

jQueryMobile & PhoneGap

跨平台手機開發實務

恆逸資深講師 戴玉佩

patty_tai@uuu.com.tw

認識jQueryMobile+PhoneGap

- jQueryMobile (第1章~第7章)
 - HTML5+CSS+JavaScript
 - 提供適合手持裝置的UI Layout
- PhoneGap (第8章~第21章)
 - 提供統一的JavaScript來操控不同Mobile平台中的硬體設備
 - 可發佈在多個平台
 - 只要會JavaScript即可
 - 完全免費

認識jQueryMobile+PhoneGap

- 螢幕畫面尺寸、長寬比、解析度大不同
 - jQueryMobile藉由HTML、CSS自動調整畫面比例來配合各種尺寸的Device
- 手機硬體無法由HTML操控
 - Phone Gap提供JavaScript程式庫

3

為何jQueryMobile+PhoneGap

- 門檻低
- 成本低
- 快速開發
- 跨平台

4

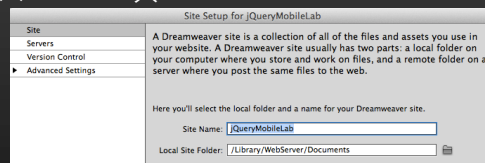
開始jQuery Mobile

- jQuery: logic, interaction
- jQuery Mobile: layout, 創造符合手機的介面
- 下載課程資源檔：
 - <http://goo.gl/z3imAb>
 - 解壓縮後，將jQueryMobile目錄Copy到WebServer上
 - Mac環境中可複製到/Library/WebServer/Documents中
- 啟動Web Server
 - 若要使用Mac環境中使用內建的Apache Server，開啟「終端機」輸入指令
`sudo apachectl restart`
- 使用開發工具撰寫jQueryMobile的網頁: DreamWeaver, Eclipse

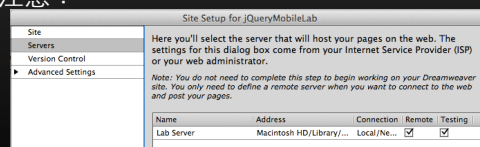
5

設定Server (一)

- 開啟DreamWeaver，從[Site]->[New Site]：
 - 在「Site」輸入 Site Name與Folder



- 注意：



6

建立HelloWorld.html(一)

1. 連結jQueryMobile官網，點選Download
2. 在CH00_Foundation新增HelloWorld.html
 - 在HelloWorld.html中加上引用CSS與JavaScript

```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=Big5">
    <title>Hello World</title>
    <link rel="stylesheet"
      href="http://code.jquery.com/mobile/1.4.4/jquery.mobile-1.4.4.min.css" />
    <script src="http://code.jquery.com/jquery-1.11.1.min.js"></script>
    <script src="http://code.jquery.com/mobile/1.4.4/jquery.mobile-1.4.4.min.js">
      </script>
    </head>
  ...
```

7

建立HelloWorld.html(二)

- 加上page

```
<body>
  <div data-role="page">
    <div data-role="header">
      <h1>HEADER</h1>
    </div>
    <div data-role="content">
      <h1>CONTENT</h1>
      <a href="#" data-role="button">Test BUTTON</a>
    </div>
    <div data-role="footer">
      <h1>FOOTER</h1>
    </div>
  </div>
</body>
```

8

建立HelloWorld.html(三)

- 注意要加ViewPort

```
<head>
  <meta http-equiv="Content-Type" content="text/html;
    charset=Big5">
  <meta name="viewport"
    content="width=device-width,initial-scale=1">
  <title></title>
  <link rel="stylesheet" href="http://code.jquery.com/mobile/
    1.4.4/jquery.mobile-1.4.4.min.css" />
  <script src="http://code.jquery.com/jquery-1.11.1.min.js"></
    script>
  <script src="http://code.jquery.com/mobile/1.4.4/
    jquery.mobile-1.4.4.min.js"></script>
</head>
```

9

jQuery Mobile基本頁面與對話框

Chapter 01

加上Style

- Sample1-2.html

```
<!DOCTYPE html>
<html>
  <head>
    ...
    <style>
      div [data-role="content"] > h1 {
        text-shadow: 1px 3px 2px #999;
      }

      div [data-role="content"] > p {
        font:1em/1.5 Georgia, "Time New Roman", Times, serif;
      }
    </style>
  </head>
```

11

多頁結構

- Sample1- 3.html : 注意這裡有兩個page

```
<div data-role="page" id="home">
  ...
</div>
<div data-role="page" id="detail">
  <div data-role="header">
    <h1>Detail</h1>
  </div>
  <div data-role="content">
    <h1>Easy to use: Try it now!</h1>
    <p>jQuery Mobile's emphasis on semantic markup and progressive
      enhancement makes it super easy to use. ...</p>
    <a href="#">Demos & documentation</a>
    <a href="#">Resources: Books, plugins, tutorials</a><br>
    <a href="#home" data-role="button">Goback Home</a>
  </div>
</div>
```

12

Title on Ajax Navigation

- Sample1-4.html(加上 data-title)

```
<div data-role="page" id="home" data-title="Home Page">
...
</div>

<div data-role="page" id="detail" data-title="Detail Page">
...
</div>
```

13

帶出下個Page

- Sample1- 5.html中以3種方式來帶出Sample1-5-1.html

- 預設方式：AJAX
- 對話窗（可設定對話窗出現的方式）
- 非 Ajax方式

```
<a href="Sample1-5-1.html" data-role="button">Go to Sample 1-5-1 with Ajax</a>
<a href="Sample1-5-1.html" data-role="button" data-rel="dialog"
  data-transition="pop">Go to Sample 1-5-1 with Dialog</a>
<a href="Sample1-5-1.html" data-role="button" data-ajax="false">
  Go to Sample 1-5-1 without Ajax</a>
```

- 若為外部網站的網頁，應為：

```
<a href="http://maps.google.com" data-role="button" data-rel="external">Go to
Google Map</a>
```

14

回到上一頁

- Sample1- 5-1.html中以2種方式來回到Sample1-5.html

```
<a href = "Sample1-5.html" data-role="button">Go Back</a>
<a href = "#" data-role="button" data-rel="back">Go Back</a>
```

- 若data-transition用slide，要注意back的方向(

- Sample1-6.html

```
<a href="Sample1-6-1.html" data-role="button" data-icon="arrow-r"
  data-transition="slide">Go to Sample1-6-1.html with AJAX</a>
```

- Sample1-6-1.html

```
<a href="#" data-role="button" data-icon="back" data-rel="back"
  data-transition="slide" data-direction="reverse">Go back</a>
```

15

發mail、打電話、傳簡訊

- Sample1- 7.html加入下列button

```
<a href="mailto:admin@uuu.com.tw" data-role='button'>
  Mail to Admin</a>

<a href="mailto:admin@uuu.com.tw?
cc=patty_tai@uuu.com.tw&subject=Test%20Mail&body=This%20is
%20a%20test%20from%20HTML5" data-role='button'>Mail to Patty
with Content</a>

<a href="tel:0225149191" data-role='button'>Make Phone Call to
UCOM</a>

<a href="sms:0987654321?body=Test%20Data" data-role='button'>
  Send MSG to Someone</a>
```

16

Page Transition

- 10種Transition
 - fade(default)
 - flip
 - pop
 - turn
 - flow
 - slide
 - slideup
 - slidedown
 - slidefade
 - none
- Sample1- 8.html加入使用不同Transition方式的button來帶出 Sample1- 8-1.html

17

Page Transition

- Sample1- 8.html加入下列button

```
<a href="Sample1-8-1.html" data-role="button" >
  Go to Sample1-8-1.html with default</a>
<a href="Sample1-8-1.html" data-role="button" data-transition="fade">
  Go to Sample1-8-1.html with fade</a>
<a href="Sample1-8-1.html" data-role="button" data-transition="flip">
  Go to Sample1-8-1.html with flip</a>
<a href="Sample1-8-1.html" data-role="button" data-transition="pop">
  Go to Sample1-8-1.html with pop</a>
<a href="Sample1-8-1.html" data-role="button" data-transition="slide">
  Go to Sample1-8-1.html with slide</a>
<a href="Sample1-8-1.html" data-role="button" data-transition="slidedown">
  Go to Sample1-8-1.html with slidedown</a>
<a href="Sample1-8-1.html" data-role="button" data-transition="slidefade">
  Go to Sample1-8-1.html with slidefade</a>
<a href="Sample1-8-1.html" data-role="button" data-transition="slideup">
  Go to Sample1-8-1.html with slideup</a>
<a href="Sample1-8-1.html" data-role="button" data-transition="turn">
  Go to Sample1-8-1.html with turn</a>
<a href="Sample1-8-1.html" data-role="button" data-transition="flow">
  Go to Sample1-8-1.html with flow</a>
```

18

緩存頁面

- 在Smple1-9-1.html加上緩存頁面的設定

```
<div data-role="page" data-dom-cache="true">
```

- 小心效能問題

19

jQuery Mobile的ToolBar

Chapter 02

Toolbar

- jQuery Mobile有兩種標準Toolbar
 - Header
 - Footer

21

可以調整Toolbar的位置

- Sample2-1.html

```
<div data-role="header" data-position="fixed">
  <h1>iPhone</h1>
</div>
<div data-role="content">
  <h1>iTunes in the Cloud</h1>
  <h2>Your music, movies, and TV shows. Wherever you want
    them.</h2>
  <p>... </p>
  
  <a href="#">Learn more about the iTunes app</a>
</div>
<div data-role="footer" data-position="fixed">
  <h1>Fixed mode footer</h1>
</div>
```

22

full-screen效果

- Sample2-2.html

```
<div data-role="header" data-position="fixed"
    data-fullscreen="true">
  <h1>iPhone</h1>
</div>
<div data-role="content">
  <h1>iTunes in the Cloud</h1>
  <h2>Your music, movies, and TV shows. Wherever you want
    them.</h2>
  <p>...</p>
  
  <a href="#">Learn more about the iTunes app</a>
</div>
<div data-role="footer" data-position="fixed"
    data-fullscreen="true">
  <h1>Fixed mode footer</h1>
</div>
```

23

Header結構- 導覽按鍵 (1)

- Sample2-3.html的header加入下列button

```
<div data-role="page">
  <div data-role="header">
    <a href="#help">HELP</a>
    <h1>Setting</h1>
    <a href="#" data-theme="b">DONE</a>
  </div>
  ...
</div>
<div id="help" data-role="page">
  <div data-role="header">
    <a data-rel="back">Back</a>
    <h1>HELP</h1>
  </div>
  ...
</div>
```

24

Header結構- 導覽按鍵 (2)

- Sample2-4.html的header加入下列button

```
<div data-role="page">
  <div data-role="header" data-theme='a'>
    <h1>Setting</h1>
    <a href="#help" data-icon="plus" data-transition="slide"
      class="ui-btn-left" >HELP</a>
    <a id="doneButton" href="#" data-icon="check"
      class="ui-btn-right">DONE</a>
  </div>
  ...
</div>
<div id="help" data-role="page" data-add-back-btn="true">
  <div data-role="header" data-theme="e" data-add-back-
    btn="true">
    <a data-rel="back">Back</a>
    <h1>HELP</h1>
  </div>
</div>
...
</div>
```

1.3.2

1.4.x

25

Footer結構- 工具列按鍵

- Sample2-5.html的footer加入下列button

```
<div data-role="footer" class="ui-bar">
  <a href="#" data-role="button" data-icon="home">Home</a>
  <a href="#" data-role="button" data-icon="plus">
    Add new Setting</a>
</div>
```

- 加上controlgroup建立按鈕集

```
<div data-role="footer" class="ui-bar">
  <div data-role="controlgroup" data-type="horizontal">
    <a href="#" data-role="button" data-icon="home">Home</a>
    <a href="#" data-role="button" data-icon="plus">
      Add new Setting</a>
  </div>
</div>
```

26

Footer結構- 加上navbar

- Sample2-6.html的footer加入下列navbar

```
<div data-role="footer" data-position="fixed">
  <div data-role="navbar">
    <ul>
      <li><a href="#" data-icon="alert"
        class="ui-btn-active">page 1</a></li>
      <li><a href="#" data-icon="alert">page 2</a></li>
      <li><a href="#" data-icon="alert">page 3</a></li>
      <li><a href="#" data-icon="alert">page 4</a></li>
    </ul>
  </div>
</div>
```

27

Footer結構- 加上navbar

- Sample2-7.html的footer加入下列navbar

```
<div data-role="footer" data-position="fixed">
  <div data-role="navbar" data-iconpos="left">
    <ul>
      <li><a href="#page1" data-icon="gear"
        data-transition="none">PAGE1</a></li>
      <li><a href="#page2" data-icon="grid"
        data-transition="none">PAGE2</a></li>
      <li><a href="#page3" data-icon="info"
        data-transition="none">PAGE3</a></li>
      <li><a href="#page4" data-icon="star"
        data-transition="none">PAGE4</a></li>
    </ul>
  </div>
</div>
```

28

jQueryMobile按鈕

Chapter03

各種表單按鈕

- Sample3-1.html

```
<div data-role="content">
  <a href="#" data-role="button">Default Button</a>
  <a href="#" data-role="button" data-mini="true">Default Button</a>
  <a href="#" class="ui-btn ui-mini">Mini Button</a>
  <a href="#" class="ui-btn ui-btn-inline">Inline 1</a>
  <a href="#" data-role="button" data-inline="true">Inline 2</a>
  <a href="#" data-role="button" data-inline="true">Inline 3</a>

  <button>Button</button>
  <button data-role="none">None</button>
  <input type="button" value='Input Button'>
  <input type="submit" value='Submit'>
  <input type="reset" value='Reset'>
</div>
```

各種按鈕圖示

- 50種圖示
- Sample3-2

```
<div data-role="content">
<a data-role="button" data-icon="action">action</a>
<a data-role="button" data-icon="arrow-d-l">arrow-d-l</a>
<a data-role="button" data-icon="arrow-d-r">arrow-d-r</a>
<a data-role="button" data-icon="arrow-d">arrow-d</a>
<a data-role="button" data-icon="arrow-l">arrow-l</a>
<a data-role="button" data-icon="arrow-r">arrow-r</a>
<a data-role="button" data-icon="arrow-u-l">arrow-u-l</a>
<a data-role="button" data-icon="arrow-u-r">arrow-u-r</a>
<a data-role="button" data-icon="arrow-u">arrow-u</a>
<a data-role="button" data-icon="audio">audio</a>

<a data-role="button" data-icon="calendar">calendar</a>
<a data-role="button" data-icon="camera">camera</a>
<a data-role="button" data-icon="carat-d">carat-d</a>
<a data-role="button" data-icon="carat-l">carat-l</a>
<a data-role="button" data-icon="carat-r">carat-r</a>
<a data-role="button" data-icon="carat-u">carat-u</a>
<a data-role="button" data-icon="check">check</a>
<a data-role="button" data-icon="clock">clock</a>
<a data-role="button" data-icon="cloud">cloud</a>
<a data-role="button" data-icon="grid">grid</a>

<a data-role="button" data-icon="mail">mail</a>
<a data-role="button" data-icon="eye">eye</a>
<a data-role="button" data-icon="gear">gear</a>
<a data-role="button" data-icon="heart">heart</a>
<a data-role="button" data-icon="home">home</a>
```

31

各種按鈕圖示(續)

- 續Sample3-2

```
<a data-role="button" data-icon="info">info</a>
<a data-role="button" data-icon="bullets">bullets</a>
<a data-role="button" data-icon="bars">bars</a>
<a data-role="button" data-icon="navigation">navigation</a>
<a data-role="button" data-icon="lock">lock</a>

<a data-role="button" data-icon="search">search</a>
<a data-role="button" data-icon="location">location</a>
<a data-role="button" data-icon="minus">minus</a>
<a data-role="button" data-icon="forbidden">forbidden</a>
<a data-role="button" data-icon="edit">edit</a>
<a data-role="button" data-icon="user">user</a>
<a data-role="button" data-icon="phone">phone</a>
<a data-role="button" data-icon="plus">plus</a>
<a data-role="button" data-icon="power">power</a>
<a data-role="button" data-icon="recycle">recycle</a>

<a data-role="button" data-icon="forward">forward</a>
<a data-role="button" data-icon="refresh">refresh</a>
<a data-role="button" data-icon="shop">shop</a>
<a data-role="button" data-icon="comment">comment</a>
<a data-role="button" data-icon="star">star</a>
<a data-role="button" data-icon="tag">tag</a>
<a data-role="button" data-icon="back">back</a>
<a data-role="button" data-icon="video">video</a>
<a data-role="button" data-icon="alert">alert</a>
<a data-role="button" data-icon="delete">delete</a>

</div>
```

32

文字、圖案位置

- Sample3-3.html

```
<a href="#" data-role="button" data-icon="alert">Button</a>
<a href="#" data-role="button" data-icon="alert"
  data-iconpos='left' >Button</a>
<a href="#" data-role="button" data-icon="alert"
  data-iconpos='right'>Button</a>
<a href="#" data-role="button" data-icon="alert"
  data-iconpos='top'>Button</a>
<a href="#" data-role="button" data-icon="alert"
  data-iconpos='bottom'>Button</a>
<a href="#" data-role="button" data-icon="alert"
  data-iconpos='notext'>Button</a>
<a href="#" data-role="button" data-icon="alert"
  data-iconpos='notext' data-inline="true">Button</a>
<a href="#" data-role="button" data-icon="alert"
  data-iconpos='left' data-inline="true">Button</a>
```

33

自訂圖示

- 白色36x36圖示
- Sample3-4.html

```
<style>
.ui-icon-fb:after{ background: url(images/fb.png);
  background-size: 20px 20px;}

.ui-icon-tw:after{ background: url(images/tw.png);
  background-size: contain; }

.ui-icon-rss:after{ background: url(images/rss.png); }
</style>
...
<div data-role="content">
  <a href="#" data-role="button" data-icon="fb">Facebook</a>
  <a href="#" data-role="button" data-icon="tw">Twitter</a>
  <a href="#" data-role="button" data-icon="rss">RSS</a>
</div>
```

34

按鍵集

- 直向排列：Sample3-5.html

```
<div data-role="content">
  <div data-role="controlgroup" data-type="vertical">
    <a href="#" data-role="button">Facebook</a>
    <a href="#" data-role="button">Twitter</a>
    <a href="#" data-role="button">RSS</a>
  </div>
</div>
```

- 水平排列：Sample3-6.html

```
<div data-role="content">
  <div data-role="controlgroup" data-type="horizontal">
    <a href="#" data-role="button">Facebook</a>
    <a href="#" data-role="button">Twitter</a>
    <a href="#" data-role="button">RSS</a>
  </div>
</div>
```

35

jQuery Mobile格式化排版

第4章

Grid樣式

- 網格效果
- 4種佈局：
 - 二列（使用ui-grid-a）
 - 三列（使用ui-grid-b）
 - 四列（使用ui-grid-c）
 - 五列（使用ui-grid-d）

Two columns



37

基本Grid

- Sample4-1.html

```
<h1>Two columns</h1>
<div class="ui-grid-a">
  <div class="ui-block-a">
    
  </div>
  <div class="ui-block-b">
    
  </div>
</div>
```

```
<h1>Three columns</h1>
<div class="ui-grid-b">
  <div class="ui-block-a">
    
  </div>
  <div class="ui-block-b">
    
  </div>
  <div class="ui-block-c">
    
  </div>
</div>
```

38

Collapse(收合折疊)

- Sample4-2.html

```
<div data-role="collapsible-set">
  <div data-role="collapsible" data-collapsed="false"
    data-theme="a" data-content-theme="b" >
    <h1>Photoshop</h1>
    <p>
      The Adobe® Photoshop® family of products is the ultimate playground for
      bringing out the best in your digital images, transforming them into
      anything you can imagine, and showcasing them in extraordinary ways.
    </p>
  </div>
  <div data-role="collapsible" data-theme="b"
    data-content-theme="a">
    <h1>Acrobat</h1>
    <p>
      Adobe® Acrobat® X software lets you communicate easier, faster, and
      better with PDF. Increase your productivity by editing and reusing
      content, streamlining reviews, and quickly creating fillable PDF forms.
    </p>
  </div>
</div>
```

39

jQuery Mobile表單元件

第5章

基本輸入項

```
<div data-role="fieldcontain">
  <label for="name">姓名 :</label>
  <input id="name" name="name">
</div>
<div data-role="fieldcontain">
  <label for="password">密碼 :</label>
  <input type="password" id="password" name="password">
</div>
<div data-role="fieldcontain" class="ui-disabled">
  <label for="search">查詢產品 :</label>
  <input type="search" id="search" name="search">
</div>
<div data-role="fieldcontain">
  <label for="gender">性別 :</label>
  <select data-role="slider" id="gender" name="gender">
    <option value="0">女</option><option value="1">男</option>
  </select>
</div>
```

41

基本輸入項

```
<div data-role="fieldcontain">
  <label for="numberp">Number + [0-9]* pattern:</label>
  <input type="number" id="numberp" name="number"
    pattern="[0-9]*" value="0">
</div>
<div data-role="fieldcontain">
  <label for="birthday">生日 :</label>
  <input type="date" id="fieldcontain" name="birthday">
</div>
<div data-role="fieldcontain">
  <label for="phone">電話 :</label>
  <input type="tel" id="phone" name="phone">
</div>
```

42

Range輸入項

```
<div data-role="fieldcontain">
  <label for="age">年齡: </label>
  <input type="range" id="age" name="age" min="1" max="99"
    value="18">
</div>

<div data-role="rangeslider">
  <label for="range-1a">Rangeslider:</label>
  <input type="range" name="range-1a" id="range-1a"
    min="0" max="100" value="40">
  <label for="range-1b">Rangeslider:</label>
  <input type="range" name="range-1b" id="range-1b"
    min="0" max="100" value="80">
</div>
```

43

選擇輸入項：Checkbox

```
<div data-role="fieldcontain">
  <fieldset data-role="controlgroup" data-type="horizontal">
    <legend>選擇你的嗜好: </legend>
    <input type="checkbox" id="hobby1" name="hobby">
    <label for="hobby1">閱讀</label>
    <input type="checkbox" id="hobby2" name="hobby">
    <label for="hobby2">運動</label>
    <input type="checkbox" id="hobby3" name="hobby">
    <label for="hobby3">電影</label>
  </fieldset>
</div>
```

44

選擇輸入項：Radio

```
<div data-role="fieldcontain">
  <fieldset data-role="controlgroup" data-type="horizontal">
    <legend>選擇你的衣服尺寸：</legend>
    <input type="radio" id="size1" name="size">
    <label for="size1">XS</label>
    <input type="radio" id="size2" name="size">
    <label for="size2">S</label>
    <input type="radio" id="size3" name="size">
    <label for="size3">M</label>
    <input type="radio" id="size4" name="size">
    <label for="size4">L</label>
    <input type="radio" id="size5" name="size">
    <label for="size5">XL</label>
  </fieldset>
</div>
```

45

選擇輸入項：Select

```
<div data-role="fieldcontain">
  <label for="shipping">運送方式：</label>
  <select id="shipping" name="shipping"
    data-native-menu="false">
    <option value="standard">標準(7日)</option>
    <option value="rush">急件(3日)</option>
    <option value="express">快捷(1日)</option>
  </select>
</div>
```

46

jQuery Mobile列表

第6章

基本列表

- Sample6-1.html

```
<div data-role="content">
  <ul data-role="listview">
    <li><a href="#">item 1</a></li>
    <li><a href="#">item 2</a></li>
    <li><a href="#">item 3</a></li>
    <li><a href="#">item 4</a></li>
    <li><a href="#">item 5</a></li>
    <li><a href="#">item 6</a></li>
    <li><a href="#">item 7</a></li>
    <li><a href="#">item 8</a></li>
    <li><a href="#">item 9</a></li>
    <li><a href="#">item 10</a></li>
  </ul>
</div>
```


數字排序清單

- Sample6-2.html

```
<div data-role="content">
  <ol data-role="listview">
    <li><a href="#">item 1</a></li>
    <li><a href="#">item 2</a></li>
    <li><a href="#">item 3</a></li>
    <li><a href="#">item 4</a></li>
    <li><a href="#">item 5</a></li>
    <li><a href="#">item 6</a></li>
    <li><a href="#">item 7</a></li>
    <li><a href="#">item 8</a></li>
    <li><a href="#">item 9</a></li>
    <li><a href="#">item 10</a></li>
  </ol>
</div>
```

49

唯讀列表

- Sample6-3.html

```
<div data-role="content">
  <ol data-role="listview">
    <li>item 1</li>
    <li>item 2</li>
    <li>item 3</li>
    <li>item 4</li>
    <li>item 5</li>
    <li>item 6</li>
    <li>item 7</li>
    <li>item 8</li>
    <li>item 9</li>
    <li>item 10</li>
  </ol>
</div>
```

50

搜尋過濾

- Sample6-4.html

```
<div data-role="content">
  <ul data-role="listview" data-filter="true" data-filter-placeholder="Keyword">
    <li data-filtertext="client"><a href="#">jQuery</a></li>
    <li data-filtertext="client"><a href="#">HTML5</a></li>
    <li data-filtertext="client server"><a href="#">JAVA</a></li>
    <li data-filtertext="client server"><a href="#">JavaScript</a></li>
    <li data-filtertext="server"><a href="#">PHP</a></li>
    <li data-filtertext="client server"><a href="#">SQL</a></li>
    <li data-filtertext="server"><a href="#">ColdFusion</a></li>
    <li data-filtertext="server"><a href="#">Ruby</a></li>
    <li data-filtertext="client server"><a href="#">Objective-C</a></li>
    <li data-filtertext="client"><a href="#">CSS</a></li>
  </ul>
</div>
```

51

搜尋過濾(續)

- Sample6-5.html

```
<div data-role="content">
  <ul data-role="listview" data-filter="true" data-filter-placeholder="Keyword">
    <li data-role="list-divider">Client</li>
    <li><a href="#">jQuery</a></li>
    <li><a href="#">HTML5</a></li>
    <li><a href="#">JAVA</a></li>
    <li><a href="#">JavaScript</a></li>
    <li><a href="#">CSS</a></li>
    <li data-role="list-divider">Server</li>
    <li><a href="#">JAVA</a></li>
    <li><a href="#">PHP</a></li>
    <li><a href="#">SQL</a></li>
    <li><a href="#">ColdFusion</a></li>
    <li><a href="#">Ruby</a></li>
    <li><a href="#">Objective-C</a></li>
  </ul>
</div>
```

52

Nested lists

- 在jQueryMobile 1.4.x 停用
- Sample6-6.html

```
<div data-role="content">
  <ul data-role="listview">
    <li>jQuery
      <ul>
        <li>Download</li>
        <li>Documentation</li>
        <li>Tutorials</li>
        <li>Bug Tracker</li>
        <li>Discussion</li>
      </ul>
    </li>
  </ul>
</div>
```

53

Text Formatting & counts

- Sample6-7.html

```
<div data-role="content">
  <ul data-role="listview">
    <li><a href="#">Inbox<span class="ui-li-count">5</span></a></li>
    <li><a href="#">Outbox<span class="ui-li-count">99</span></a></li>
    <li><a href="#">Drafts<span class="ui-li-count">0</span></a></li>
    <li><a href="#">Sent<span class="ui-li-count">123</span></a></li>
    <li><a href="#">Trash<span class="ui-li-count">35</span></a></li>
  </ul>
</div>
```

54

清單中的圖示

- Sample6-8.html

```
<div data-role="content">
  <ul data-role="listview">
    <li>
      <a href="#">
        
        <h1>Illustrator</h1>
        <p>Adobe design tool</p>
      </a>
    </li>
  </ul>
</div>
```

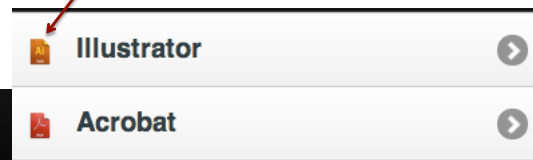


55

清單中的縮圖

- Sample6-9.html

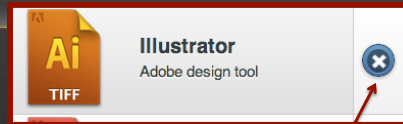
```
<div data-role="content">
  <ul data-role="listview">
    <li>
      <a href="#">
        
        Acrobat
      </a>
    </li>
  </ul>
</div>
```



56

Split button lists

- Sample6-10.html



```
<div data-role="content">
  <ul data-role="listview" data-split-icon="delete">
    <li>
      <a href="#">
        
          Acrobat
        </a>
      <a href="#">Delete</a>
    </li>
  </ul>
</div>
```

57

jQuery API與事件

第七章

撰寫jQueryMobile的事件

- 可以override jQueryMobile的事件
 - 注意要撰寫jQueryMobile的mobileinit事件時，<script>的位置:

```
<script src="jquery.js"></script>
&ltscript src="自己的Javascript.js"></script>
&ltscript src="jquery.mobile.js"></script>
```

59

jQueryMobile自動初始化(續)

- mobileinit事件
- Sample7-1.html

```
<script src="http://code.jquery.com/jquery-1.11.1.min.js"></script>
&ltscript>
    $(document).on("mobileinit", mobileinitHandler);
    function mobileinitHandler () {
        //alert("Hello...")
        $.mobile.ajaxEnabled=false;
        $.mobile.defaultPageTransition = "slide";
        $.mobile.defaultDialogTransition = "flip";
    }
</script>
&ltscript src="http://code.jquery.com/mobile/1.4.4/jquery.mobile-1.4.4.min.js"></script>
```

60

loadingMessage

- Sample7-2.html

```
<script>
$(document).on("mobileinit", mobileinitHandler);

function mobileinitHandler (){
$.mobile.pageLoadErrorMessage="系統忙碌中, 請稍候再試!";
$.mobile.pageLoadErrorMessageTheme = 'a';
}

function showHandler(){
//$.mobile.showPageLoadingMsg(); //1.4已不支援此method

$.mobile.loading("show",
{text: "載入中..., 稍候...", textVisible: true, html: "<b>載入中..., 請稍候...</b>"}
);
}

function hideHandler(){
//$.mobile.hidePageLoadingMsg(); //1.4已不支援此method
$.mobile.loading("hide");
}
</script>
```

61

註冊事件

- Sample7-3.html

```
<script>
$(document).on("mobileinit", mobileinitHandler);
$(document).on("click", "#backButton", backButtonHandler);

function mobileinitHandler (){
alert("mobileinit");
}

function backButtonHandler (){
$.mobile.changePage("#home");
}
</script>
```

62

ignoreContentEnabled

- Sample7-4.html

```
<script>
$(document).on("mobileinit", mobileinitHandler);

function mobileinitHandler (){
  $.mobile.ignoreContentEnabled=true; //需搭配data-enhance來使用...
}
</script>

<!-- 省略程式碼-->
<div data-role="content" data-enhance="false" style="margin: 0px 0 0 15px">
  <div data-role="fieldcontain">
    <label for="name">Name:</label>
  </div>
</div>

<!-- 省略程式碼-->
```

63

ns

- Sample7-6.html

```
<script>
$(document).on("mobileinit", mobileinitHandler);

function mobileinitHandler (){
  $.mobile.ns = "ucom-";
}
</script>

<!-- 省略程式碼-->

<a href="NoPage.html" data-ucom-role="button" data-ucom-icon="arrow-r"
  data-ucom-iconpos="right">External Next</a>

<!-- 省略程式碼-->
```

64

autoInitializePage

- Sample7-7.html

```
<script>
$(document).on("mobileinit", mobileinitHandler);
function mobileinitHandler (){
    $.mobile.autoInitializePage = false;
}

$(document).ready(readyHandler);
function readyHandler(){
    $("body").html("<div data-role='page'><div data-role='header'>"
        + "<h1>Test Title</h1></div><div data-role='content'>"
        + "<p>Page content goes here.</p></div><div data-role='footer'>"
        + "<h4>Page Footer</h4></div></div>");

    $.mobile.initializePage();
    $.mobile.loading("hide");
}
</script>
```

65

Swipe事件

- swipeleft
- swiperight
- 類比滑鼠事件
 - vmouseover
 - vmousedown
 - vmousemove
 - vmouseup
 - vclick
 - vmousecancel

66

swipe

- Sample7-8.html

```
<script>
$(document).on("mobileinit", mobileinitHandler);

function mobileinitHandler (){
    $.mobile.defaultPageTransition = "slide";
    $(document).on("pageinit", "#home", homelnitHandler);
}

function homelnitHandler(){
    $(".swipe").bind("swipeleft", function(){
        alert("SWIPE LEFT");
        $.mobile.changePage("#right");
    });

    $(document).on("swiperight", ".swipe", function(){
        alert("SWIPE RIGHT");
        $.mobile.changePage("#left", {reverse:true});
    });
}
</script>
```

67

orientationChangeEnabled

- Sample7-9.html

```
<script>
$(document).on("mobileinit", mobileinitHandler);

function mobileinitHandler (){
    $.mobile.orientationChangeEnabled = true;
    $(window).on("orientationchange", changeHandler);
}

function changeHandler(){
    alert(window.orientation);
    if(window.orientation==0){
        $("#orientation").text("Portrait");
    }else{
        $("#orientation").text("Landscape");
    }
}
</script>
```

68

\$.mobile.loadPage

- Sample7-10.html

```
<script>
$(document).on("mobileinit", mobileinitHandler);

function mobileinitHandler(){
  $(document).on("pageinit", "#home", homeInitHandler);
}

function homeInitHandler(){
  $(document).on("click", "#detailButton", detailHandler);
  $.getScript("Detail.js");
}

function detailHandler(){
  $.mobile.changePage("Detail.html", {transition:"slide"});
}
</script>
```

69

jqmData(), jqmRemoveData()

- Sample7-11.html

```
<script>
$(document).on("mobileinit", mobileinitHandler);
function mobileinitHandler (){
  $.mobile.ns = "jqm-";
  $(document).on("pageshow", ".jqmData(role='page')", pageHandler);
}

function pageHandler(){
  var whichPageId = $.mobile.activePage.attr( "id" );
  //alert("This is " + whichPageId);
  if(whichPageId=="home"){
    $(document).on("click", "#resultButton1", resultHandler);
  }else{
    $(document).on("click", "#resultButton2", resultHandler);
  }
}

function resultHandler(){
  var button = $(".jqmData(role='button')").first();
  var result = button.jqmData("icon");
  alert(result);
}
</script>
```

70

\$.mobile.silentScroll

- Sample7-12.html

```
<script>
$(document).on("mobileinit", mobileinitHandler);
function mobileinitHandler (){
    $(document).on("pageinit", "#home", pageInitHandler);
}

function pageInitHandler(){
    $("#scrollButton").bind("click", function(){
        $.mobile.silentScroll(150);
    });
}
</script>
```

71

mobile.path.makeUrlAbsolute

- Sample7-13.html

```
<script>
$(document).on("mobileinit", mobileinitHandler);
function mobileinitHandler (){
    var url1 = "../result.png";
    var url2 = "http://somewhere/img/a/pic.png";

    var resultURL = $.mobile.path.makeUrlAbsolute(url1, url2);
    alert(resultURL);

    var p1 = "../result.png";
    var p2 = "/somewhere/img/a/pic.png";
    var resultPath = $.mobile.path.makePathAbsolute(p1, p2);

    alert(resultPath);
}
</script>
```

72

mobile.path.isRelativeUrl

- Sample7-14.html

```
<script>
$(document).on("mobileinit", mobileinitHandler);
function mobileinitHandler(){
    $(document).on("pageinit", "#home", homelnitHandler);
}

function homelnitHandler(){
    $("#checkRelativeButton").bind("click", checkRelativeHandler);
}

function checkRelativeHandler(){
    //      var url = "../result.png";
    //      var url = http://somewhere/img/a/pic.png;
    var url = "#anchor";
    var resultURL = $.mobile.path.isRelativeUrl(url);
    alert(resultURL);
}
</script>
```

73

下載安裝PhoneGap

第八章

安裝PhoneGap

- 下載PhoneGap
 - <http://phonegap.com/>
 - 解壓縮至 `~/Desktop/app/` 路徑中(解壓縮前請先清空 `~/Desktop/app/` 的內容)
- 下載課程練習資源Template
 - <http://goo.gl/IPbrNa>
 - 解壓縮至 `~/Desktop/app/` 路徑中

75

建立iOS專案

- 找到PhoneGap/lib/ios
- 開啟「終端機」，並切換路徑至 `PhoneGap/lib/ios/bin`
- 執行建立專案的指令：

```
./create ~/Desktop/app/HelloWorld_iOS com.test.HelloWorld HelloWorld
```

- 用Xcode打開專案並執行之

76

在Mac OS建立Android專案

- 找到PhoneGap/lib/android
- 開啟「Eclipse」確定已安裝Android SDK，並找出其安裝路徑
- 開啟「終端機」，並切換路徑至PhoneGap/lib/android/bin
- 執行下列export PATH指令

```
export PATH=${PATH}:/Applications/eclipse/android-sdk-macosx/platform-tools:
/Applications/eclipse/android-sdk-macosx/tools
```

- 執行建立專案的指令：

```
./create ~/Desktop/app/HelloWorld_android com.test.HelloWorld HelloWorld
```

- 從「Eclipse」建立新專案對應到剛才建立之專案並測試執行

77

在Windows上建立Android專案

- 找到/PhoneGap/lib/android
- 開啟「Eclipse」確定其中已安裝Android SDK，並找出安裝路徑
- 在PATH環境變數中加入：

```
...;/Android SDK/platform-tools;/Android SDK/tools;XXX/ant.bat;
```

- 執行「cmd」，並切換路徑至/PhoneGap/lib/android/bin
- 執行建立專案的指令：

```
./create ~/Desktop/app/HelloWorld_android com.test.HelloWorld HelloWorld
```

- 從「Eclipse」建立新專案對應到剛才建立的專案並測試執行

78

PhoneGap API

	iPhone / Phone 3G	Phone 3GS and newer	Android	BlackBerry OS 5.x	BlackBerry OS 6.0+	WebOS	Windows Phone 7 + 8	Symbian	Bada
Accelerometer	✓	✓	✓	✓	✓	✓	✓	✓	✓
Camera	✓	✓	✓	✓	✓	✓	✓	✓	✓
Compass	×	✓	✓	×	×	✓	✓	×	✓
Contacts	✓	✓	✓	✓	✓	×	✓	✓	✓
File	✓	✓	✓	✓	✓	×	✓	×	×
Geolocation	✓	✓	✓	✓	✓	✓	✓	✓	✓
Media	✓	✓	✓	×	×	×	✓	×	×
Network	✓	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Alert)	✓	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Sound)	✓	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Vibration)	✓	✓	✓	✓	✓	✓	✓	✓	✓
Storage	✓	✓	✓	✓	✓	✓	✓	×	×

79

PhoneGap Notification

第九章

Notification

- notification.alert
- notification.confirm
- notification.beep
- notification.vibrate

81

初始化事件

- mobileinit
- pageinit
- deviceready

82

建立iOS專案

- 找到PhoneGap/lib/ios
- 開啟「終端機」，並切換路徑至PhoneGap/lib/ios/bin
- 執行建立專案的指令：

```
./create ~/Desktop/app/Notification_iOS com.test.Notification Notification
```

- 將HTML5_APP_Template/PhoneGap/Notification/*.*複製到iOS專案的www路徑中
- 用Xcode打開專案，開啟並修改www/index.html

83

加上程式(1)

- 開啟專案(/assets)/www/index.html

```
function devicereadyHandler(){
    $("#alertButton").on("click", function(){
        //alert("Hello");
        navigator.notification.alert("歡迎來到PhoneGap的世界！",
            alertDismissHandler, "Question", "確定");
    });

    $("#confirmButton").on("click", confirmHandler);

    $("#beepButton").on("click", function(){
        navigator.notification.beep(5); //iOS中次數設定無作用
    });

    $("#vibrateButton").on("click", function(){
        navigator.notification.vibrate(5000);
    });
}
```

84

加上程式(2)

```
function alertDismissHandler(){
    alert("alertDismissHandler");
}

function confirmHandler(){
    navigator.notification.confirm("你喜歡PhoneGap嗎？", confirmCallbackHandler,
    "Question", "YES, NO");
}

function confirmCallbackHandler(buttonIndex){
    if(buttonIndex==1){ //YES
        navigator.notification.alert("很好! :) ", null, "Answer", "OK");
    }else if(buttonIndex==2){ //NO
        navigator.notification.alert("真可惜! :( ", null, "Answer", "OK");
    }
}
```

85

在Mac OS建立Android專案

- 找到PhoneGap/lib/android
- 開啟「終端機」，並切換路徑至PhoneGap/lib/android/bin
- 執行下列export PATH指令

```
export PATH=${PATH}:/Applications/eclipse/android-sdk-macosx/platform-tools:
/Applications/eclipse/android-sdk-macosx/tools
```

- 執行建立專案的指令：

```
./create ~/Desktop/app/Notification_android com.test.Notification Notification
```

- 將HTML5_APP_Template/PhoneGap/Notification/*.*複製到此專案的assets/www路徑中
- 將iOS改好的index.html複製到此專案的assets/www路徑中
- 從「Eclipse」建立新專案對應到剛才建立之專案並測試執行

86

PhoneGap Device API

第十章

Device

- device.model
- device.cordova
- device.platform
- device.uuid
- device.version

建立iOS專案

- 找到PhoneGap/lib/ios
- 開啟「終端機」，並切換路徑至PhoneGap/lib/ios/bin
- 執行建立專案的指令：

```
./create ~/Desktop/app/Device_iOS com.test.Device Device
```

- 將HTML5_APP_Template/PhoneGap/Device/*.*複製到iOS專案的www路徑中
- 用Xcode打開專案，開啟並修改www/index.html

89

加上程式

- 開啟專案(/assets)/www/index.html

```
$(document).ready(function(e) {
    $(document).on("deviceready", deviceReadyHandler);
});

function deviceReadyHandler(){
    var info = "Name: " + device.model + "<br>";
    info += "Cordova: " + device.cordova + "<br>";
    info += "Platform: " + device.platform + "<br>";
    info += "Version: " + device.version + "<br>";
    info += "UUID: " + device.uuid;
    $("#status").html(info);
}
```

90

在Mac OS建立Android專案

- 找到PhoneGap/lib/android
- 開啟「終端機」，並切換路徑至PhoneGap/lib/android/bin
- 執行下列export PATH指令

```
export PATH=${PATH}:/Applications/eclipse/android-sdk-macosx/platform-tools:  
/Applications/eclipse/android-sdk-macosx/tools
```

- 執行建立專案的指令：

```
./create ~/Desktop/app/Device_android com.test.Device Device
```

- 將HTML5_APP_Template/PhoneGap/Device/*.html複製到此專案的assets/www路徑中
- 將iOS改好的index.html複製到此專案的assets/www路徑中
- 從「Eclipse」建立新專案對應到剛才建立之專案並測試執行

91

PhoneGap Event API

第十一章

事件類型

- deviceready
- pause
- resume
- online
- offline
- batterycritical
- batterylow
- batterystatus

93

建立iOS專案

- 找到PhoneGap/lib/ios
- 開啟「終端機」，並切換路徑至PhoneGap/lib/ios/bin
- 執行建立專案的指令：

```
.create ~/Desktop/app/Event_iOS com.test.Event Event
```

- 將HTML5_APP_Template/PhoneGap/Event/*. *複製到iOS專案的www路徑中
- 用Xcode打開專案，開啟並修改www/index.html

94

加上程式(1)

- 開啟專案(/assets)/www/index.html

```
$(document).ready(function(e) {
    $(document).on("deviceready", deviceReadyHandler);

    $(document).on("pause", function(){$("#log").append("<li>Pause!</li>");});

    $(document).on("resume", function(){$("#log").append("<li>Resume!</li>");});

    $(document).on("online", function(){
        $("#log").append("<li>On Line!</li>");
        $("#log").append("<li>OnLine Type: " + navigator.connection.type + "</li>");
    });

    $(document).on("offline", function(){
        $("#log").append("<li>Off Line!</li>");
    });
    //後面還有...
});
```

95

加上程式(2)

```
$(document).ready(function(e) {
    //前面省略....
    $(document).on("batterycritical", function(info){
        $("#log").append("<li>Battery Critical: " + info.level + "%</li>");
    });

    $(document).on("batterylow", function(info){
        $("#log").append("<li>Battery Low: " + info.level + "%</li>");
    });

    $(document).on("batterystatus", function(info){
        $("#log").append("<li>Battery Status: " + info.level + "%</li>");
    });
});
```

96

在Mac OS建立Android專案

- 找到PhoneGap/lib/android
- 開啟「終端機」，並切換路徑至PhoneGap/lib/android/bin
- 執行下列export PATH指令

```
export PATH=${PATH}:/Applications/eclipse/android-sdk-macosx/platform-tools:  
/Applications/eclipse/android-sdk-macosx/tools
```

- 執行建立專案的指令：

```
./create ~/Desktop/app/Event_android com.test.Event Event
```

- 將HTML5_APP_Template/PhoneGap/Event/*. *複製到此專案的assets/www路徑中
- 將iOS改好的index.html複製到此專案的assets/www路徑中
- 從「Eclipse」建立新專案對應到剛才建立之專案並測試執行

97

第十二章

PhoneGap Connection API

connection

- connection.type(直接用11章的練習)

```
$(document).on("deviceready", function () {
    alert('Connection type: ' + checkConnection());
});

function checkConnection() {
    var networkState = navigator.connection.type;

    var states = {};
    states[Connection.UNKNOWN] = 'Unknown connection';
    states[Connection.ETHERNET] = 'Ethernet connection';
    states[Connection.WIFI] = 'WiFi connection';
    states[Connection.CELL_2G] = 'Cell 2G connection';
    states[Connection.CELL_3G] = 'Cell 3G connection';
    states[Connection.CELL_4G] = 'Cell 4G connection';
    states[Connection.NONE] = 'No network connection';
    return states[networkState];
}
```

第十三章

PhoneGap Geolocation API

Geolocation

- geolocation.getCurrentPosition
- geolocation.watchPosition
- geolocation.clearWatch

建立iOS專案

- 找到PhoneGap/lib/ios
- 開啟「終端機」，並切換路徑至PhoneGap/lib/ios/bin
- 執行建立專案的指令：

```
. /create ~/Desktop/app/Geolocation_ios com.test.Geolocation Geolocation
```

- 將HTML5_APP_Template/PhoneGap/Geolocation/*.複製到iOS專案的www路徑中
- 用Xcode打開專案，開啟並修改www/index.html

geolocation程式(1)

```
function watchButtonHandler() {
    if(navigator.geolocation){
        watchID = navigator.geolocation.watchPosition(
            successHandler, errorHandler,
            {"maximumAge":1000, "timeout":3000}
        );
    }else{
        alert("No PhoneGapService!");
    }
}
```

geolocation程式(2)

```
function successHandler(e) {
    var location = e.coords.latitude + ", " +
                  e.coords.longitude;

    $("#location").val(location);
    $("#info").text("Times:" + times++ + ", " + new Date());
    try{
        latlng = new google.maps.LatLng(
            e.coords.latitude, e.coords.longitude);
    }catch(err){
        alert("There was an error: " + err.message + "!\n");
    }
}

function errorHandler() {
    navigator.notification.alert("GPS Service is not available!",
        null, "ERROR", "OK");
}
```

geolocation程式(3)

```
function stopButtonHandler() {
    if(navigator.geolocation){
        navigator.geolocation.clearWatch(watchID);
    }
    $("#info").html("停止定位服務");
    times=0;
}
```

在Mac OS建立Android專案

- 找到PhoneGap/lib/android
- 開啟「終端機」，並切換路徑至PhoneGap/lib/android/bin
- 執行下列export PATH指令

```
export PATH=${PATH}:/Applications/eclipse/android-sdk-macosx/platform-tools:
/Applications/eclipse/android-sdk-macosx/tools
```

- 執行建立專案的指令：

```
./create ~/Desktop/app/Geolocation_android com.test.Geolocation Geolocation
```

- 將HTML5_APP_Template/PhoneGap/Geolocation/*. *複製到此專案的assets/www路徑中
- 將iOS改好的index.html複製到此專案的assets/www路徑中
- 從「Eclipse」建立新專案對應到剛才建立之專案並測試執行

第14章

PhoneGap Compass API

Compass

- `compass.getCurrentHeading`
- `compass.watchHeading`
- `compass.clearWatch`

建立iOS專案

- 找到PhoneGap/lib/ios
- 開啟「終端機」，並切換路徑至PhoneGap/lib/ios/bin
- 執行建立專案的指令：

```
.create ~/Desktop/app/Compass_ios com.test.Compass Compass
```

- 將HTML5_APP_Template/PhoneGap/Compass/*.複製到iOS專案的www路徑中
- 用Xcode打開專案，開啟並修改www/index.html

Compass

```
function devicereadyHandler() {
    navigator.compass.watchHeading(successHandler, errorHandler, {frequency:100});
}

function successHandler(e){
    var headingInfo = parseInt(e.magneticHeading);
    $("#degree").text(headingInfo);
}

function errorHandler(){
    navigator.notification.alert("Compass service is not available!", null, "ERROR", "OK");
}
```

在Mac OS建立Android專案

- 找到PhoneGap/lib/android
- 開啟「終端機」，並切換路徑至PhoneGap/lib/android/bin
- 執行下列export PATH指令

```
export PATH=${PATH}:/Applications/eclipse/android-sdk-macosx/platform-tools:  
/Applications/eclipse/android-sdk-macosx/tools
```

- 執行建立專案的指令：

```
./create ~/Desktop/app/Compass_android com.test.Compass Compass
```

- 將HTML5_APP_Template/PhoneGap/Compass/*.html複製到此專案的assets/www路徑中
- 將iOS改好的index.html複製到此專案的assets/www路徑中
- 從「Eclipse」建立新專案對應到剛才建立之專案並測試執行

111

第15章

PhoneGap Acceleration API

Acceleration

- `accelerometer.getCurrentAcceleration()`
- `accelerometer.watchAcceleration()`
- `accelerometer.clearWatch ()`

建立iOS專案

- 找到PhoneGap/lib/ios
- 開啟「終端機」，並切換路徑至PhoneGap/lib/ios/bin
- 執行建立專案的指令：

```
./create ~/Desktop/app/Accelerometer_ios com.test.Accelerometer Accelerometer
```

- 將HTML5_APP_Template/PhoneGap/Accelerometer/*.*複製到iOS專案的www路徑中
- 用Xcode打開專案，開啟並修改www/index.html

accelerometer程式(1)

```
function devicereadyHandler(){
  try{
    navigator.accelerometer.watchAcceleration(
      successHandler, errorHandler, {"frequency":100});
  }catch(ex){ console.log(ex); }
}

var range = [-20.0,40.0, $(window).width()-110.0, $(window).height()-160.0];

function successHandler(e){
  var xinfo = "x: " + e.x + "<br>";
  var yinfo = "y: " + e.y + "<br>";
  var zinfo = "z: " + e.z + "<br>";

  $("#status").html(xinfo+yinfo+zinfo);
  var posX = $("#robot").position().left;
  var posY = $("#robot").position().top;

  //變更robot的位置，程式在下一頁
}
```

accelerometer程式(2)

```
function successHandler(e){
  //前面程式省略，請見上頁
  //變更robot的位置：
  posX-=e.x;
  posY+=e.y;

  if (posX <= range[0]){
    posX = range[0];
  } else if(posX >= range[2]){
    posX = range[2];
  }

  if (posY<=range[1]){
    posY = range[1];
  }else if(posY >= range[3]){
    posY = range[3];
  }

  $("#robot").css({"left":posX+"px", "top":posY+"px"});
}
```

在Mac OS建立Android專案

- 找到PhoneGap/lib/android
- 開啟「終端機」，並切換路徑至PhoneGap/lib/android/bin
- 執行下列export PATH指令

```
export PATH=${PATH}:/Applications/eclipse/android-sdk-macosx/platform-tools:  
/Applications/eclipse/android-sdk-macosx/tools
```

- 執行建立專案的指令：

```
./create ~/Desktop/app/Accelerometer_android com.test.Accelerometer Accelerometer
```

- 將HTML5_APP_Template/PhoneGap/Accelerometer/*.*複製到此專案的assets/www路徑中
- 將iOS改好的index.html複製到此專案的assets/www路徑中
- 從「Eclipse」建立新專案對應到剛才建立之專案並測試執行

117

第16章

PhoneGap Camera API

Camera

- camera.getPicture(successHandler, errorHandler,
- {options,...})
- options:
 - sourceType
 - destinationType
 - allowEdit
 - quality
 - encodingType

建立iOS專案

- 找到PhoneGap/lib/ios
- 開啟「終端機」，並切換路徑至PhoneGap/lib/ios/bin
- 執行建立專案的指令：

```
./create ~/Desktop/app/Camera_ios com.test.Camera Camera
```

- 將HTML5_APP_Template/PhoneGap/Camera/*.*複製到iOS專案的www路徑中
- 用Xcode打開專案，開啟並修改www/index.html

Camera

```
function pageInitHandler(){
$("#shootButoon").on("click", function(){
/*
navigator.camera.getPicture(successHandler, errorHandler,
{ "sourceType":Camera.PictureSourceType.PHOTOLIBRARY,
"destinationType":Camera.DestinationType.DATA_URL}); //FILE_URI
*/

navigator.camera.getPicture(successHandler, errorHandler, {
"sourceType":Camera.PictureSourceType.CAMERA,
"destinationType":Camera.DestinationType.DATA_URL,
"allowEdit":false, "quality":50,"encodingType":Camera.EncodingType.JPEG});
}

function successHandler(imageData){
//$("#photo").get(0).src=imageUrl; //FILE_URI
$("#photo").get(0).src = "data:image/jpeg;base64," + imageUrl; //DATA_URI
}
```

在Mac OS建立Android專案

- 找到PhoneGap/lib/android
- 開啟「終端機」，並切換路徑至PhoneGap/lib/android/bin
- 執行下列export PATH指令

```
export PATH=${PATH}:/Applications/eclipse/android-sdk-macosx/platform-tools:
/Applications/eclipse/android-sdk-macosx/tools
```

- 執行建立專案的指令：

```
./create ~/Desktop/app/Camera_android com.test.Camera Camera
```

- 將HTML5_APP_Template/PhoneGap/Camera/*.*複製到此專案的assets/www路徑中
- 將iOS改好的index.html複製到此專案的assets/www路徑中
- 從「Eclipse」建立新專案對應到剛才建立之專案並測試執行

第17章

PhoneGap Capture API

Capture

- `capture.captureAudio`
- `capture.captureImage`
- `capture.captureVideo`

建立iOS專案

- 找到PhoneGap/lib/ios
- 開啟「終端機」，並切換路徑至PhoneGap/lib/ios/bin
- 執行建立專案的指令：

```
./create ~/Desktop/app/Capture_ios com.test.Capture Capture
```

- 將HTML5_APP_Template/PhoneGap/Capture/*.*複製到iOS專案的www路徑中
- 用Xcode打開專案，開啟並修改www/index.html

Capture

```
function devicereadyHandler (){
    $("#audioButton").on("click", function(){
        navigator.device.capture.captureAudio(successHandler, errorHandler, {"limit":2});
    });
    $("#imageButton").on("click", function(){
        navigator.device.capture.captureImage(successHandler, errorHandler, {"limit":2});
    });
    $("#videoButton").on("click", function(){
        navigator.device.capture.captureVideo(successHandler, errorHandler, {"limit":2});
    });
}

function successHandler(mediaFiles){
    var info = "You have recorded " + mediaFiles.length + " Media files <br>";
    for(var i=0;<mediaFiles.length;i++){
        info += mediaFiles[i].name + "<br>";
    }
    $("#status").html(info);
}

function errorHandler(){
    navigator.notification.alert("Capture Error!", null, "ERROR", "OK");
}
```

在Mac OS建立Android專案

- 找到PhoneGap/lib/android
- 開啟「終端機」，並切換路徑至PhoneGap/lib/android/bin
- 執行下列export PATH指令

```
export PATH=${PATH}:/Applications/eclipse/android-sdk-macosx/platform-tools:  
/Applications/eclipse/android-sdk-macosx/tools
```

- 執行建立專案的指令：

```
./create ~/Desktop/app/Capture_android com.test.Capture Capture
```

- 將HTML5_APP_Template/PhoneGap/Capture/*.html複製到此專案的assets/www路徑中
- 將iOS改好的index.html複製到此專案的assets/www路徑中
- 從「Eclipse」建立新專案對應到剛才建立之專案並測試執行

127

第18章

PhoneGap Media API

Media

- media.getCurrentPosition
- media.getDuration
- media.play
- media.pause
- media.release
- media.seekTo
- media.startRecord
- media.stopRecord
- media.stop

建立iOS專案

- 找到PhoneGap/lib/ios
- 開啟「終端機」，並切換路徑至PhoneGap/lib/ios/bin
- 執行建立專案的指令：

```
./create ~/Desktop/app/Media_ios com.test.Media Media
```

- 將HTML5_APP_Template/PhoneGap/Media/*.html複製到iOS專案的www路徑中
- 用Xcode打開專案，開啟並修改www/index.html

Media範例(1)

```
function playButtonHandler(){
    alert(device.platform);
    if(media==undefined){
        if(device.platform=="Android"){
            media = new Media("file:///android_asset/www/song.mp3",
                successHandler, errorHandler);
        }else{
            media = new Media("song.mp3",successHandler, errorHandler);
        }
    }
    media.play();
}

function successHandler(){
    navigator.notification.alert("Play Success!", null, "SUCCESS", "OK");
}

function errorHandler(){
    navigator.notification.alert("Play Failed!", null, "ERROR", "OK");
}
```

Media範例(2)

```
function pauseButtonHandler(){
    if(media){
        media.pause();
    }
}

function stopButtonHandler(){
    if(media){
        media.stop();
        media.release();
        media = undefined;
    }
}
```

在Mac OS建立Android專案

- 找到PhoneGap/lib/android
- 開啟「終端機」，並切換路徑至PhoneGap/lib/android/bin
- 執行下列export PATH指令

```
export PATH=${PATH}:/Applications/eclipse/android-sdk-macosx/platform-tools:
/Applications/eclipse/android-sdk-macosx/tools
```

- 執行建立專案的指令：

```
./create ~/Desktop/app/Media_android com.test.Media Media
```

- 將HTML5_APP_Template/PhoneGap/Media/*.*複製到此專案的assets/www路徑中
- 將iOS改好的index.html複製到此專案的assets/www路徑中
- 從「Eclipse」建立新專案對應到剛才建立之專案並測試執行

133

第19章

WebService in Phonegap

使用jQuery來傳送ajax請求

- `$.ajax({...})`
 - `url`: urlStr,
 - `type`: "POST",
 - `dataType`: "xml",
 - `contentType`: "text/xml; charset=utf-8",
 - `data`: soapMessage,
 - `beforeSend`: function(xhr){...}
 - `xhr.setRequestHeader("header", "value");`
 - `success`: processSuccessHandler,
 - `error`: processErrorHandler,
 - `complete`: processCompleteHandler

135

處理xml

- `window.XMLSerializer`
 - 或用 `xhr.xml` 來取得xml
- `$.parseXML(xml)`

136

建立iOS專案

- 找到PhoneGap/lib/ios
- 開啟「終端機」，並切換路徑至PhoneGap/lib/ios/bin
- 執行建立專案的指令：

```
.create ~/Desktop/app/WebService_ios com.test.WebService WebService
```

- 將HTML5_APP_Template/PhoneGap/WebService/*.複製到iOS專案的www路徑中
- 用Xcode打開專案，開啟並修改www/index.html

程式示範(1)

```
function pageInitHandler() {
    $("#convertButton").on("click", convertHandler);
}

function convertHandler () {
    var fromStr = $("#fromInput").val();
    var toStr = $("#toInput").val();
    var soapMessage = "<soap:Envelope xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance' " +
        "xmlns:xsd='http://www.w3.org/2001/XMLSchema' " +
        "xmlns:soap='http://schemas.xmlsoap.org/soap/envelope/'>"+
        "<soap:Body><ConversionRate xmlns='http://www.webserviceX.NET/'>"+
        "<FromCurrency>"+fromStr+"</FromCurrency><ToCurrency>"+toStr+"</ToCurrency>"+
        "</ConversionRate></soap:Body></soap:Envelope>";

    //這裡要呼叫ajax...請看下一頁
}
```

程式示範(2)

```
function convertHandler (){
  //前面省略...請看上一頁
  //這裡要呼叫ajax...
  try{
    $.ajax({
      url: "http://www.webserviceX.NET/CurrencyConvertor.asmx",
      type:"POST",
      dataType: "xml",
      contentType: "text/xml; charset=utf-8",
      data: soapMessage,
      beforeSend: function(xhr){
        xhr.setRequestHeader("SOAPAction", "http://www.webserviceX.NET/ConversionRate");
      },
      success: processSuccessHandler,
      error: processErrorHandler,
      complete: processCompleteHandler
    });

    $.mobile.loading("show");
  }catch(err){
    alert("error:" + err);
  }
}
```

139

程式示範(3)

```
function processSuccessHandler(xmlHttpRequest, status){
  var xml;
  if (window.XMLSerializer!=undefined) {
    xml =(new window.XMLSerializer()).serializeToString(xmlHttpRequest);
    //alert("XMLSerializer:" + xml);
  }else if(xmlHttpRequest.xml!=undefined) {
    xml =xmlHttpRequest.xml
    //alert("xmlHttpRequest.xml:" + xmlHttpRequest.xml);
  }
  var xmlDoc = $.parseXML(xml),
  $xml = $(xmlDoc),
  $title = $xml.find( "ConversionRateResult" );
  $("#result").val($title.text());
}
```

140

程式示範(4)

```
function processErrorHandler(xhr, status, error){
    var errMsg;
    if(status=="error"){
        errMsg = "無法取得服務，請更正或稍候再試!"; // "status:" + status + ", error:" + error;
    }else{
        errMsg = "無法取得服務，請稍候再試！";
    }
    if(navigator.notification){
        navigator.notification.alert(errMsg);
    }else{
        alert(errMsg);
    }
}

function processCompleteHandler(xmlHttpRequest, status){
    //alert("complete: " + xmlHttpRequest + ", status:" + status);
    $.mobile.loading("hide");
}
```

141

在Mac OS建立Android專案

- 找到PhoneGap/lib/android
- 開啟「終端機」，並切換路徑至PhoneGap/lib/android/bin
- 執行下列export PATH指令

```
export PATH=${PATH}:/Applications/eclipse/android-sdk-macosx/platform-tools:
/Applications/eclipse/android-sdk-macosx/tools
```

- 執行建立專案的指令：

```
./create ~/Desktop/app/WebService_android com.test.WebService WebService
```

- 將HTML5_APP_Template/PhoneGap/WebService/*. *複製到此專案的assets/www路徑中
- 將iOS改好的index.html複製到此專案的assets/www路徑中
- 從「Eclipse」建立新專案對應到剛才建立之專案並測試執行

142

第20章

PhoneGap File API

File

- File Entry
- Directory Entry

建立iOS專案

- 找到PhoneGap/lib/ios
- 開啟「終端機」，並切換路徑至PhoneGap/lib/ios/bin
- 執行建立專案的指令：

```
.create ~/Desktop/app/File_ios com.test.File File
```

- 將HTML5_APP_Template/PhoneGap/File/*.*複製到iOS專案的www路徑中
- 用Xcode打開專案，開啟並修改www/index.html

File範例(1) 取得FileSystem

```
function devicereadyHandler()
{
    //window.requestFileSystem(LocalFileSystem.TEMPORARY, 0, fileSystemSuccessHandler,
    errorHandler);
    window.requestFileSystem(LocalFileSystem.PERSISTENT, 0, fileSystemSuccessHandler,
    errorHandler);
}

function fileSystemSuccessHandler(fileSystem){
    // alert(fileSystem.root.fullPath);
    fs = fileSystem;
    logData("File System ready");

    console.log(fileSystem.root.fullPath);
}

function errorHandler(e){
    navigator.notification.alert("File process has got some trouble:"
    + e.code , null, "ERROR", "OK");
}
```

File範例(2) Create File

```
function deviceHandler(){
    $(document).on("click", "#createDirButton", createHandler);
    $(document).on("click", "#listDirButton", listHandler);
    $(document).on("click", "#removeDirButton", removeHandler);
    $(document).on("click", "#readFileButton", readFileHandler);
    $(document).on("click", "#writeFileButton", writeFileHandler);
    $(document).on("click", "#removeFileButton", removeFileHandler);
}

function createHandler(){
    fs.root.getDirectory("newDir", {create:true}, createDirSuccessHandler, errorHandler);
}

function createDirSuccessHandler(dirEntry){
    newDir = dirEntry;
    logData("getDirectory Success!");
}
```

File範例(3) List

```
function listHandler(){
    var docReader = fs.root.createReader();
    docReader.readEntries(listDirSuccessHandler, errorHandler);
}

function listDirSuccessHandler(entries){
    logData("You found " + entries.length + " entries:");
    for(var i=0;i<entries.length;i++){
        logData(entries[i].name);
    }
}
```

File範例(4) remove Dir

```
function removeHandler(){
  if(newDir){
    newDir.remove(removeDirSuccessHandler, errorHandler);
  }else{
    errorHandler();
  }
}

function removeDirSuccessHandler(){
  logData("Remove dir success.");
}
```

File範例(5) Read File

```
function readFileHandler(){//取得或建立Sample.txt
  fs.root.getFile("Sample.txt", {create:true}, readFileSuccessHandler, errorHandler);
}

function readFileSuccessHandler(fileEntry){
  sampleFile = fileEntry;
  logData("Get file success!");
  sampleFile.file(fileReaderHandler, errorHandler); //取得檔案的fileRef
}

function fileReaderHandler(fileRef){ //讀取fileRef中的內容
  var reader = new FileReader();
  reader.onloadend = fileLoadCompleteHandler;
  reader.readAsText(fileRef); //開始讀取該文字檔案的內容
}

function fileLoadCompleteHandler(e){
  //在讀取文字檔案內容完成時的事件將結果顯示在文字欄位中
  var result = e.target.result;
  $("#editor").val(result);
}
```

File範例(6) Write File

```
function writeFileHandler(){
  if(sampleFile){
    sampleFile.createWriter(fileWriterHandler, errorHandler); //取得檔案的fileWriter
  }else{
    errorHandler("");
  }
}

function fileWriterHandler(writer){
  //writer.onload = fileWriterCompleteHandler;
  writer.onwriteend = fileWriterCompleteHandler;
  writer.write($("#editor").val());
}

function fileWriterCompleteHandler(){
  alert("write File!");
  logData("Write Complete!");
}
```

File範例(7) remove File

```
function removeHandler(){
  if(sampleFile){
    sampleFile.remove(removeFileSuccessHandler, errorHandler);
  }else{
    errorHandler();
  }
}

function removeFileSuccessHandler (){
  logData("Remove File Success.");
}
```

在Mac OS建立Android專案

- 找到PhoneGap/lib/android
- 開啟「終端機」，並切換路徑至PhoneGap/lib/android/bin
- 執行下列export PATH指令

```
export PATH=${PATH}:/Applications/eclipse/android-sdk-macosx/platform-tools:
/Applications/eclipse/android-sdk-macosx/tools
```

- 執行建立專案的指令：

```
./create ~/Desktop/app/File_android com.test.File File
```

- 將HTML5_APP_Template/PhoneGap/File/*. *複製到此專案的assets/www路徑中
- 將iOS改好的index.html複製到此專案的assets/www路徑中
- 從「Eclipse」建立新專案對應到剛才建立之專案並測試執行

153

第21章

PhoneGap Storage API

Storage

- window.openDatabase
- Database
 - transaction
 - changeVersion
- SQLTransaction
 - executeSql
- SQLResultSet

建立iOS專案

- 找到PhoneGap/lib/ios
- 開啟「終端機」，並切換路徑至PhoneGap/lib/ios/bin
- 執行建立專案的指令：

```
.create ~/Desktop/app/Storage_ios com.test.Storage Storage
```

- 將HTML5_APP_Template/PhoneGap/Storage/*.*複製到iOS專案的www路徑中
- 用Xcode打開專案並修改專案

Storage範例

```
var db;
function deviceReadyHandler(){
  db = window.openDatabase("test", "1.0", "test database", 1000);

  $("#createButton").on("click", function(){
    if(db){
      db.transaction(createDBTransactionSuccessHandler, errorHandler, createDbSQLSuccessHandler);
    }
  });

  $("#insertButton").bind("click", function(){
    if(db){
      db.transaction(insertDBTransactionSuccessHandler, errorHandler, insertDBSQLSuccessHandler);
    }
  });

  $("#selectButton").bind("click", function(){
    if(db){
      db.transaction(selectDBTransactionSuccessHandler, errorHandler, selectDBSQLSuccessHandler);
    }
  });
}
```

Storage範例- CreateTable

```
function createDBTransactionSuccessHandler(sqlTransaction){//執行Transaction成功
  //alert("sqlTransaction");
  sqlTransaction.executeSql("DROP TABLE IF EXISTS demo")
  var sqlValue = "CREATE TABLE demo (id INTEGER PRIMARY KEY AUTOINCREMENT, name TEXT)";
  sqlTransaction.executeSql(sqlValue);
}

function createDbSQLSuccessHandler(){//執行SQL成功
  $("#log").append("<i>Create DB Table Success.</i>")
}

function errorHandler(){//執行Transaction失敗
  navigator.notification.alert("SQLite has some problem!", null, "Error", "OK");
}
```

Storage範例- insert

```
function insertDBTransactionSuccessHandler(sqlTransaction){//執行Transaction成功
//alert("insertDBTransactionSuccessHandler");
var nameValue = $("#nameInput").val();
var sqlValue = "INSERT INTO demo (name) VALUES('" + nameValue + "')";
sqlTransaction.executeSql(sqlValue);
$("#log").append("<li>" + sqlValue + "</li>");
}

function insertDBSQLSuccessHandler(){//執行SQL成功
$("#log").append("<li>INSERT DB Table Success.</li>");
}
```

Storage範例-select

```
function selectDBTransactionSuccessHandler(sqlTransaction){//執行Transaction成功
var sqlValue = "SELECT * FROM demo";
sqlTransaction.executeSql(sqlValue, [], successSelectResultSetHandler, errorHandler);
}

function successSelectResultSetHandler(sqlTransaction, sqlResultSet){
//alert(sqlResultSet.rows.length);
var sqlResultSetList = sqlResultSet.rows;
$("#log").append("<li>You got " + sqlResultSetList.length + "records.</li>");

for(var i=0; i<sqlResultSetList.length;i++){
var info = "ID: " + sqlResultSetList.item(i).id + ", Name: " +
sqlResultSetList.item(i).name;
$("#log").append("<li>" + info + "</li>");
}
}

function selectDBSQLSuccessHandler(){//執行SQL成功
$("#log").append("<li>SELECT DB Table Success.</li>");
}
```


在Mac OS建立Android專案

- 找到PhoneGap/lib/android
- 開啟「終端機」，並切換路徑至PhoneGap/lib/android/bin
- 執行下列export PATH指令

```
export PATH=${PATH}:/Applications/eclipse/android-sdk-macosx/platform-tools:  
/Applications/eclipse/android-sdk-macosx/tools
```

- 執行建立專案的指令：

```
./create ~/Desktop/app/Storage_android com.test.Storage Storage
```

- 將HTML5_APP_Template/PhoneGap/Storage/*. *複製到此專案的assets/www路徑中
- 將iOS改好的index.html複製到此專案的assets/www路徑中
- 從「Eclipse」建立新專案對應到剛才建立之專案並測試執行

161

第22章

PhoneGap Contacts API

Contacts

- contacts
 - create
 - find
- Contact
 - remove
 - save

建立iOS專案

- 找到PhoneGap/lib/ios
- 開啟「終端機」，並切換路徑至PhoneGap/lib/ios/bin
- 執行建立專案的指令：

```
./create ~/Desktop/app/Contacts_ios com.test.Contacts Contacts
```

- 將HTML5_APP_Template/PhoneGap/Contacts/*.html複製到iOS專案的www路徑中
- 用Xcode打開專案，開啟並修改www/index.html

Contacts範例(1) find Contacts

```
function pageInitHandler(){
    $("#searchButton").on("click", function(){
        var keyword = $("#keyword").val();
        var fields = ["firstName", "name"];

        var option = new ContactFindOptions();
        option.filter = keyword;
        option.multiple = true;
        navigator.contacts.find(fields, findSuccessHandler, errorHandler, option);
    });

    $("#addButton").on("click", addContactHandler);
}
```

Contacts範例(2) find Contacts

```
function findSuccessHandler(contacts){
    console.log(contacts);
    $("#count").text(contacts.length);
    $("#list").html(""); //清空清單

    for(var i=0; i<contacts.length; i++){
        console.log(contacts[i]);

        var name = (contacts[i].name.familyName==null ? "" :
        contacts[i].name.familyName);
        name+=((contacts[i].name.givenName==null ? "" : contacts[i].name.givenName);
        console.log(name);

        $("#list").append("<li>" + name + "</li>");
    }
}
```

Contacts範例(3) Add Contacts

```
function addContactHandler(){
  var familyName = $("#keyword").val();
  alert(familyName);
  var contact = navigator.contacts.create();
  contact.name = {"familyName":familyName};
  contact.save();

  navigator.notification.alert("Add Contact success!", null, "Contact", "OK");
}

function errorHandler(){
  navigator.notification.alert("Find contacts failed!", null, "ERROR", "OK");
}
```

在Mac OS建立Android專案

- 找到PhoneGap/lib/android
- 開啟「終端機」，並切換路徑至PhoneGap/lib/android/bin
- 執行下列export PATH指令

```
export PATH=${PATH}:/Applications/eclipse/android-sdk-macosx/platform-tools:
/Applications/eclipse/android-sdk-macosx/tools
```

- 執行建立專案的指令：

```
./create ~/Desktop/app/Contacts_android com.test.Contacts Contacts
```

- 將HTML5_APP_Template/PhoneGap/ontacts/*. *複製到此專案的assets/www路徑中
- 將iOS改好的index.html複製到此專案的assets/www路徑中
- 從「Eclipse」建立新專案對應到剛才建立之專案並測試執行