

Software Engineering Large Practical

Debugging and Testing Android apps

Stephen Gilmore

School of Informatics, University of Edinburgh

November 13th, 2013

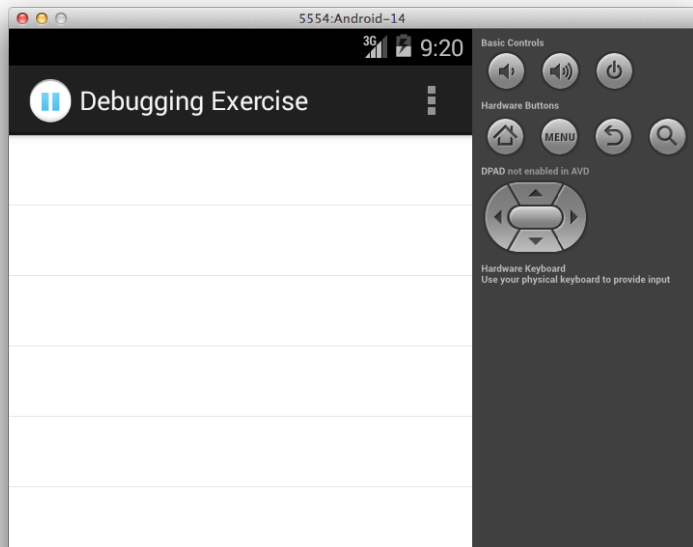
- ▶ This is the last SELP lecture.

Debugging Android Apps

In this lecture we consider debugging and testing our Android apps, using the Android debugger and the Android Robotium unit testing suite.

We follow an example due to Lars Vogel, available at <http://www.vogella.com/articles/AndroidDebugging/article.html>

The problem: our app should display a list of countries, but it doesn't



Getting started: Choose "Debug As"

The screenshot shows an IDE window with the following components:

- Package Explorer:** Shows the project structure for 'Countries', including 'src', 'res', and various resource folders.
- MainActivity.java:** The code editor displays the following code:

```
super(context, android.R.layout.simple_list_item_1, android.R.id.text1, items);
}
}
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    // get list of countries
    List<String> countries = CountryUtils.getCountries();
    // create and assign list adapter
    setListAdapter(new StringListAdapter(this, countries));
}
protected void onListItemClick(ListView l, View v, int position, long id) {
    String item = (String) getListAdapter().getItem(position);
    Toast.makeText(this, "Selected: " + item, Toast.LENGTH_SHORT);
}
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // create options menu
    getMenuInflater().inflate(R.menu.main, menu);
    return true;
}
```
- LogCat:** The bottom window shows a table of log messages:

Len	Time	PID	TID	Application	Tag	Text
I	11-12 09:59:29.393	1425	1425	com.vogella.andro...	System.out	waiting for debugger t
I	11-12 09:59:29.593	1425	1425	com.vogella.andro...	System.out	waiting for debugger t
I	11-12 09:59:29.793	1425	1425	com.vogella.andro...	System.out	waiting for debugger t
I	11-12 09:59:30.193	1425	1425	com.vogella.andro...	System.out	waiting for debugger t
I	11-12 09:59:30.403	1425	1425	com.vogella.andro...	System.out	waiting for debugger t
I	11-12 09:59:30.603	1425	1425	com.vogella.andro...	System.out	debugger has settled (
D	11-12 09:59:47.143	1425	1425	com.vogella.andro...	dalvikvm	GC_FOR_ALLOC freed 142
I	11-12 09:59:47.143	1425	1425	com.vogella.andro...	dalvikvm-heap	Grow heap (frag case)

Debug configurations

Debug Configurations

Create, manage, and run configurations

Android Application

Name: Countries

Project: Countries

Launch Action:

- Launch Default Activity
- Launch:
- Do Nothing

Filter matched 13 of 13 items

Apply Revert

Switch to the “Debug” perspective to see new views such as “Variables” and “Breakpoints”

The screenshot shows the Android Studio IDE in the Debug perspective. The top toolbar includes icons for Run, Stop, Step Over, Step Into, Step Out, and Breakpoints, along with a "Quick Access" search bar. The "Debug" tab is active in the top right. The main interface is divided into several panels:

- Breakpoints:** Located at the top, it shows a single breakpoint set at line 26 of MainActivity.java for the `onCreate(Bundle)` method.
- Outline:** Located at the bottom right, it displays a tree view of the project structure. The `StringListAdapter` class is selected, showing its methods: `onCreate(Bundle) : void`, `onListItemClick(ListView, View, int, long) : void`, and `onCreateOptionsMenu(Menu) : boolean`.
- Source Editor:** The bottom left shows a snippet of Java code: `id.R.id.text1, items);`.
- Logcat:** On the far left, a portion of the log is visible, showing lines 2140, 2226, and 135.

Stopping at a breakpoint in the *onCreate* method

Debug - Countries/src/com/vogella/android/countries/MainActivity.java - ADT - /Users/stg/Documents/T

Debug

Thread [<1> main] (Suspended (breakpoint at line 26 in MainActivity))

- <VM does not provide monitor information>
- MainActivity.onCreate(Bundle) line: 26
- MainActivity(Activity).performCreate(Bundle) line: 5243
- Instrumentation.callActivityOnCreate(Activity, Bundle) line: 1087
- ActivityThread.performLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2140
- ActivityThread.handleLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2226
- ActivityThread.access\$5700(ActivityThread, ActivityThread\$ActivityClientRecord, Intent) line: 135
- ActivityThread\$H.handleMessage(Message) line: 1397
- ActivityThread\$H(Handler).dispatchMessage(Message) line: 102
- Looper.loop() line: 137
- ActivityThread.main(String[]) line: 4998
- Method.invokeNative(Object, Object[], Class, Class[], Class, int, boolean) line: not available [native method]

Variables Breakpoints

MainActivity [line: 26] - onCreate(B...

project.properties MainActivity.java

```
        super(context, android.R.layout.simple_list_item_1, android.R.id.text1, items);
    }

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

        // get list of countries
        List<String> countries = CountryUtils.getCountries();

        // create and assign list adapter
        setListAdapter(new StringListAdapter(this, countries));
    }

    protected void onListItemClick(ListView l, View v, int position, long id) {
        String item = (String) getListAdapter().getItem(position);
        Toast.makeText(this, "Selected: " + item, Toast.LENGTH_SHORT);
    }
}
```


Assigning a value to the list of countries

Debug - Countries/src/com/vogella/android/countries/MainActivity.java - ADT - /Users/stg/Docum

Debug

- Thread [<1> main] (Suspended)
 - <VM does not provide monitor information>
 - MainActivity.onCreate(Bundle) line: 29
 - MainActivity(Activity).performCreate(Bundle) line: 5243
 - Instrumentation.callActivityOnCreate(Activity, Bundle) line: 1087
 - ActivityThread.performLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2140
 - ActivityThread.handleLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2226
 - ActivityThread.access\$5700(ActivityThread, ActivityThread\$ActivityClientRecord, Intent) line: 135
 - ActivityThread\$H.handleMessage(Message) line: 1397
 - ActivityThread\$H(Handler).dispatchMessage(Message) line: 102
 - Looper.loop() line: 137
 - ActivityThread.main(String[]) line: 4998
 - Method.invokeNative(Object, Object[], Class, Class[], Class, int, boolean) line: not available [native method]

Variables Breakpoints

- MainActivity [line: 26] - on

project.properties MainActivity.java

```
        super(context, android.R.layout.simple_list_item_1, android.R.id.text1, items);
    }
}

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

    // get list of countries
    List<String> countries = CountryUtils.getCountries();

    // create and assign list adapter
    setListAdapter(new StringListAdapter(this, countries));
}

protected void onListItemClick(ListView l, View v, int position, long id) {
    String item = (String) getListAdapter().getItem(position);
```

Inspecting the value of the *countries* variable: an array of empty strings ["", "", "", "", "", ..., ""]

The screenshot shows an IDE window for MainActivity.java. The Variables panel displays the following structure:

Name	Value
this	MainActivity (id=831945308166)
savedInstanceState	null
countries	ArrayList (id=831945446560)
array	Object[202] (id=831945508464)
[0]	"" (id=831942433832)
count	0
hashCode	0
offset	0
value	(id=831942433864)
[1]	"" (id=831942433832)

The Outline panel shows the following class structure:

- com.vogella.android.countries
 - MainActivity
 - StringListAdapter
 - onCreate(Bundle) : void
 - onListItemClick(ListView, View, int, long) : void
 - onCreateOptionsMenu(Menu) : boolean

Problem identified: Restart the debugging session

Debug – Countries/src/com/vogella/android/countries/MainActivity.java – ADT – /Users/stg/Docume

Debug

Thread [<1> main] (Suspended)

- <VM does not provide monitor information>
- MainActivity.onCreate(Bundle) line: 32
- MainActivity(Activity).performCreate(Bundle) line: 5243
- Instrumentation.callActivityOnCreate(Activity, Bundle) line: 1087
- ActivityThread.performLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2140
- ActivityThread.handleLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2226
- ActivityThread.access\$5700(ActivityThread, ActivityThread\$ActivityClientRecord, Intent) line: 135
- ActivityThread\$H.handleMessage(Message) line: 1397
- ActivityThread\$H(Handler).dispatchMessage(Message) line: 102
- Looper.loop() line: 137
- ActivityThread.main(String[]) line: 4998
- Method.invokeNative(Object, Object[], Class, Class[], Class, int, boolean) line: not available [native method]

Variables

Name

- trns
- savedInstanceState
- countries
 - array
 - [0...99]
 - [0]
 - count
 - hashCode
 - offset
 - value
 - [1]

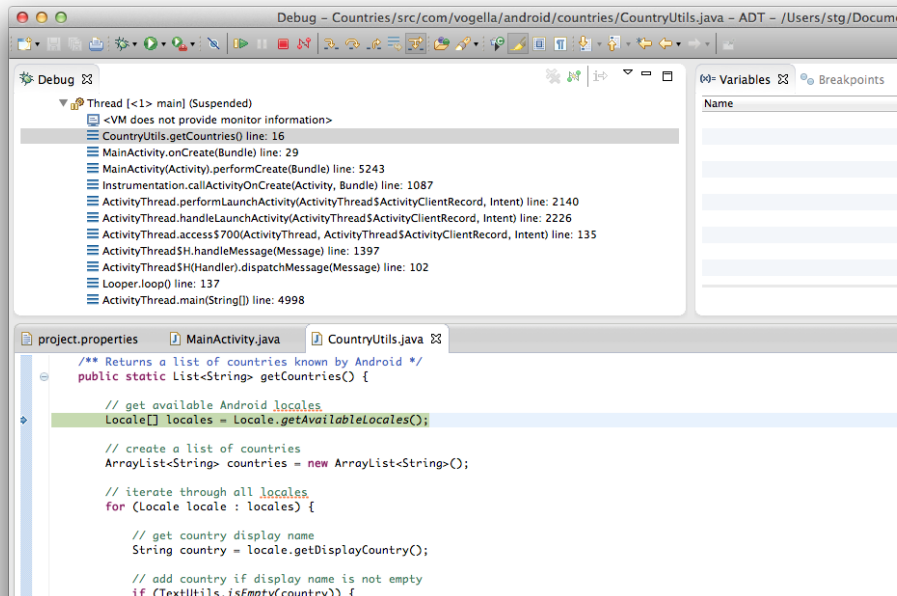
project.properties MainActivity.java

```
super(context, android.R.layout.simple_list_item_1, android.R.id.text1, items);
}
}
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    // get list of countries
    List<String> countries = CountryUtils.getCountries();

    // create and assign list adapter
    setListAdapter(new StringListAdapter(this, countries));
}
protected void onListItemClick(ListView l, View v, int position, long id) {
    String item = (String) getListAdapter().getItem(position);
```

Entering the getCountries method



Debug – Countries/src/com/vogella/android/countries/CountryUtils.java – ADT – /Users/stg/Docum

Debug

Thread [<1> main] (Suspended)

- <VM does not provide monitor information>
- CountryUtils.getCountries() line: 16
- MainActivity.onCreate(Bundle) line: 29
- MainActivity(Activity).performCreate(Bundle) line: 5243
- Instrumentation.callActivityOnCreate(Activity, Bundle) line: 1087
- ActivityThread.performLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2140
- ActivityThread.handleLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2226
- ActivityThread.access\$700(ActivityThread, ActivityThread\$ActivityClientRecord, Intent) line: 135
- ActivityThread\$H.handleMessage(Message) line: 1397
- ActivityThread\$H(Handler).dispatchMessage(Message) line: 102
- Looper.loop() line: 137
- ActivityThread.main(String[]) line: 4998

Variables Breakpoints

Name

project.properties MainActivity.java CountryUtils.java

```
/** Returns a list of countries known by Android */
public static List<String> getCountries() {

    // get available Android locales
    Locale[] locales = Locale.getAvailableLocales();

    // create a list of countries
    ArrayList<String> countries = new ArrayList<String>();

    // iterate through all locales
    for (Locale locale : locales) {

        // get country display name
        String country = locale.getDisplayCountry();

        // add country if display name is not empty
        if (TextUtils.isEmpty(country)) {
```

Continue executing, looking for updates to countries

Debug – Countries/src/com/vogella/android/countries/CountryUtils.java – ADT – /Users/stg/Docum

Debug

Thread [<1> main] (Suspended)

- <VM does not provide monitor information>
- CountryUtils.getCountries() line: 22
- MainActivity.onCreate(Bundle) line: 29
- MainActivity(Activity).performCreate(Bundle) line: 5243
- Instrumentation.callActivityOnCreate(Activity, Bundle) line: 1087
- ActivityThread.performLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2140
- ActivityThread.handleLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2226
- ActivityThread.access\$700(ActivityThread, ActivityThread\$ActivityClientRecord, Intent) line: 135
- ActivityThread\$H.handleMessage(Message) line: 1397
- ActivityThread\$H(Handler).dispatchMessage(Message) line: 102
- Looper.loop() line: 137
- ActivityThread.main(String[]) line: 4998

Variables

Name

- locales
 - [0...99]
 - [0]
 - cachedToStringResult
 - countryCode
 - languageCode
 - variantCode
 - [1]
 - cachedToStringResult
 - countryCode
 - languageCode
- af_NA

project.properties MainActivity.java CountryUtils.java

```
/** Returns a list of countries known by Android */
public static List<String> getCountries() {

    // get available Android locales
    Locale[] locales = Locale.getAvailableLocales();

    // create a list of countries
    ArrayList<String> countries = new ArrayList<String>();

    // iterate through all locales
    for (Locale locale : locales) {

        // get country display name
        String country = locale.getDisplayCountry();

        // add country if display name is not empty
        if (TextUtils.isEmpty(country)) {
```

Continue execution

Debug – Countries/src/com/vogella/android/countries/CountryUtils.java – ADT – /Users/stg/Docum

Debug

Thread [<1> main] (Suspended)

- <VM does not provide monitor information>
- CountryUtils.getCountries() line: 25
- MainActivity.onCreate(Bundle) line: 29
- MainActivity(Activity).performCreate(Bundle) line: 5243
- Instrumentation.callActivityOnCreate(Activity, Bundle) line: 1087
- ActivityThread.performLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2140
- ActivityThread.handleLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2226
- ActivityThread.access\$700(ActivityThread, ActivityThread\$ActivityClientRecord, Intent) line: 135
- ActivityThread\$H.handleMessage(Message) line: 1397
- ActivityThread\$H(Handler).dispatchMessage(Message) line: 102
- Looper.loop() line: 137
- ActivityThread.main(String[]) line: 4998

Variables

Name

- locales
 - [0...99]
 - [0]
 - cachedToStringResult
 - countryCode
 - languageCode
 - variantCode
 - [1]
 - cachedToStringResult
 - countryCode
 - languageCode
- af_NA

project.properties MainActivity.java CountryUtils.java

```
/** Returns a list of countries known by Android */
public static List<String> getCountries() {

    // get available Android locales
    Locale[] locales = Locale.getAvailableLocales();

    // create a list of countries
    ArrayList<String> countries = new ArrayList<String>();

    // iterate through all locales
    for (Locale locale : locales) {

        // get country display name
        String country = locale.getDisplayCountry();

        // add country if display name is not empty
        if (TextUtils.isEmpty(country)) {
```

A simple logic error is discovered: we want the country name if it is *not* empty

Debug - Countries/src/com/vogella/android/countries/CountryUtils.java - ADT - /Users/stg/Docum

Debug

Thread [<1> main] (Suspended)
<VM does not provide monitor information>
CountryUtils.getCountries() line: 28
MainActivity.onCreate(Bundle) line: 29
MainActivity(Activity).performCreate(Bundle) line: 5243
Instrumentation.callActivityOnCreate(Activity, Bundle) line: 1087
ActivityThread.performLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2140
ActivityThread.handleLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2226
ActivityThread.access\$700(ActivityThread, ActivityThread\$ActivityClientRecord, Intent) line: 135
ActivityThread\$H.handleMessage(Message) line: 1397
ActivityThread\$H(Handler).dispatchMessage(Message) line: 102
Looper.loop() line: 137
ActivityThread.main(String[]) line: 4998

Variables

Name

locales

[0...99]

[0]

cachedToStringResult

countryCode

languageCode

variantCode

[1]

cachedToStringResult

countryCode

languageCode

af_NA

project.properties MainActivity.java CountryUtils.java

```
// create a list of countries
ArrayList<String> countries = new ArrayList<String>();

// iterate through all locales
for (Locale locale : locales) {

    // get country display name
    String country = locale.getDisplayCountry();

    // add country if display name is not empty
    if (TextUtils.isEmpty(country)) {
        countries.add(country);
    }
}
```

We are not able to swap in the new version of the code

Debug - Countries/src/com/vogella/android/countries/CountryUtils.java - ADT - /Users/stg/Documents/Teaching/SELP/2013/Workspa

Q Quick


> main (Suspended)
es not provide monitor information>
Utils.getCountries() line: 28
ivity.onCreate(Bundle) line: 29
ivity(Activity).performCreate(Bundle) line: 5243
entation.callActivityOnCreate(Activity, Bundle) line: 1087
Thread.performLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2140
Thread.handleLaunchActivity(ActivityThread\$ActivityClientRecord, Intent) line: 2226
Thread.access\$700(ActivityThread, ActivityThread\$ActivityClientRecord, Intent) line: 135
Thread\$H.handleMessage(Message) line: 1397
Thread\$H(Handler).dispatchMessage(Message) line: 102
oop() line: 137
Thread.main(String[]) line: 4998

Name	Value
locales	Locale[597] (id=831945)
[0...99]	
[0]	Locale (id=831945)
cachedToStringResult	"af" (id=831945)
countryCode	"" (id=831942)
languageCode	"af" (id=831945)
variantCode	"" (id=831942)
[1]	Locale (id=831945)
cachedToStringResult	"af_NA" (id=831945)
countryCode	"NA" (id=831942)
languageCode	"af" (id=831945)
variantCode	"" (id=831942)

MainActivity.java CountryUtils.java

```
ate a list of countries  
ist<String> countries = new ArrayList<String>  
  
rate through all locales  
ocale locale : locales) {  
  
get country display name  
ring country = locale.getDisplayCountry();  
  
add country if display name is not empty  
(!TextUtils.isEmpty(country)) {  
countries.add(country);  
  
countries;
```

Hot Code Replace Failed

 Some code changes cannot be hot swapped into a running virtual machine, such as changing method names or introducing errors into running code.

The current target virtual machine [DalvikVM[localhost:8704]] from launch [Countries] was unable to replace the running code with the code in the workspace.

It is safe to continue running the application, but you may notice discrepancies when debugging this application.

Reason:
The target VM does not support hot code replace

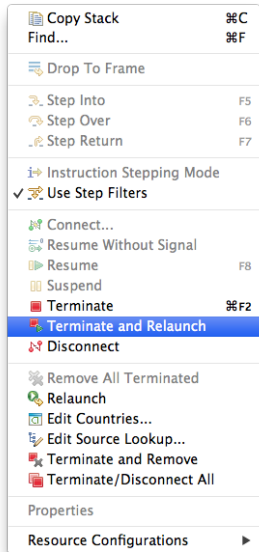
Do not show error when hot code replace is not supported

Terminate Disconnect Continue

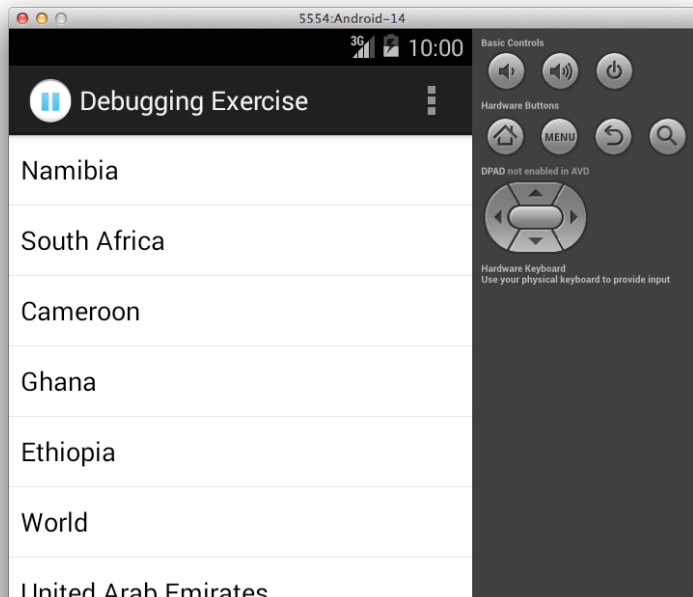
Outline

- com.v
- Count
- get

We can terminate and relaunch this debugging session



Success: a list of countries is displayed



Software testing

A test framework called Robotium is available for testing Android applications. Using Robotium we can automate tests of Android apps, and collect statistics on which tests passed and which tests failed, using the JUnit test framework.

Import Robotium and make an instance of Solo

The screenshot shows the Eclipse IDE interface. The Package Explorer on the left displays the project structure: TestProject > src > com.testcalculator > TestApk.java > TestApk. The main editor shows the following Java code:

```
package com.testcalculator;

import com.jayway.android.robotium.solo.Solo;
import android.test.ActivityInstrumentationTestCase2;

@SuppressWarnings("unchecked")
public class TestApk extends ActivityInstrumentationTestCase2{

    private static final String TARGET_PACKAGE_ID="com.calculator";
    private static final String LAUNCHER_ACTIVITY_FULL_CLASSNAME="com.calculator.Main";
    private static Class launcherActivityClass;

    static{

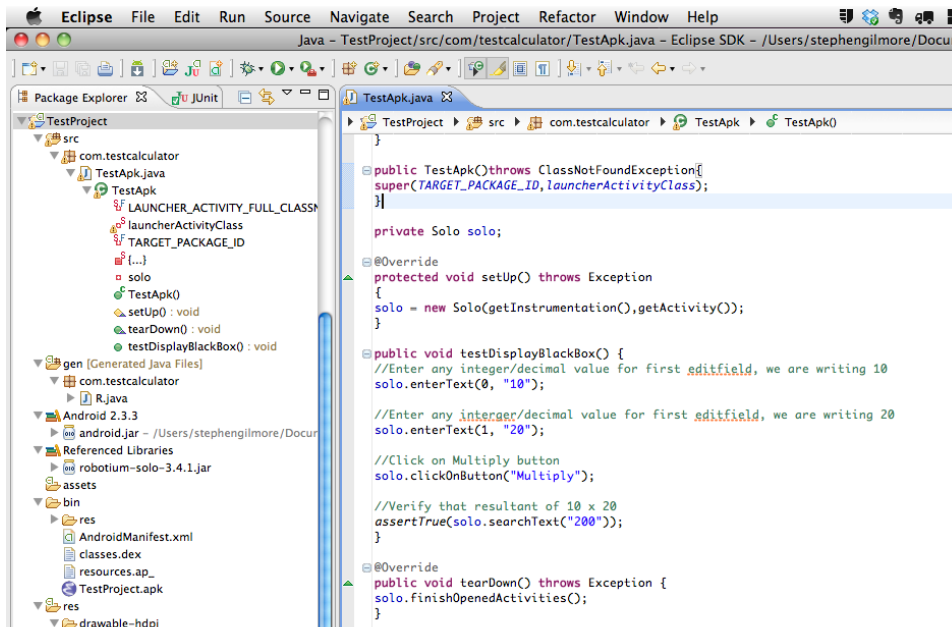
        try
        {
            launcherActivityClass = Class.forName(LAUNCHER_ACTIVITY_FULL_CLASSNAME);
        } catch (ClassNotFoundException e){
            throw new RuntimeException(e);
        }
    }

    public TestApk()throws ClassNotFoundException{
        super(TARGET_PACKAGE_ID, launcherActivityClass);
    }

    private Solo solo;

    @Override
    protected void setUp() throws Exception
    {
        solo = new Solo(getInstrumentation(),getActivity());
    }
}
```

Write a void method whose name begins with "test"



The screenshot shows the Eclipse IDE interface. The Package Explorer on the left displays the project structure: TestProject > src > com.testcalculator > TestApk.java > TestApk. The TestApk.java file is open in the editor, showing the following code:

```
public TestApk() throws ClassNotFoundException {
    super(TARGET_PACKAGE_ID, launcherActivityClass);
}

private Solo solo;

@Override
protected void setUp() throws Exception {
    solo = new Solo(getInstrumentation(), getActivity());
}

public void testDisplayBlackBox() {
    //Enter any integer/decimal value for first editfield, we are writing 10
    solo.enterText(0, "10");

    //Enter any integer/decimal value for first editfield, we are writing 20
    solo.enterText(1, "20");

    //Click on Multiply button
    solo.clickOnButton("Multiply");

    //Verify that resultant of 10 x 20
    assertTrue(solo.searchText("200"));
}

@Override
public void tearDown() throws Exception {
    solo.finishOpenedActivities();
}
```

Run as an Android JUnit test

The screenshot shows the Eclipse IDE interface. The Package Explorer on the left displays the project structure: TestProject > src > com.testcalculator > TestApk > TestApk(). A context menu is open over the TestApk.java file, listing various actions. The 'Run As' option is selected, and its sub-menu is visible, showing four options: 1 Android Application, 2 Android JUnit Test (highlighted), 3 Java Applet, and 4 Java Application. The main editor window shows the code for TestApk.java, including the setUp() and testDisplayBlackBox() methods.

```
public TestApk() throws ClassNotFoundException {
    (TARGET_PACKAGE_ID, launcherActivityClass);
}

Solo solo;

@Override
protected void setUp() throws Exception {
    new Solo(getInstrumentation(), getActivity());
}

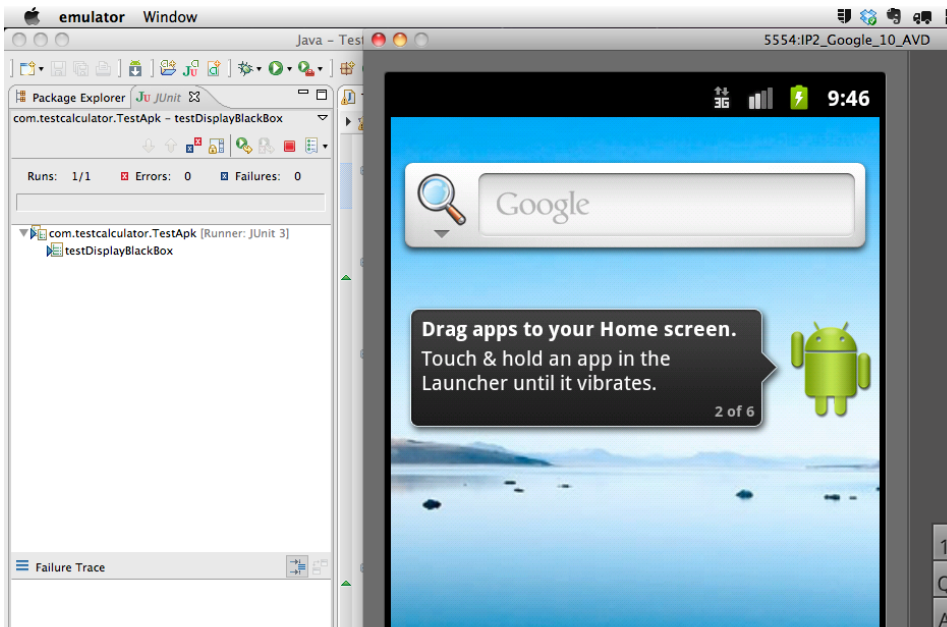
void testDisplayBlackBox() {
    for any integer/decimal value for first editfield, we are writing 10
    enterText(0, "10");

    for any integer/decimal value for first editfield, we are writing 20
    enterText(1, "20");

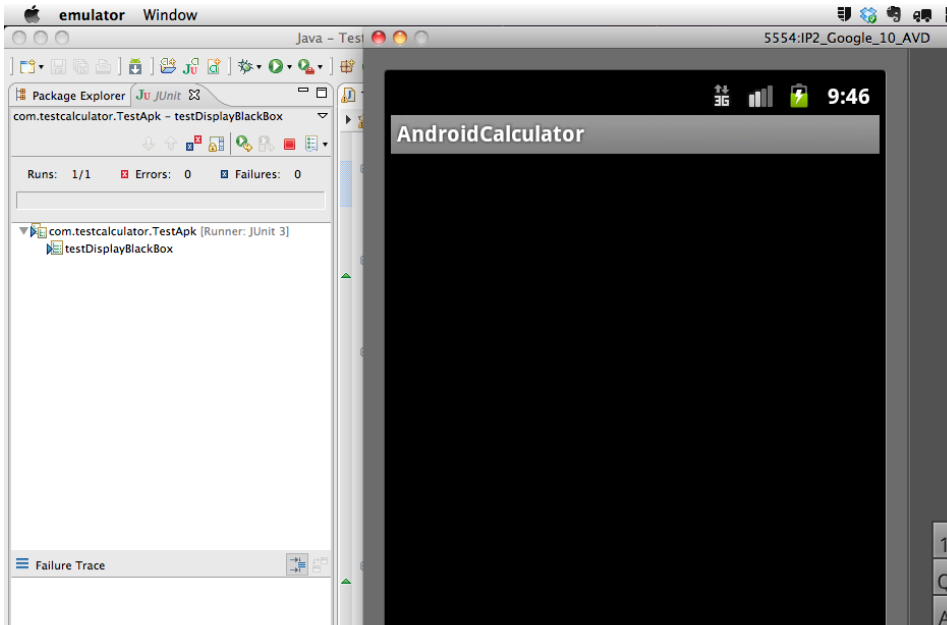
    click on Multiply button
    clickOnButton("Multiply");

    verify that resultant of 10 x 20
    assertTrue(solo.searchText("200"));
}
```

The emulator appears



The app under test launches



Values are entered in the fields

The image shows a screenshot of an Android emulator window. The window title is "emulator Window" and the system bar at the top right shows "5554:IP2_Google_10_AVD". The emulator displays an app titled "AndroidCalculator" with the instruction "Enter two values and click on Calculate to multiply them." Below this, there are two input fields: "Enter First Number" and "Enter Second Number". The first input field is highlighted with an orange border. Below the second input field, the value "0.00" is displayed. At the bottom of the screen, there is a "Multiply" button. On the left side of the emulator, the Package Explorer shows the project structure for "com.testcalculator.TestApk - testDisplayBlackBox". The Package Explorer shows a tree view with "com.testcalculator.TestApk [Runner: JUnit 3]" and "testDisplayBlackBox". The Package Explorer also shows "Runs: 1/1", "Errors: 0", and "Failures: 0". At the bottom left of the Package Explorer, there is a "Failure Trace" section.

These values have been entered programmatically

The image shows a screenshot of an Android emulator window. The window title is "emulator Window" and the system bar at the top shows "Java - Test" and "5554:IP2_Google_10_AVD". The emulator displays an application titled "AndroidCalculator" with the following text: "Enter two values and click on Calculate to multiply them." Below this, there are two input fields. The first field is labeled "Enter First Number" and contains the value "10". The second field is labeled "Enter Second Number" and contains the value "20". Below these fields, the value "0.00" is displayed. At the bottom of the screen, there is a button labeled "Multiply".

The left side of the image shows the IDE interface. The Package Explorer displays the package structure for "com.testcalculator.TestApk - testDisplayBlackBox". The Run console shows the following output:

```
Runs: 1/1  Errors: 0  Failures: 0
```

Below the Run console, the test runner is shown as "com.testcalculator.TestApk [Runner: JUnit 3]" with a sub-entry for "testDisplayBlackBox".

The button click also is performed for us

The image shows a screenshot of an Android emulator window titled "emulator Window". The window displays the "AndroidCalculator" app interface. The app has a dark background with white text and input fields. The text reads: "Enter two values and click on Calculate to multiply them.", "Enter First Number", "Enter Second Number", and "200.0". There are two input fields, one containing "10" and another containing "20". A large "Multiply" button is visible at the bottom. The emulator's status bar at the top shows the time as 9:46, signal strength, and battery level. The Package Explorer on the left shows the test results for "com.testcalculator.TestApk - testDisplayBlackBox", indicating 1/1 runs, 0 errors, and 0 failures. The test method "testDisplayBlackBox" is listed under the runner "JUnit 3".

emulator Window

Java - Test

5554:IP2_Google_10_AVD

Package Explorer JUnit

com.testcalculator.TestApk - testDisplayBlackBox

Runs: 1/1 Errors: 0 Failures: 0

com.testcalculator.TestApk [Runner: JUnit 3]

testDisplayBlackBox

Failure Trace

AndroidCalculator

Enter two values and click on Calculate to multiply them.

Enter First Number

10

Enter Second Number

20

200.0

Multiply

All tests passed (there was only one)

The image shows a screenshot of an Android emulator window. The window title is "emulator Window". The top status bar shows "Java - Test" and "5554:IP2_Google_10_AVD". The main content area displays the Google search bar and a tutorial bubble that says "Drag apps to your Home screen. Touch & hold an app in the Launcher until it vibrates." with a green Android robot icon and "2 of 6".

On the left side, the Package Explorer shows the test results for the JUnit runner. The text reads: "Finished after 11.394 seconds". Below this, it shows "Runs: 1/1", "Errors: 0", and "Failures: 0". A green progress bar is visible. The test class is identified as "com.testcalculator.TestApk [Runner: JUnit 3] (11.85)".

At the bottom left, there is a "Failure Trace" section which is currently empty.

If some failed, we can see which

The image shows a screenshot of an Android emulator window. The window title is "emulator Window". The top status bar shows "Java - Test" and "5554:IP2_Google_10_AVD". The Package Explorer on the left shows a test run for "com.testcalculator.TestApk" with the following details:

- Finished after 11.394 seconds
- Runs: 1/1
- Errors: 0
- Failures: 0

The test run tree shows:

- com.testcalculator.TestApk [Runner: JUnit 3] (11.85 s)
 - testDisplayBlackBox (11.851 s)

The bottom of the Package Explorer shows a "Failure Trace" section.

The emulator screen displays a home screen with a Google search bar at the top. A tutorial bubble is visible with the text:

Drag apps to your Home screen.
Touch & hold an app in the Launcher until it vibrates.

The bubble also indicates "2 of 6" steps and features the Android logo.

Software testing

A testing framework such as Robotium allows us to formally document a series of tests which we expect our app to pass. These are embedded in the code of the test harness. After every change to our code we can re-run the test harness and check that all tests still pass.

Automating testing in this way can turn a boring manual task which is a chore to do into a simple automatic task which is easy to re-run and operates entirely without human intervention.