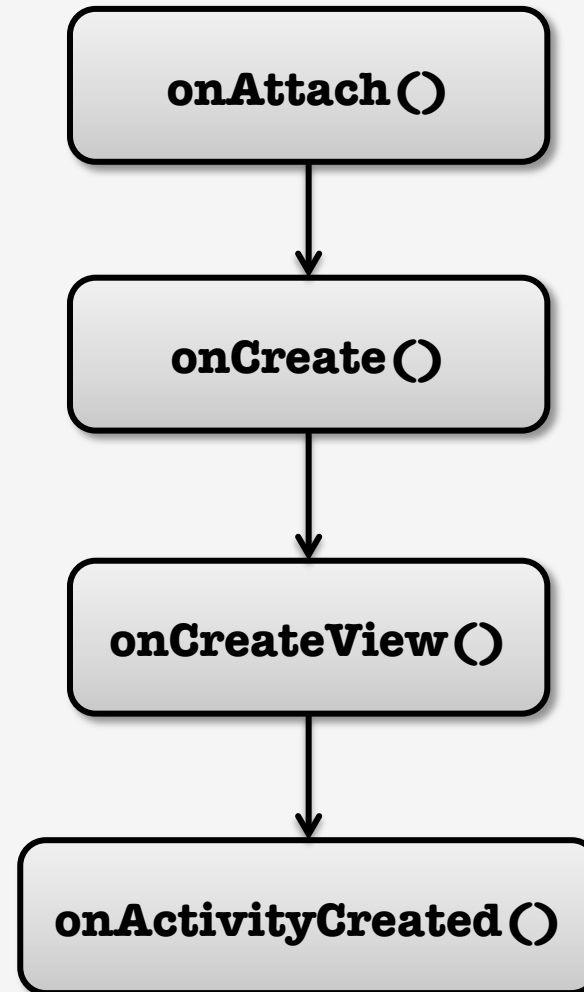
# ONCREATEVIEW()

FRAGMENT SETS UP &  
RETURNS ITS USER  
INTERFACE



# ONACTIVITYCREATED()

CONTAINING ACTIVITY  
HAS COMPLETED  
onCreate() AND THE  
FRAGMENT HAS BEEN  
INSTALLED



# ACTIVITY STARTED: onStart()

HOSTING ACTIVITY  
ABOUT TO BECOME  
VISIBLE

**onStart ()**

# ACTIVITY RESUMED: onResume()

HOSTING ACTIVITY IS  
ABOUT TO BECOME  
VISIBLE AND READY  
FOR USER INTERACTION

**onResume()**

# ACTIVITY PAUSED: onPause()

HOSTING ACTIVITY IS  
VISIBLE, BUT DOES NOT  
HAVE FOCUS

A rectangular button with rounded corners, a black border, and a light gray gradient background. The text "onPause()" is centered on the button in a bold, black, sans-serif font.

**onPause()**

# ACTIVITY STOPPED: ONSTOP()

HOSTING ACTIVITY IS  
NO LONGER VISIBLE



**onStop ()**



# ACTIVITY DESTROYED: ONDESTROYVIEW()

VIEW PREVIOUSLY  
CREATED BY  
ONCREATEVIEW() HAS  
BEEN DETACHED FROM  
THE ACTIVITY

## TYPICAL ACTIONS

CLEAN UP VIEW  
RESOURCES

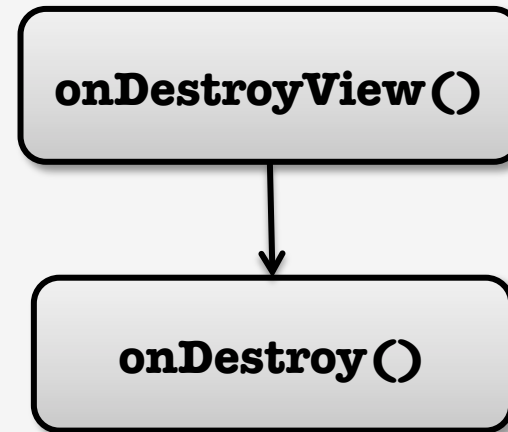
**onDestroyView()**

# ONDESTROY()

FRAGMENT IS NO  
LONGER IN USE

TYPICAL ACTIONS

CLEAN UP FRAGMENT  
RESOURCES

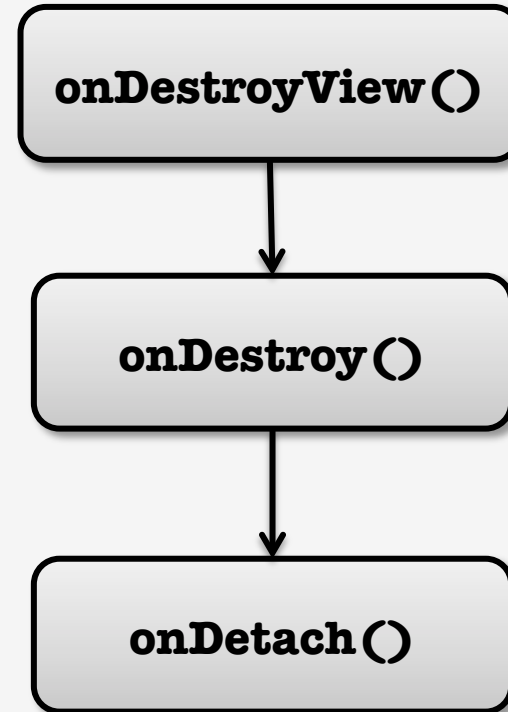


# ONDETACH()

FRAGMENT NO LONGER  
ATTACHED TO ITS  
ACTIVITY

## TYPICAL ACTIONS

NULL OUT REFERENCES  
TO HOSTING ACTIVITY



# ADDING FRAGMENTS TO ACTIVITIES

TWO GENERAL WAYS TO ADD  
FRAGMENTS TO AN ACTIVITY'S LAYOUT

DECLARE IT STATICALLY IN THE  
ACTIVITY'S LAYOUT FILE

ADD IT PROGRAMMATICALLY USING THE  
FRAGMENTMANAGER

# FRAGMENT LAYOUT

LAYOUT CAN BE INFLATED/  
IMPLEMENTED IN ONCREATEVIEW()

ONCREATEVIEW() MUST RETURN THE  
VIEW AT THE ROOT OF THE FRAGMENT'S  
LAYOUT

THIS VIEW IS ADDED TO THE CONTAINING  
ACTIVITY



Hamlet

King Lear

Julius Ceasar

Now cracks a noble heart. Good-night, sweet prince; And  
flights of angels sing thee to thy rest.



# FRAGMENTSTATICLAYOUT

```
public class QuoteViewerActivity extends Activity implements
    ListSelectionListener {

    public static String[] mTitleArray;
    public static String[] mQuoteArray;
    private QuotesFragment mDetailsFragment;

    private static final String TAG = "QuoteViewerActivity";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        // Get the string arrays with the titles and quotes
        mTitleArray = getResources().getStringArray(R.array.Titles);
        mQuoteArray = getResources().getStringArray(R.array.Quotes);

        setContentView(R.layout.main);

        // Get a reference to the QuotesFragment
        mDetailsFragment = (QuotesFragment) getFragmentManager()
            .findFragmentById(R.id.details);
    }
}
```



# FRAGMENTSTATICLAYOUT

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:baselineAligned="false"
    android:orientation="horizontal" >

    <fragment
        android:id="@+id/titles"
        android:layout_width="0px"
        android:layout_height="match_parent"
        android:layout_weight="1"
        class="course.examples.Fragments.StaticLayout.TitlesFragment" />

    <fragment
        android:id="@+id/details"
        android:layout_width="0px"
        android:layout_height="match_parent"
        android:layout_weight="2"
        class="course.examples.Fragments.StaticLayout.QuotesFragment" />

</LinearLayout>
```

# FRAGMENTSTATICLAYOUT

```
@Override
public void onCreate(Bundle savedInstanceState) {
    Log.i(TAG, getClass().getSimpleName() + ":entered onCreate()");
    super.onCreate(savedInstanceState);
}
```

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

    <TextView
        android:id="@+id/quoteView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:padding="5dip"
        android:textSize="32sp" >
    </TextView>

</LinearLayout>
```

# ADDING FRAGMENTS DYNAMICALLY

WHILE AN ACTIVITY'S RUNNING YOU CAN  
ADD A FRAGMENT TO ITS LAYOUT

GET REFERENCE TO THE  
FRAGMENTMANAGER



BEGIN A FRAGMENTTRANSACTION

ADD THE FRAGMENT

COMMIT THE FRAGMENTTRANSACTION

# FRAGMENTPROGRAMMATICLAYOUT

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    // Get the string arrays with the titles and quotes
    mTitleArray = getResources().getStringArray(R.array.Titles);
    mQuoteArray = getResources().getStringArray(R.array.Quotes);

    setContentView(R.layout.main);

    // Get a reference to the FragmentManager
    FragmentManager fragmentManager = getFragmentManager();

    // Begin a new FragmentTransaction
    FragmentTransaction fragmentTransaction = fragmentManager
        .beginTransaction();

    // Add the TitleFragment
    fragmentTransaction.add(R.id.title_frame, new TitlesFragment());

    // Add the QuoteFragment
    fragmentTransaction.add(R.id.quote_frame, mQuoteFragment);

    // Commit the FragmentTransaction
    fragmentTransaction.commit();
}
```

# FRAGMENTPROGRAMMATICLAYOUT

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/activityFrame"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal"
    android:baselineAligned="false">

    <FrameLayout
        android:id="@+id/title_frame"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1" >
    </FrameLayout>

    <FrameLayout
        android:id="@+id/quote_frame"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="2" >
    </FrameLayout>

</LinearLayout>
```



FragmentProgrammaticLayout



My Book

Hamlet

King Lear

Julius Ceasar

As flies to wanton boys, are we to the gods; they  
kill us for their sport.





# DYNAMIC LAYOUT

FRAGMENT TRANSACTIONS ALLOW YOU  
TO DYNAMICALLY CHANGE YOUR APP'S  
USER INTERFACE

CAN MAKE THE INTERFACE MORE FLUID  
& TAKE BETTER ADVANTAGE OF  
AVAILABLE SCREEN SPACE





FragmentManager

The Tragedy of Hamlet,  
Prince of Denmark

King Lear

Julius Ceasar

Now cracks a noble heart. Good-night, sweet  
prince; And flights of angels sing thee to thy rest.



# FRAGMENTDYNAMICLAYOUT

```
@Override
protected void onCreate(Bundle savedInstanceState) {

    Log.i(TAG, getClass().getSimpleName() + ":entered onCreate()");

    super.onCreate(savedInstanceState);

    // Get the string arrays with the titles and quotes
    mTitleArray = getResources().getStringArray(R.array.Titles);
    mQuoteArray = getResources().getStringArray(R.array.Quotes);

    setContentView(R.layout.main);

    // Get references to the TitleFragment and to the QuotesFragment
    mTitleFrameLayout = (FrameLayout) findViewById(R.id.title_fragment_container);
    mQuotesFrameLayout = (FrameLayout) findViewById(R.id.quote_fragment_container);

    // Get a reference to the FragmentManager
    mFragmentManager = getFragmentManager();

    // Start a new FragmentTransaction
    FragmentTransaction fragmentTransaction = mFragmentManager
        .beginTransaction();

    // Add the TitleFragment to the layout
    fragmentTransaction.add(R.id.title_fragment_container,
        new TitlesFragment());

    // Commit the FragmentTransaction
    fragmentTransaction.commit();

    // Add a OnBackStackChangeListener to reset the layout when the back stack changes
    mFragmentManager
        .addOnBackStackChangeListener(new FragmentManager.OnBackStackChangeListener() {
            public void onBackStackChanged() {
                setLayout();
            }
        });
}
```

# FRAGMENTDYNAMICLAYOUT

```
// Called when the user selects an item in the TitlesFragment
@Override
public void onListSelection(int index) {

    // If the QuoteFragment has not been added, add it now
    if (!mQuoteFragment.isAdded()) {

        // Start a new FragmentTransaction
        FragmentTransaction fragmentTransaction = mFragmentManager
            .beginTransaction();

        // Add the QuoteFragment to the layout
        fragmentTransaction.add(R.id.quote_fragment_container,
            mQuoteFragment);

        // Add this FragmentTransaction to the backstack
        fragmentTransaction.addToBackStack(null);

        // Commit the FragmentTransaction
        fragmentTransaction.commit();

        // Force Android to execute the committed FragmentTransaction
        mFragmentManager.executePendingTransactions();
    }

    if (mQuoteFragment.getShownIndex() != index) {

        // Tell the QuoteFragment to show the quote string at position index
        mQuoteFragment.showQuoteAtIndex(index);
    }
}
```



# CONFIGURATION CHANGES

IF YOU CALL `setRetainInstance(true)`,   
ANDROID WON'T DESTROY THE FRAGMENT  
ON CONFIGURATION CHANGES

# CONFIGURATION CHANGES

RESULTS IN SOME CHANGES TO LIFECYCLE  
CALLBACK SEQUENCE

onDestroy() WILL NOT BE CALLED

onCreate() WILL NOT BE CALLED

# FRAGMENTSTATICCONFIGLAYOUT

FUNCTIONALITY IS SIMILAR TO PREVIOUS  
EXAMPLES

# FRAGMENTSTATICCONFIGLAYOUT

IN LANDSCAPE MODE

BOTH FRAGMENTS USE A LARGE FONT 

TITLEFRAGMENT TAKES MORE  
HORIZONTAL SPACE & ALLOWS LONG  
TITLES TO SPAN MULTIPLE LINES

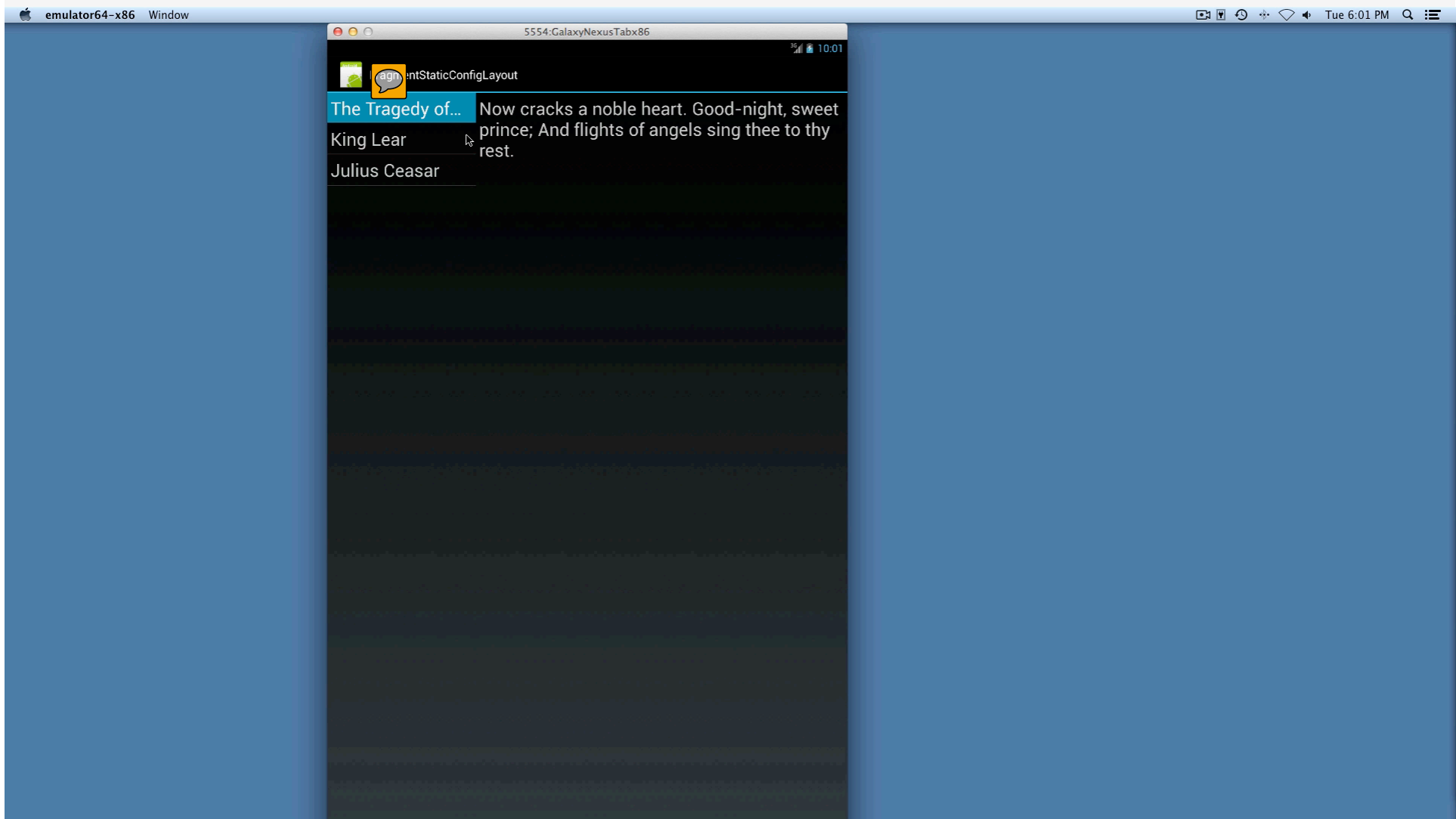
# FRAGMENTSTATICCONFIGLAYOUT

IN PORTRAIT MODE

BOTH FRAGMENTS USE A SMALLER FONT 

TITLEFRAGMENT WILL USE LESS SPACE  
AND WILL ELLIPSIZE LONG TITLES,  
LIMITING THEM TO A SINGLE LINE





# FRAGMENTDYNAMICLAYOUT

```
@Override
protected void onCreate(Bundle savedInstanceState) {

    Log.i(TAG, getClass().getSimpleName() + ":entered onCreate()");

    super.onCreate(savedInstanceState);

    // Get the string arrays with the titles and quotes
    mTitleArray = getResources().getStringArray(R.array.Titles);
    mQuoteArray = getResources().getStringArray(R.array.Quotes);

    setContentView(R.layout.main);

    // Get references to the TitleFragment and to the QuotesFragment
    mTitleFrameLayout = (FrameLayout) findViewById(R.id.title_fragment_container);
    mQuotesFrameLayout = (FrameLayout) findViewById(R.id.quote_fragment_container);

    // Get a reference to the FragmentManager
    mFragmentManager = getFragmentManager();

    // Start a new FragmentTransaction
    FragmentTransaction fragmentTransaction = mFragmentManager
        .beginTransaction();

    // Add the TitleFragment to the layout
    fragmentTransaction.add(R.id.title_fragment_container,
        new TitlesFragment());

    // Commit the FragmentTransaction
    fragmentTransaction.commit();

    // Add a OnBackStackChangeListener to reset the layout when the back stack changes
    mFragmentManager
        .addOnBackStackChangeListener(new FragmentManager.OnBackStackChangeListener() {
            public void onBackStackChanged() {
                setLayout();
            }
        });
}
```

# FRAGMENTDYNAMICLAYOUT

```
// Called when the user selects an item in the TitlesFragment
@Override
public void onListSelection(int index) {

    // If the QuoteFragment has not been added, add it now
    if (!mQuoteFragment.isAdded()) {

        // Start a new FragmentTransaction
        FragmentTransaction fragmentTransaction = mFragmentManager
            .beginTransaction();

        // Add the QuoteFragment to the layout
        fragmentTransaction.add(R.id.quote_fragment_container,
            mQuoteFragment);

        // Add this FragmentTransaction to the backstack
        fragmentTransaction.addToBackStack(null);

        // Commit the FragmentTransaction
        fragmentTransaction.commit();

        // Force Android to execute the committed FragmentTransaction
        mFragmentManager.executePendingTransactions();
    }

    if (mQuoteFragment.getShownIndex() != index) {

        // Tell the QuoteFragment to show the quote string at position index
        mQuoteFragment.showQuoteAtIndex(index);
    }
}
```

# NEXT TIME

## USER INTERFACE CLASSES