PhoneGap

Writing iPhone and Android applications in JavaScript, HTML and CSS

What is PhoneGap?

- Mobile development framework
  - iPhoneDevCamp, San Francisco 2008
  - Nitobi Software
- Bridges the gap between SDKs
- Open source
  - MIT License

Phone support

	iOS	Android	Blackberry	Symbian	Palm
Geo Location	Yes	Yes	Yes	Yes	Yes
Vibration	Yes	Yes	Yes	Yes	Yes
Camera	Yes	Yes	Yes	Yes	No
Accelerometer	Yes	Yes	OS 4.7	Yes	Yes
Sound	Yes	Yes	Yes	Yes	Yes
Contact Support	Yes	Yes	Yes	Yes	N/A

How does it work?

- Mobile SDKs ship a browser
  - UIWebView object for iPhone
  - WebView object with Android
- Interpreted language
  - JavaScript
- Deploy over standard means
  - App Store
  - Android market

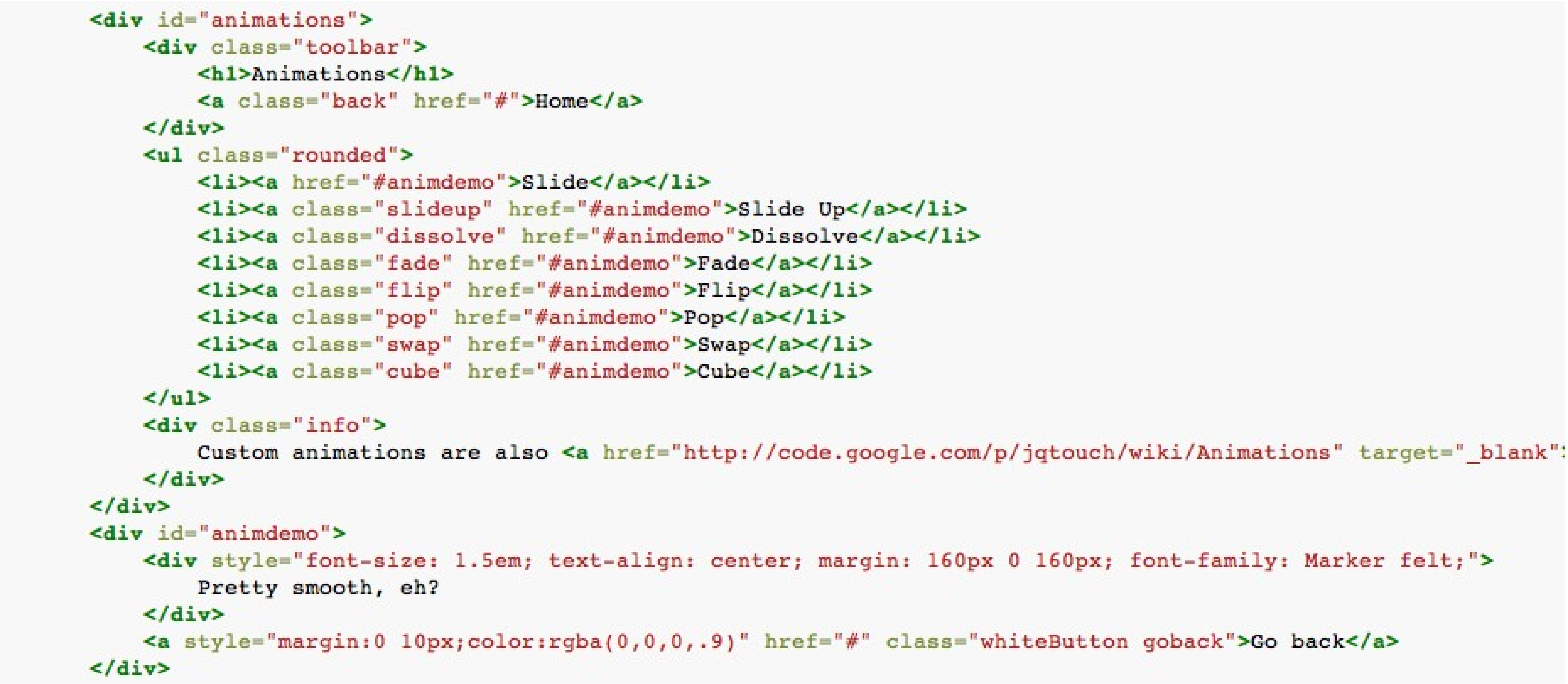
Writing Your Application

- WebKit
  - Android / iPhone
  - Blackberry 6+ / Palm WebOS
- HTML5
  - Canvas, Databases and Geolocation
- CSS3
  - Transitions, Gradients and Rounded Corners
- Cross-domain security policy does not apply
  - $.getJSON('http://search.twitter.com/trends/current.json')

jQTouch

- A jQuery plugin for mobile web development
- For iOS
- Supports:
  - Page Transitions (CSS 3)
  - Page History
  - Toolbar
  - Forms
  - Home screen Icons
- MIT Licensed

jQTouch Pages



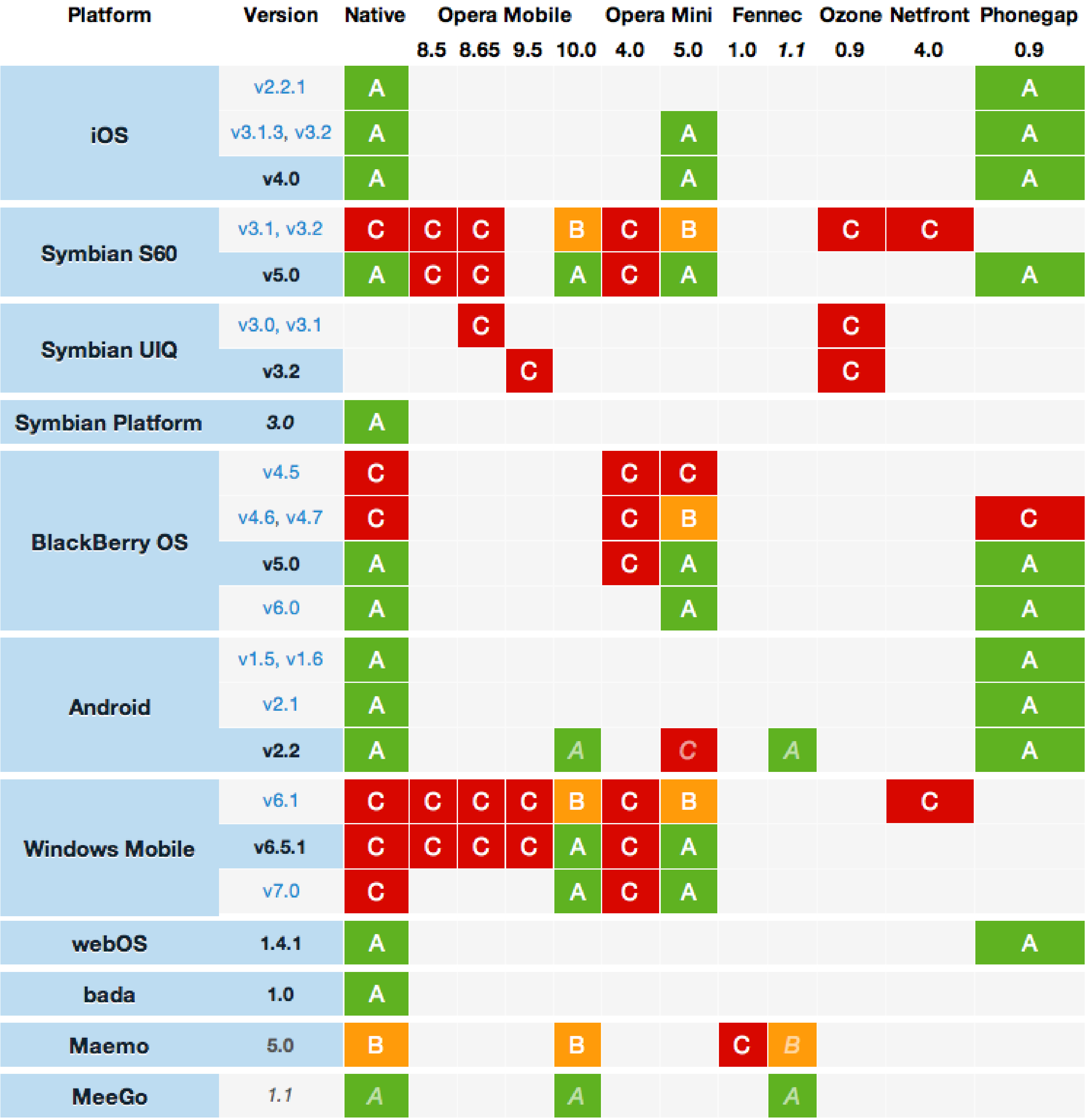
```
<div id="animations">
    <div class="toolbar">
        <h1>Animations</h1>
        <a class="back" href="#">Home</a>
    </div>
    <ul class="rounded">
        <li><a href="#animdemo">Slide</a></li>
        <li><a class="slideup" href="#animdemo">Slide Up</a></li>
        <li><a class="dissolve" href="#animdemo">Dissolve</a></li>
        <li><a class="fade" href="#animdemo">Fade</a></li>
        <li><a class="flip" href="#animdemo">Flip</a></li>
        <li><a class="pop" href="#animdemo">Pop</a></li>
        <li><a class="swap" href="#animdemo">Swap</a></li>
        <li><a class="cube" href="#animdemo">Cube</a></li>
    </ul>
    <div class="info">
        Custom animations are also <a href="http://code.google.com/p/jqtouch/wiki/Animations" target="_blank"
    </div>
</div>
<div id="animdemo">
    <div style="font-size: 1.5em; text-align: center; margin: 160px 0 160px; font-family: Marker felt;">
        Pretty smooth, eh?
    </div>
    <a style="margin:0 10px;color:rgba(0,0,0,.9)" href="#" class="whiteButton goback">Go back</a>
</div>
```

iScroll

- Provides a scroll content inside a fixed width/height element
- Android >=1.5, iPad >=3.2, iPhone >=2.0
- Uses touch JavaScript events:
  - touchstart
  - touchmove
  - touchend
- Will work on desktop also
- MIT / GPL dual licensed

jQuery Mobile

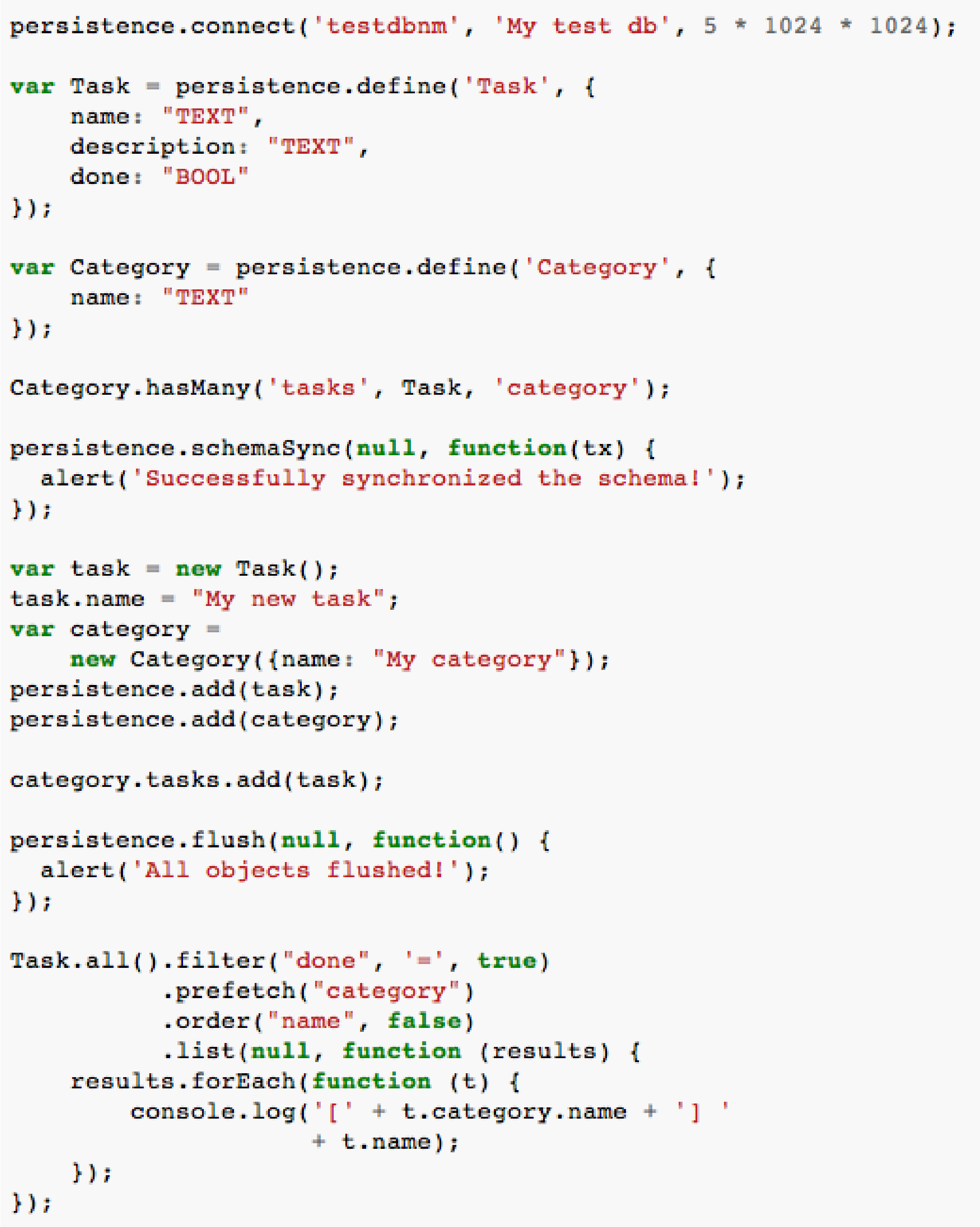
- Part of jQuery project
- Similar to jQTouch with more device support
- New project (alpha)
- MIT / GPL dual licensed



Platform	Version	Native	Opera Mobile				Opera Mini		Fennec		Ozone	Netfront	Phonegap
			8.5	8.65	9.5	10.0	4.0	5.0	1.0	*1.1*	0.9	4.0	0.9
iOS	v2.2.1	A											A
	v3.1.3, v3.2	A						A					A
	v4.0	A						A					A
Symbian S60	v3.1, v3.2	C	C	C		B	C	B			C	C	
	v5.0	A	C	C		A	C	A					A
Symbian UIQ	v3.0, v3.1			C							C		
	v3.2				C						C		
Symbian Platform	*3.0*	A											
BlackBerry OS	v4.5	C					C	C					
	v4.6, v4.7	C					C	B					C
	v5.0	A					C	A					A
	v6.0	A						A					A
Android	v1.5, v1.6	A											A
	v2.1	A											A
	v2.2	A				*A*		*C*		*A*			A
Windows Mobile	v6.1	C	C	C	C	B	C	B				C	
	v6.5.1	C	C	C	C	A	C	A					
	v7.0	C				A	C	A					
webOS	1.4.1	A											A
bada	1.0	A											
Maemo	5.0	B				B			C	*B*			
MeeGo	*1.1*	*A*				*A*				*A*			

Persistence.js

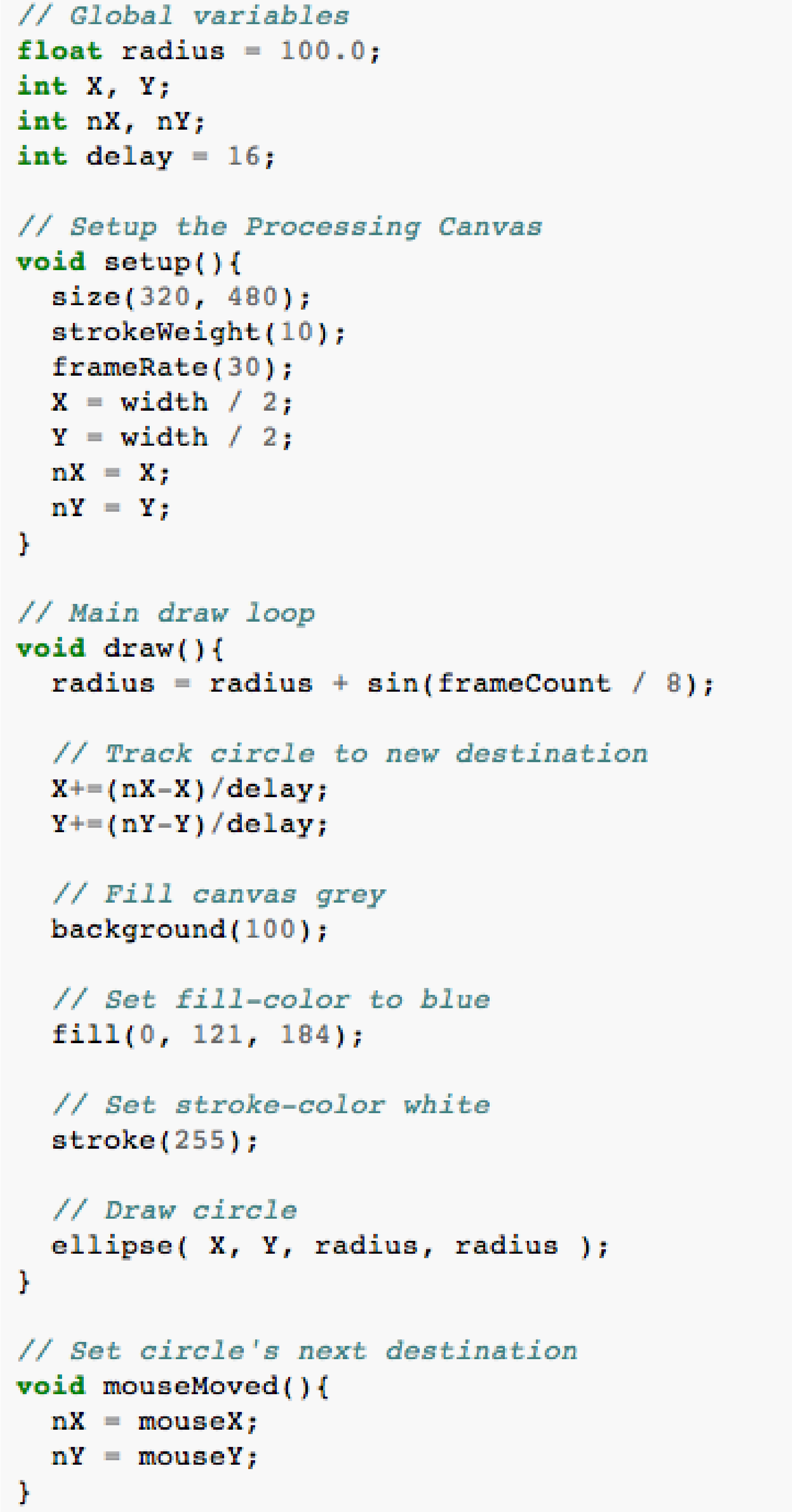
- Asynchronous Javascript object-relational mapper library
- It can be used in web browser:
  - HTML5 WebSQL database (WebKit)
  - Google Gears (Firefox / IE)
- Also on the server using node.js:
  - Node-mysql
- MIT Licensed



```
persistence.connect('testdbnm', 'My test db', 5 * 1024 * 1024);

var Task = persistence.define('Task', {
    name: "TEXT",
    description: "TEXT",
    done: "BOOL"
});

var Category = persistence.define('Category', {
    name: "TEXT"
});

Category.hasMany('tasks', Task, 'category');

persistence.schemaSync(null, function(tx) {
  alert('Successfully synchronized the schema!');
});

var task = new Task();
task.name = "My new task";
var category =
    new Category({name: "My category"});
persistence.add(task);
persistence.add(category);

category.tasks.add(task);

persistence.flush(null, function() {
  alert('All objects flushed!');
});

Task.all().filter("done", '=', true)
          .prefetch("category")
          .order("name", false)
          .list(null, function (results) {
    results.forEach(function (t) {
        console.log('[' + t.category.name + '] '
                    + t.name);
    });
});
```

Processing.js

- Port of the processing language to JavaScript
- Uses canvas element
- Good replacement for flash
- Can also be used directly as a JavaScript libary
- MIT Licensed



```
// Global variables
float radius = 100.0;
int X, Y;
int nX, nY;
int delay = 16;

// Setup the Processing Canvas
void setup(){
  size(320, 480);
  strokeWeight(10);
  frameRate(30);
  X = width / 2;
  Y = width / 2;
  nX = X;
  nY = Y;
}

// Main draw loop
void draw(){
  radius = radius + sin(frameCount / 8);

  // Track circle to new destination
  X+=(nX-X)/delay;
  Y+=(nY-Y)/delay;

  // Fill canvas grey
  background(100);

  // Set fill-color to blue
  fill(0, 121, 184);

  // Set stroke-color white
  stroke(255);

  // Draw circle
  ellipse( X, Y, radius, radius );
}

// Set circle's next destination
void mouseMoved(){
  nX = mouseX;
  nY = mouseY;
}
```

Meta Tags

- Viewport
  - changes the window size used when displaying on iOS / Android
  - <meta name="viewport" content="width=320, user-scalable=0" />

- Full-screen mode
  - <meta name="apple-mobile-web-app-capable" content="yes">

- Style of the status bar
  - <meta name="apple-mobile-web-app-status-bar-style" content="black">

Link Tags

- Add icon to bookmark
  - <link rel="apple-touch-icon" href="/custom_icon.png"/>
- Icon with no gloss
  - <link rel="apple-touch-icon-precomposed" href="/icon.png"/>
- Startup screen:
  - <link rel="apple-touch-startup-image" href="/startup.png">

Install PhoneGap

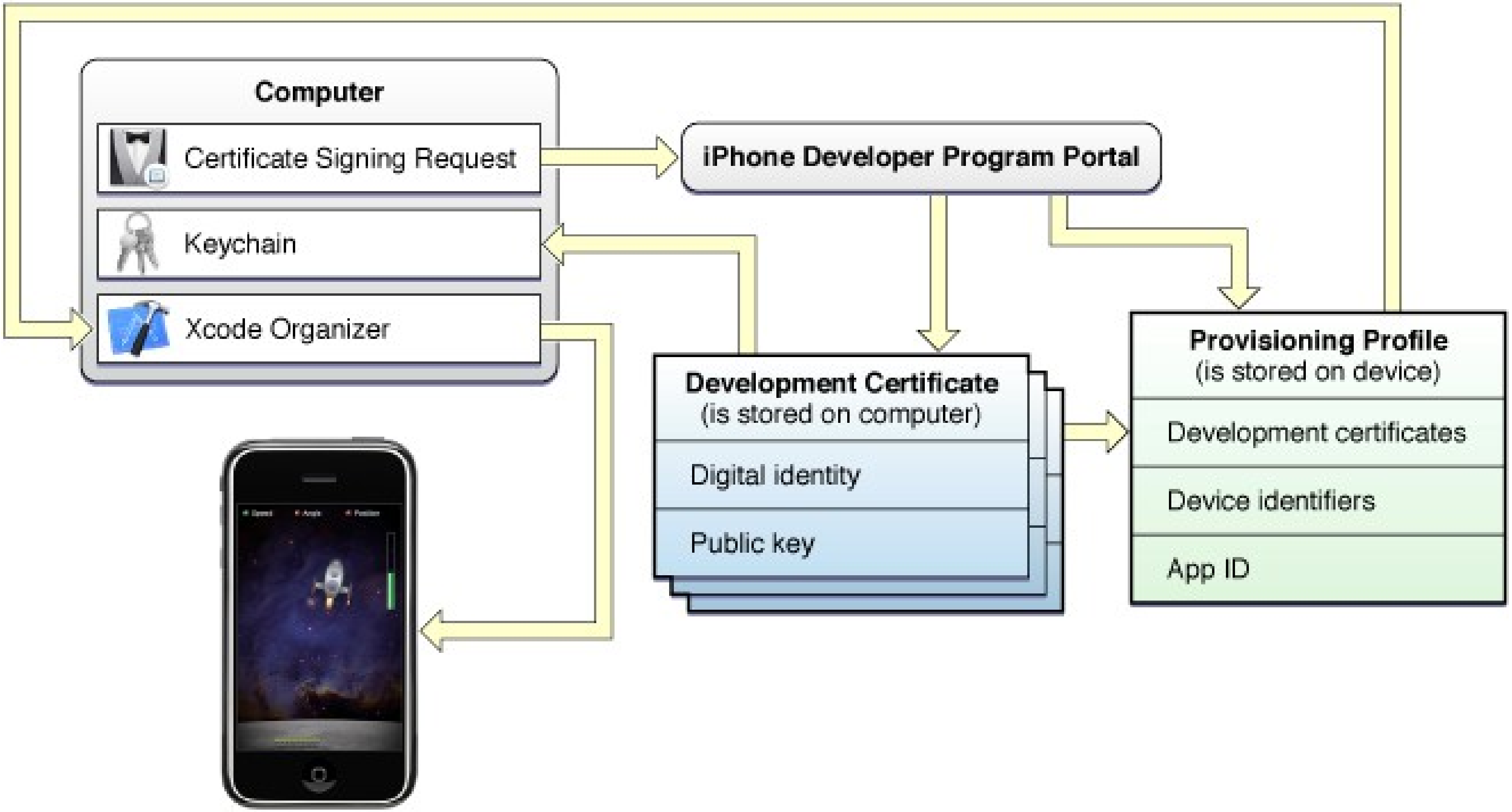
- iPhone:
  - Install Xcode plus iOS SDK (3 GB download)
  - Use installer PhoneGap provides
  - Select PhoneGap when making a new project in Xcode.

- Android
  - Install android SDK and add to $PATH
  - Run “android” to grab latest version
  - Setup Eclipse with Android plugin
  - Add PhoneGap.jar to project

Developing for iPhone

- PhoneGap builds in Xcode with iOS SDK
  - Intel-based Mac
  - Apple developer account (free)
    - Allows the use of iOS simulator
- Developer Program
  - $99 per year
  - Allows install onto device and release to store

Deploying to iOS device



Demo using Xcode

- Add HTML, CSS and JavaScript to www path
- Build and run
- App code comes after “deviceready” event
- Debugging:
  - console.log (JS)
  - NSLog (ObjC)
- Note:
  - *-Info.plist – settings file

Basic PhoneGap API

- Accelerometer
- Camera
- Contacts
- Device
- Events
- Geolocation
- Network
- Notification
- Also phone specific ones:
    - Android menu / back buttons
    - Read through phonegap.js

Beta testing

- Do a release build

- Create a release provisioning profile
    - Done through apples developer site
    - Uses phone UUID
    - Up to 100 phones

- End user drag profile and .app folder into iTunes
    - Sync with phone as normal

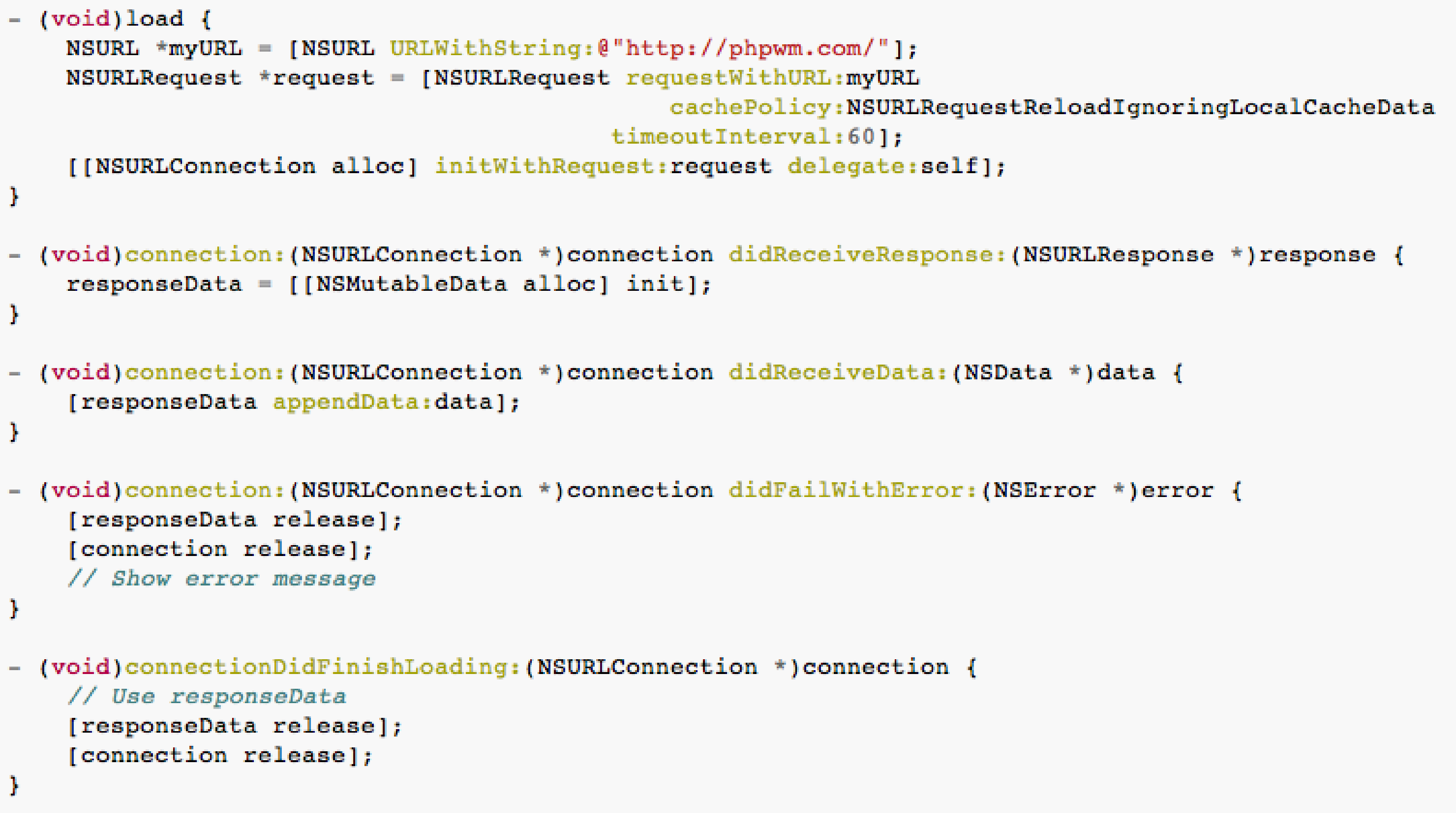
Submit to Apple

- Make sure the app does something
- Make sure app doesn't crash
- Check before using network
- Be careful how you use GPS
- Check copyright
- Apples testers seem US based

Objective C

- Strict superset of C
- Message passing to object instances
  - [obj method: argument];
- Garbage collection
  - Desktop only
    - [obj release];

Event Based



```
- (void)load {
    NSURL *myURL = [NSURL URLWithString:@"http://phpwm.com/"];
    NSURLRequest *request = [NSURLRequest requestWithURL:myURL
                                             cachePolicy:NSURLRequestReloadIgnoringLocalCacheData
                                         timeoutInterval:60];
    [[NSURLConnection alloc] initWithRequest:request delegate:self];
}

- (void)connection:(NSURLConnection *)connection didReceiveResponse:(NSURLResponse *)response {
    responseData = [[NSMutableData alloc] init];
}

- (void)connection:(NSURLConnection *)connection didReceiveData:(NSData *)data {
    [responseData appendData:data];
}

- (void)connection:(NSURLConnection *)connection didFailWithError:(NSError *)error {
    [responseData release];
    [connection release];
    // Show error message
}

- (void)connectionDidFinishLoading:(NSURLConnection *)connection {
    // Use responseData
    [responseData release];
    [connection release];
}
```

How does PhoneGap work?

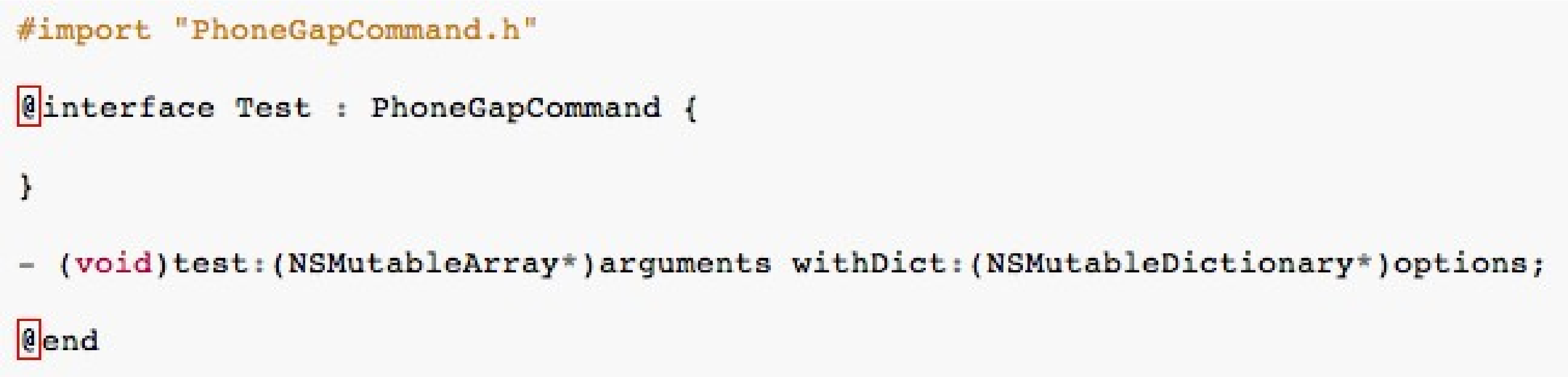
- webView:shouldStartLoadWithRequest:navigationType:
  - Messaged from instance of UIWebView

- PhoneGap looks for patterns like:
  - gap://<Class>.<command>/[<arguments>][?<dictionary>]

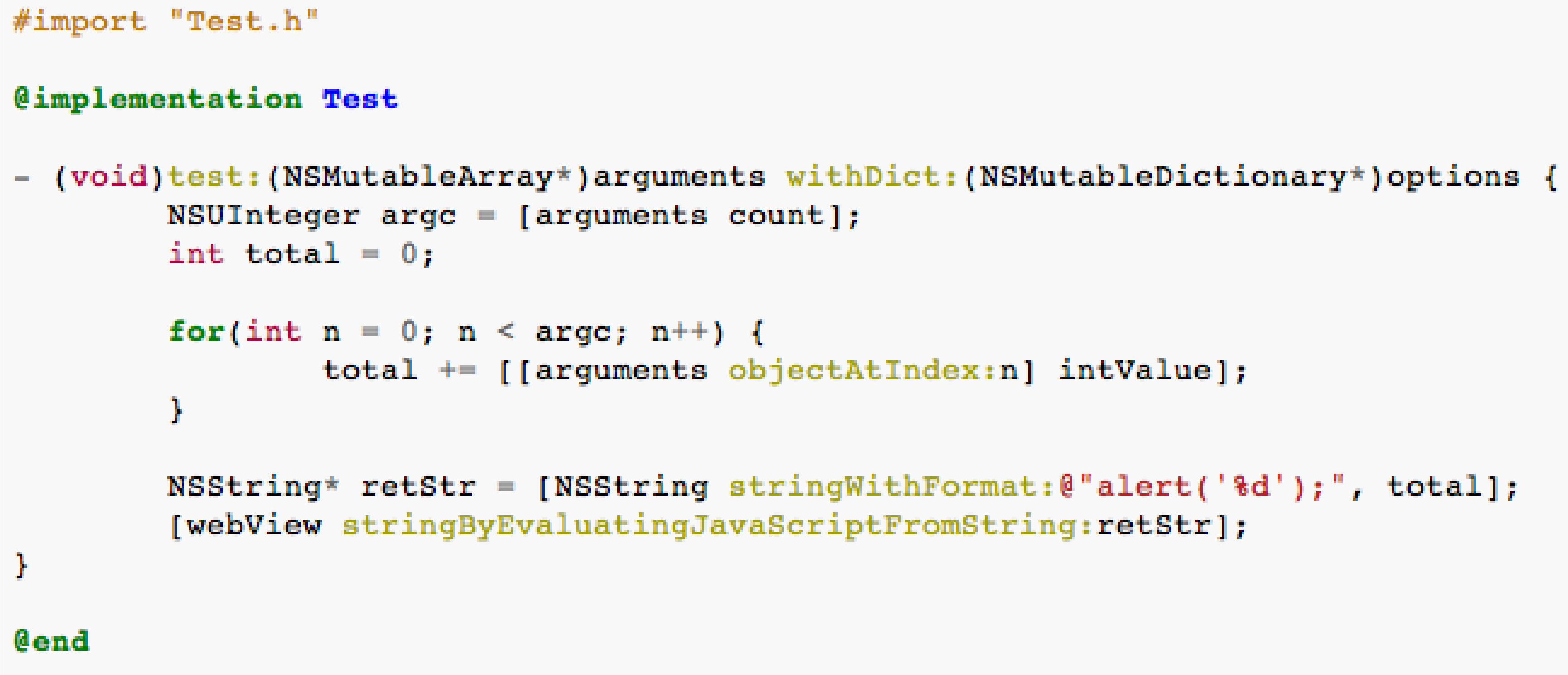
- Then creates instance of class and passes message

Writing iPhone plugins

- Create new header file (“Test.h”)
  - Extend PhoneGapCommand Class
- Create new ObjC class (“Test.m”)
  - Import new header file
- Call ObjC from JS
  - PhoneGap.exec("Test.test");
- Update JS from ObjC
  - [webView stringByEvaluatingJavaScriptFromString:@“alert('test')”]



```
#import "PhoneGapCommand.h"

@interface Test : PhoneGapCommand {

}

- (void)test:(NSMutableArray*)arguments withDict:(NSMutableDictionary*)options;

@end
```

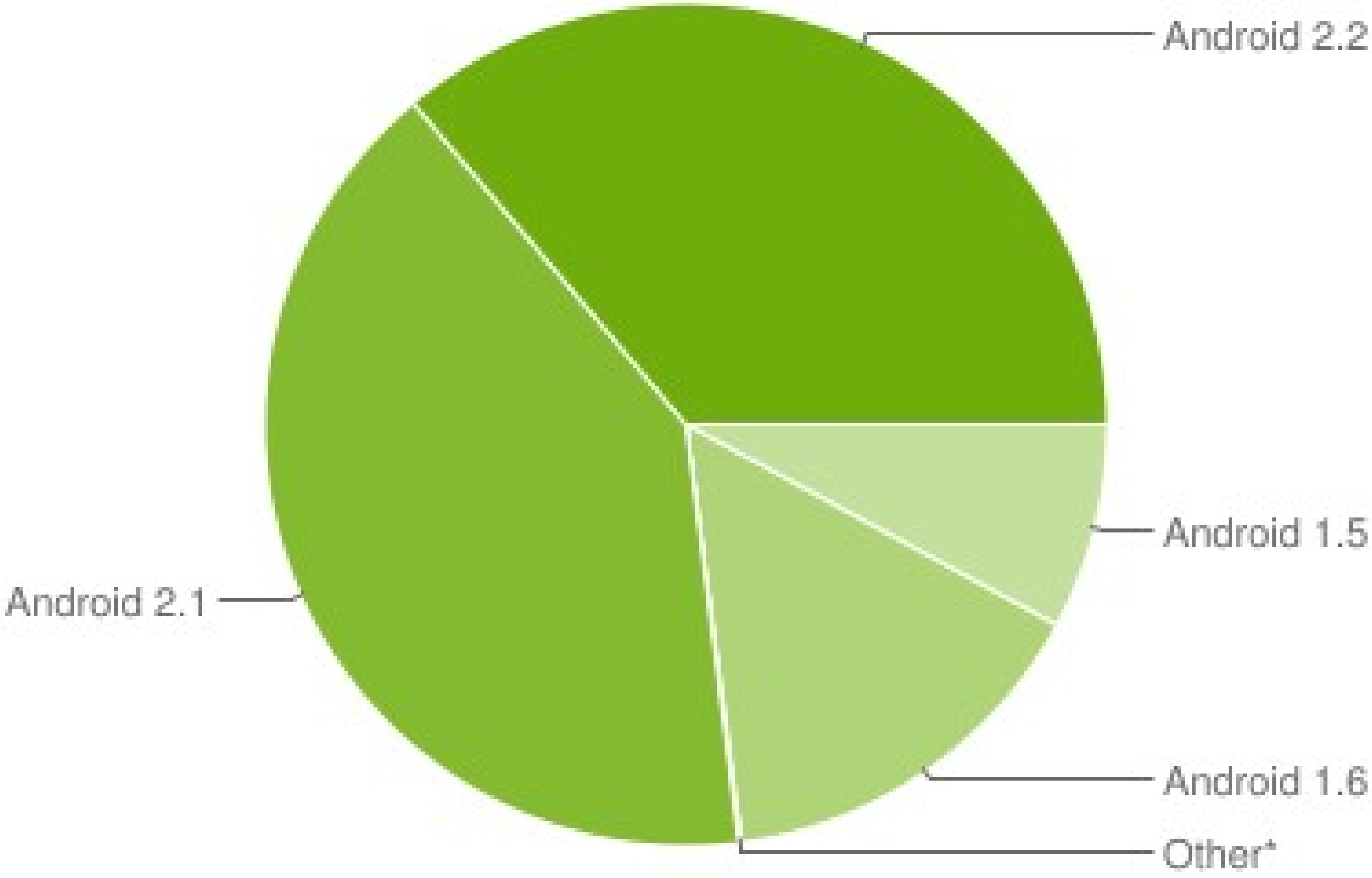


```
#import "Test.h"

@implementation Test

- (void)test:(NSMutableArray*)arguments withDict:(NSMutableDictionary*)options {
        NSUInteger argc = [arguments count];
        int total = 0;

        for(int n = 0; n < argc; n++) {
                total += [[arguments objectAtIndex:n] intValue];
        }

        NSString* retStr = [NSString stringWithFormat:@"alert('%d');", total];
        [webView stringByEvaluatingJavaScriptFromString:retStr];
}

@end
```

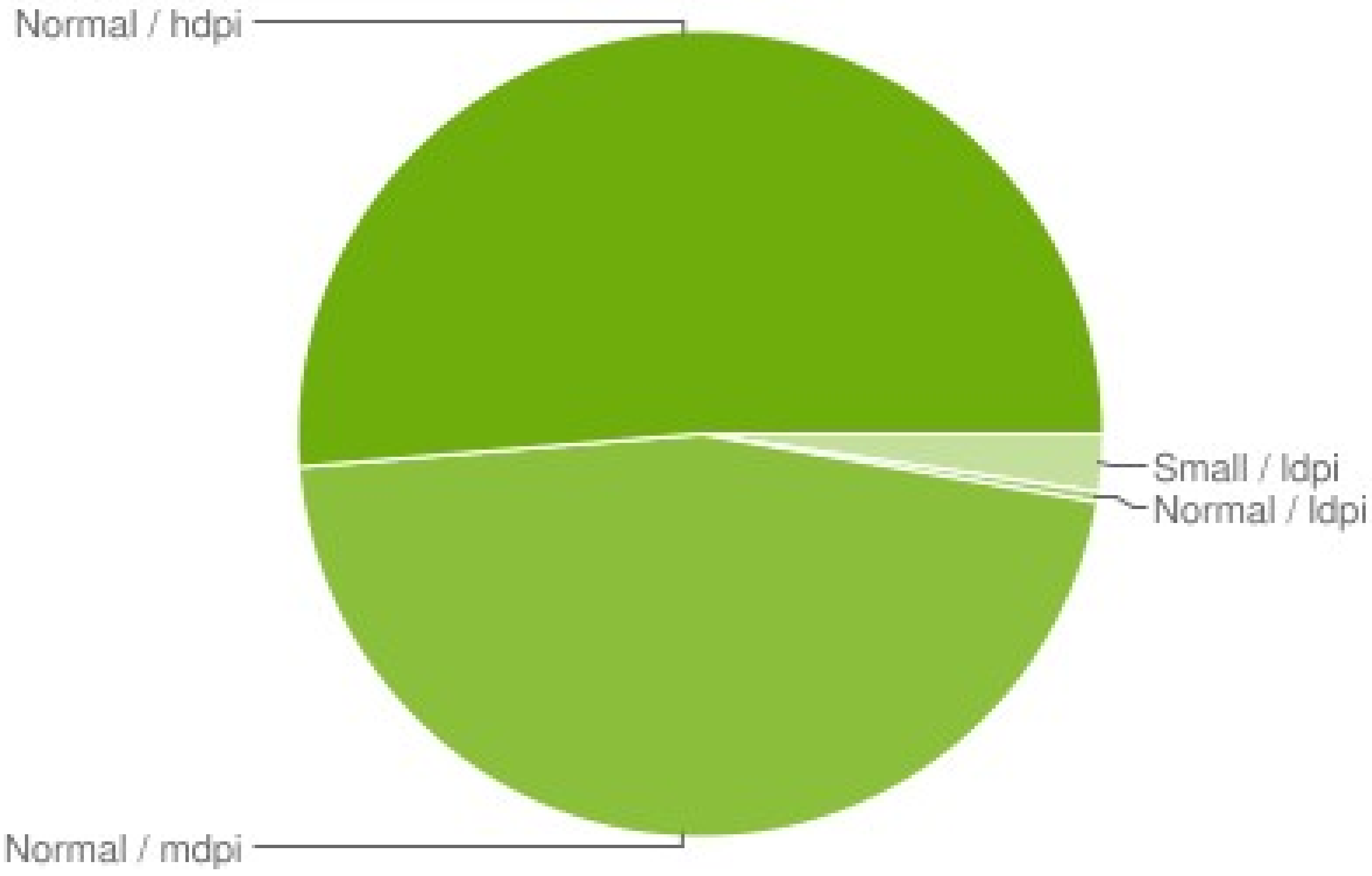
Developing for Android

- Works on linux / mac / windows
- No yearly fee (one time $25)
- No app review process
  - Instant updates vs a week
- Not all phones have Android Market
- Many 3rd party stores
- Fractured?

75% on 2.x



Screen Sizes



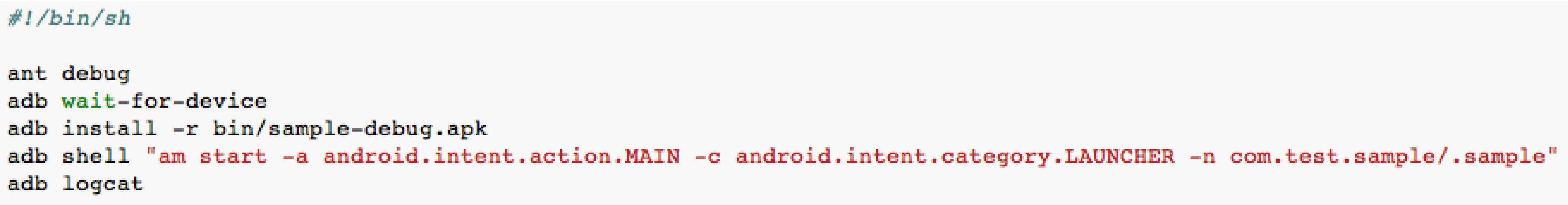
Demo using SDK

- Phone:
  - Enable usb development
  - Allow unknown sources

- Note: AndroidManifest.xml
  - Like *-Info.plist on iPhone
  - Permissions

- Debug:
  - console.log (JS)
  - Log.d (Java)

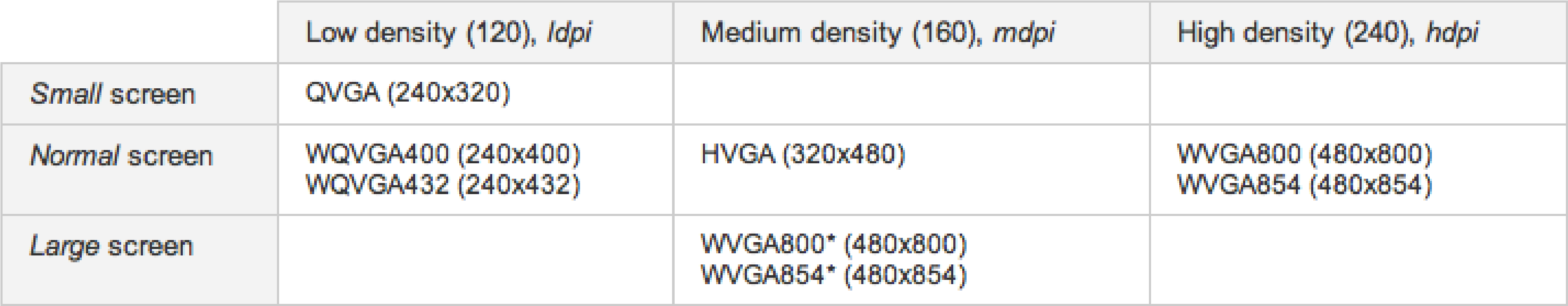
Build Process



```
#!/bin/sh

ant debug
adb wait-for-device
adb install -r bin/sample-debug.apk
adb shell "am start -a android.intent.action.MAIN -c android.intent.category.LAUNCHER -n com.test.sample/.sample"
adb logcat
```

- Also available through UI with Android Eclipse plug-in

Screen Sizes



	Low density (120), *ldpi*	Medium density (160), *mdpi*	High density (240), *hdpi*
Small screen	QVGA (240x320)		
Normal screen	WQVGA400 (240x400) WQVGA432 (240x432)	HVGA (320x480)	WVGA800 (480x800) WVGA854 (480x854)
Large screen		WVGA800* (480x800) WVGA854* (480x854)	

- AndroidManifest.xml turn on similar ipad mode
- Viewport can in force 3GS type
- window.innerHeight / element.offsetHeight

Beta testing / Release

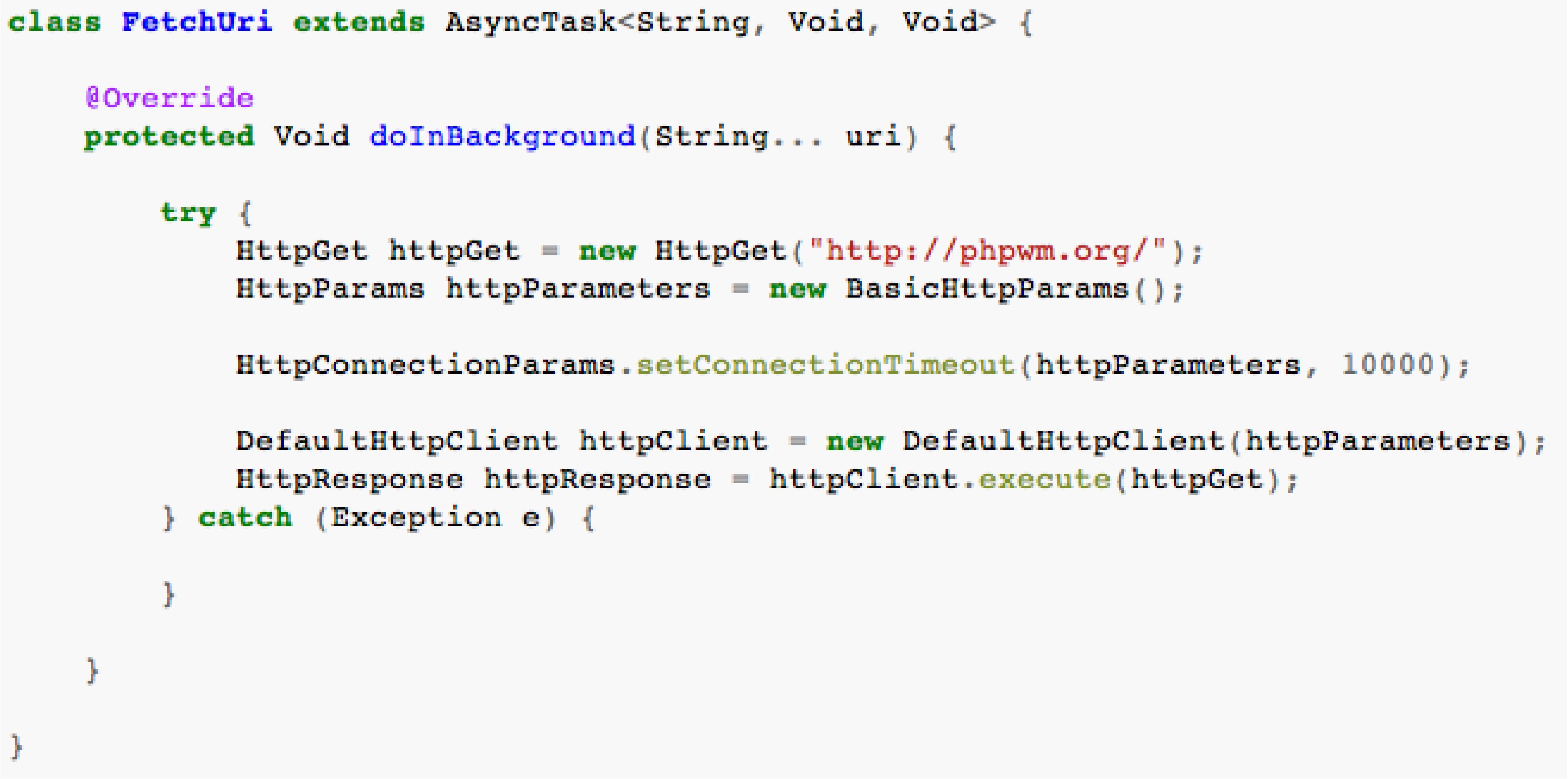
- Allow unknown sources in applications menu

- Transfer .apk file to phone:
  - Normally download in phones browser

- For release self sign .apk file

Java

- Garbage collection
- Easier to cause an app lock up
    - Google provide AsyncTask



```
class FetchUri extends AsyncTask<String, Void, Void> {

    @Override
    protected Void doInBackground(String... uri) {

        try {
            HttpGet httpGet = new HttpGet("http://phpwm.org/");
            HttpParams httpParameters = new BasicHttpParams();

            HttpConnectionParams.setConnectionTimeout(httpParameters, 10000);

            DefaultHttpClient httpClient = new DefaultHttpClient(httpParameters);
            HttpResponse httpResponse = httpClient.execute(httpGet);
        } catch (Exception e) {

        }

    }

}
```

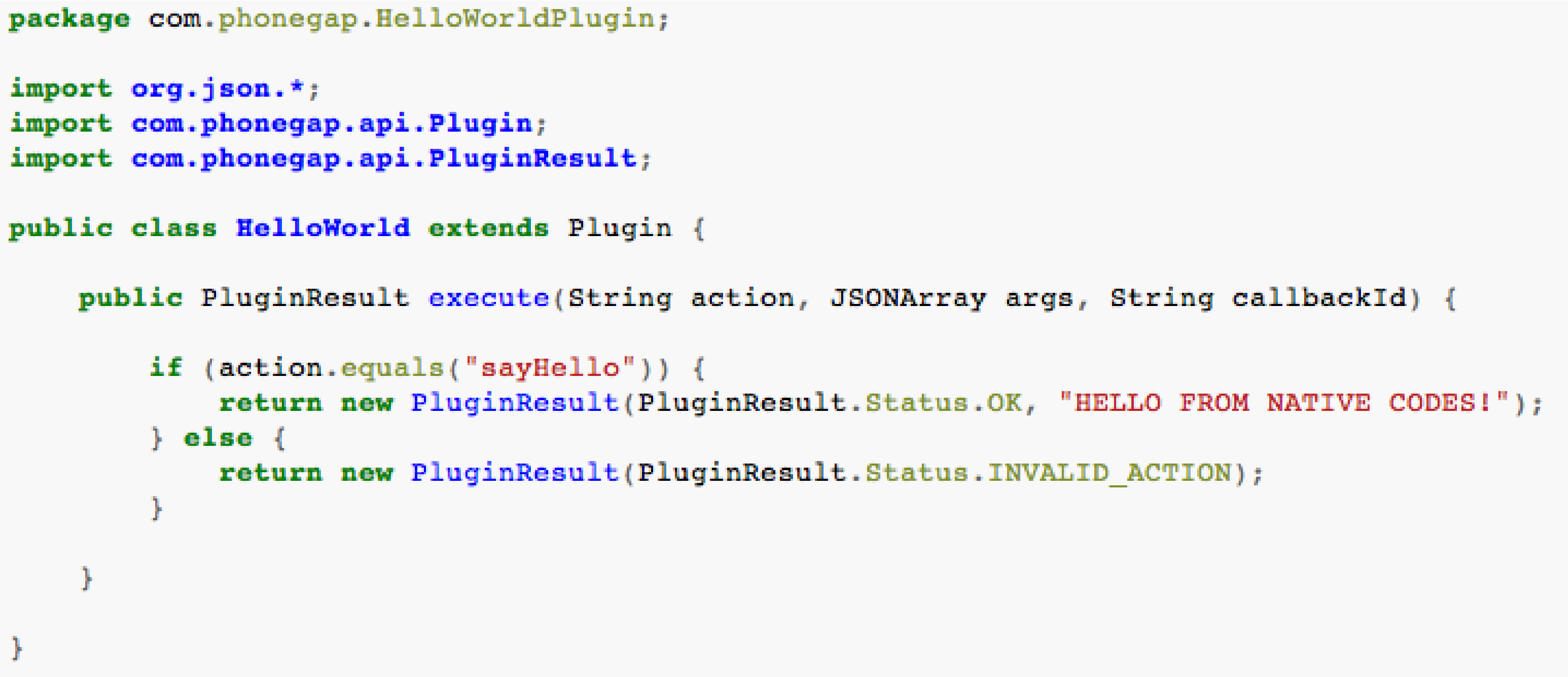
Writing Android plugins

- WebView features:
  - addJavascriptInterface(JavaObject, “NameInJs”)
  - Allows Java method to be called directly by JS

- PhoneGap 0.9.2 features JavaScript controlled plugin manager:
  - PluginManager.addService(

    ```
        "HelloWorld",
        "com.phonegap.HelloWorldPlugin.HelloWorld"
    );
    ```
  - PhoneGap.execAsync(win, fail, "HelloWorld", "sayHello", []);

Plugin Code



```
package com.phonegap.HelloWorldPlugin;

import org.json.*;
import com.phonegap.api.Plugin;
import com.phonegap.api.PluginResult;

public class HelloWorld extends Plugin {

    public PluginResult execute(String action, JSONArray args, String callbackId) {

        if (action.equals("sayHello")) {
            return new PluginResult(PluginResult.Status.OK, "HELLO FROM NATIVE CODES!");
        } else {
            return new PluginResult(PluginResult.Status.INVALID_ACTION);
        }

    }

}
```

Applications using PhoneGap

- Safari Books
- Ars Technica
- Wine.com
- CakeFest 2010
- TransLink
- Food Hygiene