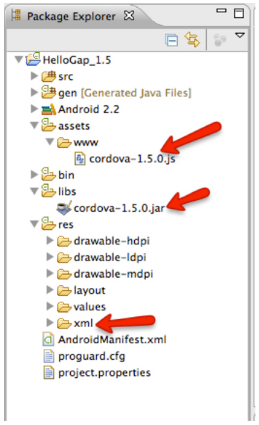
How to use PhoneGap with Eclipse and the Android SDK

1. Create a new Android Application.

2. Create an assets/www directory inside the new Android project. All of the HTML and JavaScript for your PhoneGap application interface will reside within the assets/www folder.

3. To copy the required files for PhoneGap into the project, first locate the directory where you downloaded PhoneGap, and navigate to the lib/android subdirectory.

4. Copy cordova-2.0.0.js to the assets/www directory within your Android project (/assets/www).
Copy cordova-2.0.0.jar to the libs directory within your Android project (/libs).
Copy the xml directory into the res directory within your Android project (/res/xml/).



5. Next, create a file named index.html in the assets/www folder. This file will be used as the main entry point for your PhoneGap application's interface. In index.html, add the following HTML code to act as a starting point for your user interface development:

```
<!DOCTYPE HTML>
<html>
<head>
<title>PhoneGap</title>
 <script type="text/javascript" charset="utf-8" src="cordova-1.5.0.js">
 </script>
 </head>
 <body>
 <h1>Hello PhoneGap</h1>
 </body>
 </html>
```

6. You will need to add the cordova-2.0.0.jar library to the build path for the Android project. Right-click cordova-2.0.0.jar and select Build Path > Add To Build Path

7. Open your main application Activity file. This file will have the same name as your project, followed by the word "Activity". It will be located under the src folder in the project package that you specified earlier in this process.

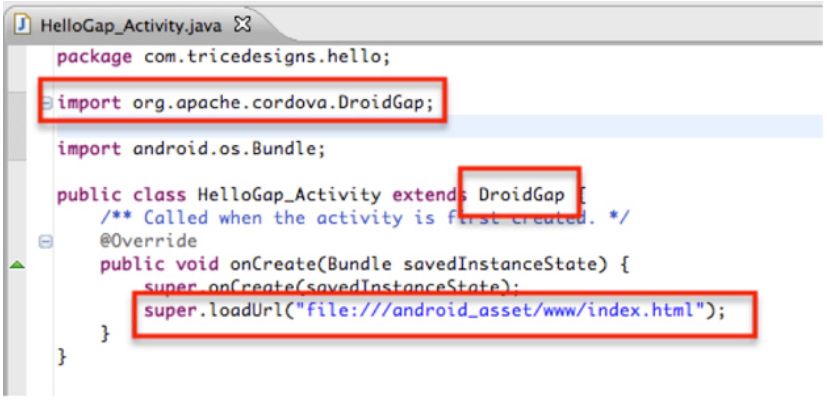
8. In the main Activity class, add an import statement for org.apache.cordova.DroidGap: import org.apache.cordova.DroidGap;

9. Change the base class from Activity to DroidGap. This is in the class definition following the word extends. The finished class definition looks like:

public class MainActivity extends DroidGap {

10. Replace the call to setContentView() with a reference to load the PhoneGap interface from the local assets/www/index.html file, which you created earlier. The finished line of code looks like:

REMOVE: setContentView(R.layout.activity_main);
REPLACE WITH: super.loadUrl("file:///android_asset/www/index.html");



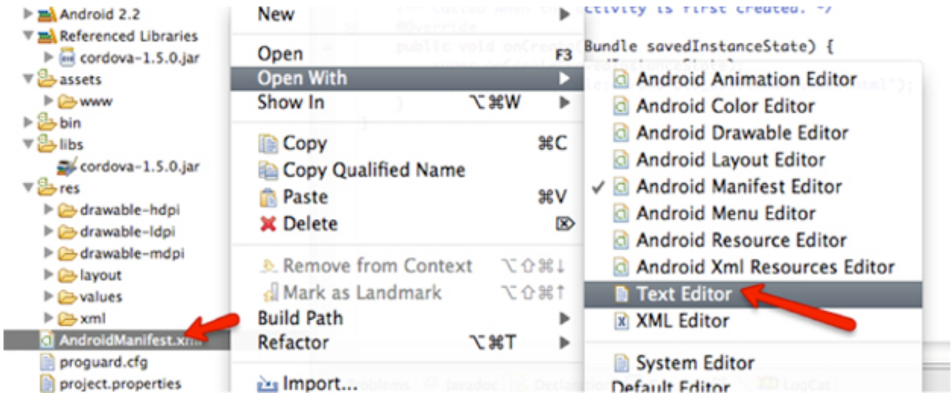
```
package com.tricedesigns.hello;

import org.apache.cordova.DroidGap;

import android.os.Bundle;

public class HelloGap_Activity extends DroidGap {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        super.loadUrl("file:///android_asset/www/index.html");
    }
}
```

11. You have now configured the files within your Android project to use PhoneGap. The last step is to configure the project metadata to enable PhoneGap to run. Begin by opening the AndroidManifest.xml file in your project root. Use the Eclipse text editor by right-clicking the AndroidManifest.xml file and selecting Open With > Text Editor.



12. Copy the following <uses-permission> XML nodes and paste them as children of the root <manifest> node in the AndroidManifest.xml file:

<uses-permission android:name="android.permission.VIBRATE" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_LOCATION_EXTRA_COMMANDS" />

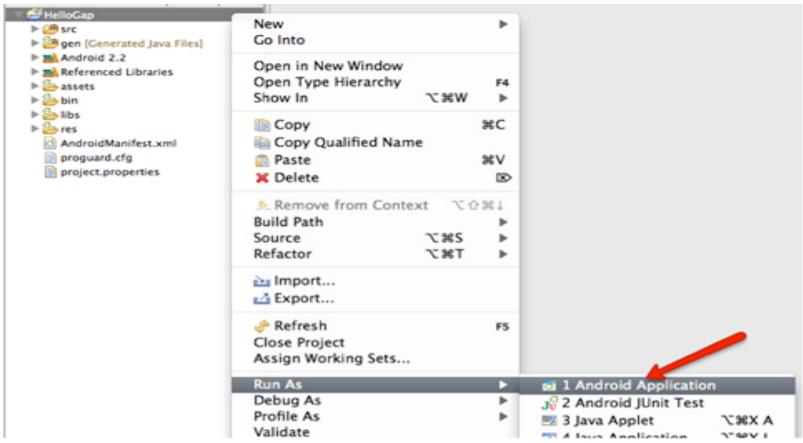
```
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.RECEIVE_SMS" />
<uses-permission android:name="android.permission.RECORD_AUDIO" />
<uses-permission android:name="android.permission.MODIFY_AUDIO_SETTINGS"
/>
<uses-permission android:name="android.permission.READ_CONTACTS" />
<uses-permission android:name="android.permission.WRITE_CONTACTS" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"
/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.GET_ACCOUNTS" />
<uses-permission android:name="android.permission.BROADCAST_STICKY" />
```

The <uses-permission> XML values identify the features that you want to be enabled for your application. The lines above enable all permissions required for all features of PhoneGap to function. After you have built your application, you may want to remove any permissions that you are not actually using; this will remove security warnings during application installation. To read more about Android permissions and the <uses-permission> element, visit the Android developer topic on the uses-permission element.

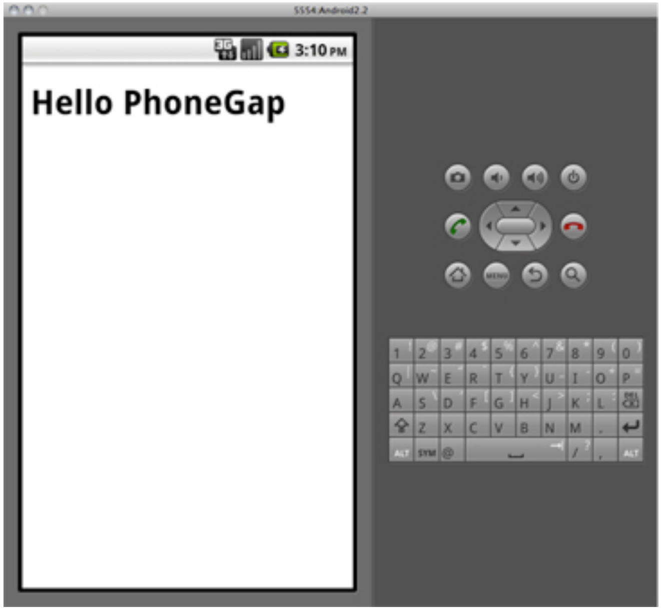
13. After you have configured application permissions, you need to modify the existing <activity> node. Locate the <activity> node, which is a child of the <application> XML node. Add the following attribute to the <activity> node:

```
android:configChanges="orientation|keyboardHidden|screenSize"
```

14. You are done! To launch your PhoneGap application in the Android emulator, right-click the project root, and select Run As > Android Application



15. Eclipse will automatically start an Android emulator instance (if one is not already running), deploy your application to the emulator, and launch the application.



Hello PhoneGap