



# JavaScript 程式設計新手村

## 單元25 - Chrome Extensions 基礎介紹

@kdchang

# Outline

1. Chrome Extensions 基礎概念

# 什麼是瀏覽器外掛？

Extensions are small software programs that can modify and enhance the functionality of the Chrome browser. You write them using web technologies such as HTML, JavaScript, and CSS.

Ex. Chrome Extensions, Firefox addon

# Chrome Extensions 基礎概念

# 檔案架構

Extensions 主要是由 HTML/CSS/JavaScript 所組成（也是我們前面單元介紹的內容）

- `manifest.json`
- 至少一個 HTML 檔案（除非是主題佈景可以不用）
- 至少一個 JavaScript 檔案
- 選擇性：CSS、圖片檔案

## 與瀏覽器、網頁互動

Extensions 可以透過 Content scripts 修改使用者瀏覽的頁面和使用 cross-origin XMLHttpRequests 取得不同伺服器資料，也可以透過 API 和瀏覽器 bookmarks 和 tab 互動

# Extensions 身分證 `manifest.json`

每個 Extensions 都會有一個 `manifest.json` 檔案，記錄應用程式的檔名、icon、版本、權限等基本設定資料

```
{
  "manifest_version": 2,

  "name": "Getting started example",
  "description": "This extension shows a Google Image search re
  "version": "1.0",

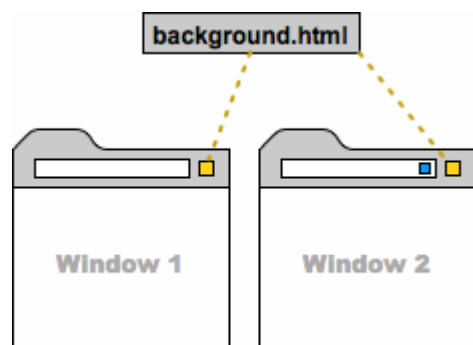
  "browser_action": {
    "default_icon": "icon.png",
    "default_popup": "popup.html"
  },
  "permissions": [
    "activeTab",
    "https://ajax.googleapis.com/"
  ]
}
```

# manifest 格式

```
{  
  // 必要  
  "name": "Extensions 名稱",  
  "version": "版本字串",  
  // 一個代表 manifest file 格式版本的整數， Chrome 18 以上，  
  // 則應該設定成 2  
  
  "manifest_version": 2,  
  
  // 建議使用，但非必要  
  "description": "A plain text description",  
  "icons": { ... },  
  "default_locale": "zh-tw",  
  
  // 選一個主要互動功能模式  
  "browser_action": {...},  
  "page_action": {...},  
  "theme": {...},  
}
```



# Background Pages



外掛通常會有一個 `background.html` 或 `background.js` 常駐。用來放置外掛的主要功能和邏輯，並透過 API 與 UI（browser actions 或 page actions）溝通或是當成 content script 和 UI 中介

```
{  
  ...  
  "background": {  
    // 也可以用 "page": "background.html"  
    "scripts": ["backgroundbackground.js"] 引入多 JavaScript 檔案  
  },  
  ...  
}
```

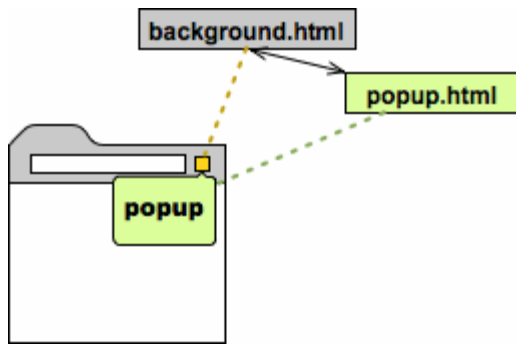
# UI pages

Chrome Extension 的 UI 是使用 HTML/CSS 來呈現，以下是主要的 UI 類型：

- Browser Actions：常駐
- Page Actions：特定頁面
- Desktop Notifications：桌面通知
- Omnibox：網址列右邊
- Override Pages：開新分頁
- Context Menu：點擊右鍵出現小選單

# Browser Actions

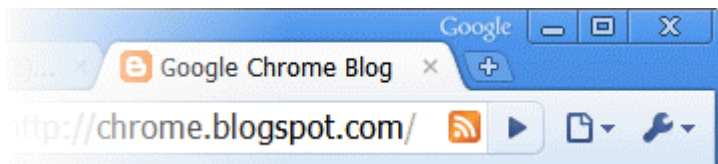
長駐程式，較耗資源 (Ex. 監聽 Email)



```
{  
  ...  
  "browser_action": {  
    "default_icon": "icon19.png", // 可選  
    "default_title": "Tooltip title", // 可選  
    "default_popup": "popup.html" // 可選，滑鼠單擊會呈現的內  
  },  
  ...  
}
```

# Page Actions

與 Browser Actions 類似，但只有特定頁面才會使用，例如：設定  
讀取 RSS 頁面



```
{
  ...
  "page_action": {
    "default_icon": "icons/foo.png", // 可選
    "default_title": "Tooltip title", // 可選
    "default_popup": "popup.html" // 可選，滑鼠單擊 icon 會呈
  },
  ...
}
```

# Content scripts

插入使用者頁面，透過 DOM API 取得使用者瀏覽的頁面的內容，甚至可以改變這些內容（例如：移除臉書廣告等）

```
{  
  ...  
  "content_scripts": [  
    {  
      "matches": ["http://www.google.com/*"],  
      "css": ["mystyles.css"],  
      "js": ["jquery.js", "myscript.js"]  
    }  
  ],  
  ...  
}
```

# matches

指定某個 host 下的所有 path :

```
http://127.0.0.1/*
```

完整網址 :

```
http://example.org/foo/bar.html
```

指定某個連結

# chrome.\* APIs

```
{  
  "name": "My extension",  
  ...  
  "permissions": [  
    "bookmarks"  
  ],  
  ...  
}
```

```
chrome.bookmarks.create({'parentId': bookmarkBar.id,  
                        'title': 'Extension bookmarks'},  
    function(newFolder) {  
        console.log("added folder: " + newFolder.title);  
    });
```

# 可用 API

1. Standard JavaScript APIs
2. XMLHttpRequest
3. HTML5 and other emerging APIs
4. WebKit APIs
5. V8 APIs, such as JSON
6. APIs in bundled libraries (jQuery)

延伸閱讀：[大兜的 Chrome Extension 學習筆記](#)



# 總結

## 1. Chrome Extensions 基礎概念