

# 物聯網實務 HW15

電機碩一 11278008 林佳慧

日期:2024/01/03

# 建立一個Firebase專案



Firebase



佳慧

你的 Firebase 專案



新增專案

AloTdemo

aiotdemo-f60d9



totoras

totoras-57d56



探索範例專案

**Firebase 專案是應用程式的容器**

同一項專案中的所有應用程式會共用即時資



iOS+



# 建立專案

× 建立專案(步驟 3 之 1)

首先，請輸入專案名稱<sup>?</sup>

專案名稱

AIOT2

 aiot2-b948f

繼續

# 分析

## 適用於 Firebase 專案的 Google Analytics (分析)

Google Analytics (分析) 是免費且無使用限制的數據分析解決方案，能讓你在 Firebase Crashlytics、雲端通訊、應用程式內通訊、遠端設定、A/B 測試和 Cloud Functions 中執行指定目標和查看報表等動作。

Google Analytics (分析) 可讓你運用下列功能：

-  A/B 測試<sup>?</sup>
-  所有 Firebase 產品中的使用者區隔和指定目標功能<sup>?</sup>
-  不受當機影響的使用者<sup>?</sup>
-  以事件為基礎的 Cloud Functions 觸發條件<sup>?</sup>
-  沒有使用限制的免費報表<sup>?</sup>

☒ 啟用這項專案的 Google Analytics (分析) 功能  
建議做法


上一部分


繼續

× 建立專案(步驟 3 之 3)

## 設定 Google Analytics (分析)

選取或建立 Google Analytics (分析) 帳戶 ?

 Default Account for Firebase ▼

自動在這個帳戶中建立新資源 

建立專案時，系統會在您選定的 Google Analytics (分析) 帳戶中建立新的 Google Analytics (分析) 資源，並將該資源連結至您的 Firebase 專案。連結完成後，資料就能在產品之間流通。從 Google Analytics (分析) 資源匯出到 Firebase 的資料需符合 Firebase 《服務條款》的規定；而匯入 Google Analytics (分析) 的 Firebase 資料則需符合 Google Analytics (分析) 《服務條款》的規定。[瞭解詳情](#)。

[上一步](#)

建立專案



AIOT2



新專案已準備就緒

繼續



接收 Firebase 新功能、相關研究和活動等最新消息的電子郵件

[立即申請](#)

## AIOT2

Spark 方案

# 將 Firebase 新增至應用程式即可開始使用




首先請新增應用程式



# 勾代管功能

## × 將 Firebase 新增至您的網頁應用程式

### 1 註冊應用程式

應用程式暱稱 

AIOT2



一併為此應用程式設定 **Firebase 託管** 功能。 [瞭解詳情](#)

你也可以之後再設定託管功能。無論何時開始使用託管功能，你都無須付費。



aiot2-b948f (尚未部署任何項目)

註冊應用程式

### 2 新增 Firebase SDK

☒ 使用 npm ☐ 使用 <script> 標記

如果你已使用 [npm](#) 和 [webpack](#) 或 [Rollup](#) 等模組整合工具，則可執行下列指令來安裝最新版 SDK ([瞭解詳情](#)):

```
$ npm install firebase
```

請初始化 Firebase，接著即可開始將 SDK 套用至要使用的產品。

```
// Import the functions you need from the SDKs you need
import { initializeApp } from "firebase/app";
import { getAnalytics } from "firebase/analytics";
// TODO: Add SDKs for Firebase products that you want to use
// https://firebase.google.com/docs/web/setup#available-libraries

// Your web app's Firebase configuration
// For Firebase JS SDK v7.20.0 and later, measurementId is optional
const firebaseConfig = {
  apiKey: "AIzaSyCeZtV0aNLubiEXKzoDzJpW8uDuQuTu5Iw",
  authDomain: "aiot2-b948f.firebaseio.com",
  projectId: "aiot2-b948f",
  storageBucket: "aiot2-b948f.appspot.com",
  messagingSenderId: "334846609242",
  appId: "1:334846609242:web:3961e561657a36e22f1563",
  measurementId: "G-FZ8N3N1YWF"
};

// Initialize Firebase
const app = initializeApp(firebaseConfig);
const analytics = getAnalytics(app);
```

**注意事項：**這個選項會使用 [模組 JavaScript SDK](#)，因此 SDK 的大小得以縮減。

如要進一步瞭解適用於網頁應用程式的 Firebase，請查看下列資源：[開始使用](#)、[Web SDK API 參考資料](#)、[使用範例](#)

下一步

## 新增 Firebase SDK

### 3 安裝 Firebase CLI

如要透過 Firebase 託管功能來代管您的網站，則必須使用 Firebase CLI 這項指令列工具。

執行下列 [npm](#) 指令，藉此安裝 CLI 或更新至最新版 CLI。









```
$ npm install -g firebase-tools
```



無法順利執行操作嗎？您不妨查看 [Firebase CLI 參考資源](#) 或變更您的 [npm 權限](#)

上一步

下一步

-  Authentication
-  App Check
-  Firestore Database
-  Realtime Database
-  Extensions
-  Storage
-  **Hosting**
-  Functions

Spark  
免付費 每月 \$0 美元

[升級](#)

AIOT2 ▾

# Hosting

利用安全的全球內容傳遞網路，在彈指之間部署網路版和行動版網頁應用程式

[開始使用](#)[瞭解詳情](#)



## × 設定 Firebase 託管

### 1 安裝 Firebase CLI

如要透過 Firebase 託管功能來代管您的網站，則必須使用 Firebase CLI 這項指令列工具。

執行下列 [npm](#) 指令，藉此安裝 CLI 或更新至最新版 CLI。

```
$ npm install -g firebase-tools
```



無法順利執行操作嗎？您不妨查看 [Firebase CLI 參考資源](#) 或變更您的 [npm 權限](#)。

下一步

## × 設定 Firebase 託管

### ✓ 安裝 Firebase CLI

### ✓ 初始化您的專案

### 3 部署至 Firebase 託管

準備就緒後，即可部署網頁應用程式

將 HTML、CSS 和 JS 等靜態檔案加入應用程式的部署目錄 (預設為「公開」)。接著，從應用程式的根目錄執行下列指令：

```
$ firebase deploy
```



部署完成之後，請前往「[aiot2-b948f.web.app](#)」查看應用程式  
需要協助嗎？您可以參閱 [代管說明文件](#)。

前往控制台

## 建立工作目錄

```
Node.js command prompt
Your environment has been set up for
d npm.
C:\Users\user>mkdir gantest0103
```

## 進入工作目錄

```
Node.js command prompt
Your environment has been set up for
d npm.
C:\Users\user>mkdir gantest0103
C:\Users\user>cd gantest0103
```

## 安裝工具

```
C:\Users\user>mkdir gantest0103
C:\Users\user>cd gantest0103
C:\Users\user\gantest0103>npm install -g firebase-tools
changed 639 packages in 33s
65 packages are looking for funding
  run `npm fund` for details
C:\Users\user\gantest0103>firebase login
Already logged in as lindahui0204@gmail.com
C:\Users\user\gantest0103>
```

## 安裝完成

```
C:\Users\user>mkdir gantest0103
C:\Users\user>cd gantest0103
C:\Users\user\gantest0103>npm install -g firebase-tools
changed 639 packages in 33s
65 packages are looking for funding
  run `npm fund` for details
C:\Users\user\gantest0103>
```

```
C:\Users\user\gantest0103>firebase login
Already logged in as lindahui0204@gmail.com
```

```
C:\Users\user\gantest0103>firebase init
```

```
#####  #####  #####  #####  #####  #####  #####  #####
##      ##  ##      ##  ##      ##      ##  ##  ##  ##
#####  ##  #####  #####  #####  #####  #####  #####
##      ##  ##      ##  ##      ##      ##  ##  ##  ##
##      #####  ##      ##  #####  #####  ##      ##  #####
```

You're about to initialize a Firebase project in this directory:

```
C:\Users\user\gantest0103
```

```
? Are you ready to proceed? (Y/n) Y
```

鍵盤下鍵>>space鍵選取>>enter ➡

← Firebase login & firebase init

```
C:\Users\user\gantest0103
```

```
? Are you ready to proceed? Yes
```

```
? Which Firebase features do you want to set up for this directory? Press Space to select features, then Enter to confirm your choices. (Press <space> to select, <a> to toggle all, <i> to invert selection, and <enter> to proceed)
```

```
>( ) Realtime Database: Configure a security rules file for Real? Which Fire
```

```
? Which Firebase features do you want to set up for this directory? Press Space to select features, then Enter to confirm your choices. (Press <space> to select, <a> to toggle all, <i> to invert selection, and <enter> to proceed)
```

```
Database and (optionally) provision default instance
```

```
( ) Firestore: Configure security rules and indexes files for Firestore
```

```
( ) Functions: Configure a Cloud Functions directory and its files
```

```
>(*) Hosting: Configure files for Firebase Hosting and (optionally) set up GitHub Action deploys
```

```
( ) Hosting: Set up GitHub Action deploys
```

```
( ) Storage: Configure a security rules file for Cloud Storage
```

```
(Move up and down to reveal more choices)
```

## === Project Setup

First, let's associate this project directory with a Firebase project. You can create multiple project aliases by running `firebase use --add`, but for now we'll just set up a default project.

? Please select an option: (Use arrow keys)

> Use an existing project

Create a new project

Add Firebase to an existing Google Cloud Platform project

Don't set up a default project



Use an existing project

鍵盤下鍵>>enter



? Are you ready to proceed? Yes

? Which Firebase features do you want to set up for this directory? Press Space to select features, then Enter to confirm your choices. (Press <space> to select, <a> to toggle all, <i> to invert selection, and <enter> to proceed)

>( ) Realtime Database: Configure a security rules file for Real? Which Fire

? Which Firebase features do you want to set up for this directory? Press Space to select features, then Enter to confirm your choices. Hosting: Configure files for Firebase Hosting and (optionally) set up GitHub Action deploys

## === Project Setup

First, let's associate this project directory with a Firebase project. You can create multiple project aliases by running `firebase use --add`, but for now we'll just set up a default project.

? Please select an option: Use an existing project

? Select a default Firebase project for this directory: (Use arrow keys)

\* aiot2-b948f (AIOT2)

aiotdemo-f60d9 (AIOTdemo)

totoras-57d56 (totoras)

### === Project Setup

First, let's associate this project directory with a Firebase project. You can create multiple project aliases by running `firebase use --add`, but for now we'll just set up a default project.

? Please select an option: Use an existing project  
? Select a default Firebase project for this directory: aiot2-b948f (AIOT2)  
i Using project aiot2-b948f (AIOT2)

### === Hosting Setup

Your public directory is the folder (relative to your project directory) that will contain Hosting assets to be uploaded with `firebase deploy`. If you have a build process for your assets, use your build's output directory.

? What do you want to use as your public directory? public  
? Configure as a single-page app (rewrite all urls to /index.html)? No  
? Set up automatic builds and deploys with GitHub? No  
+ Wrote public/404.html  
+ Wrote public/index.html

i Writing configuration info to `firebase.json`...  
i Writing project information to `.firebaserc`...  
i Writing gitignore file to `.gitignore`...

+ Firebase initialization complete!

```
C:\Users\user\gantest0103>cd public
```


```
C:\Users\user\gantest0103\public>dir
```

磁碟區 C 中的磁碟是 OS  
磁碟區序號: 4E16-8A2E

C:\Users\user\gantest0103\public 的目錄

2024/01/03	下午 03:38	<DIR>	.
2024/01/03	下午 03:38	<DIR>	..
2024/01/03	下午 03:38		1,808 404.html
2024/01/03	下午 03:38		4,606 index.html
	2 個檔案		6,414 位元組
	2 個目錄		347,869,806,592 位元組可用

# Download

 [Product](#) [Solutions](#) [Open Source](#) [Pricing](#)

[Sign in](#) [Sign up](#)

[mwdchang / tfjs-gan](#) Public[Notifications](#) [Fork 19](#) [Star 60](#)[Code](#) [Issues 2](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#)

[master](#) [1 branch](#) [0 tags](#)

[Go to file](#) [Code](#)

 **mwdchang** update tensorflowjs version, reorg images

	images	update tensorflowjs vers
	README.md	update
	data.js	gan test
	gan.js	Update interface, increa
	image-util.js	gan test
	index.html	update tensorflowjs vers
	misc.js	update
	mnist_images.png	update

Local

Codespaces

[Clone](#)

HTTPS

GitHub CLI

Use Git or checkout with SVN using the web URL.

[Open with GitHub Desktop](#)

[Download ZIP](#)

6 years ago

## About

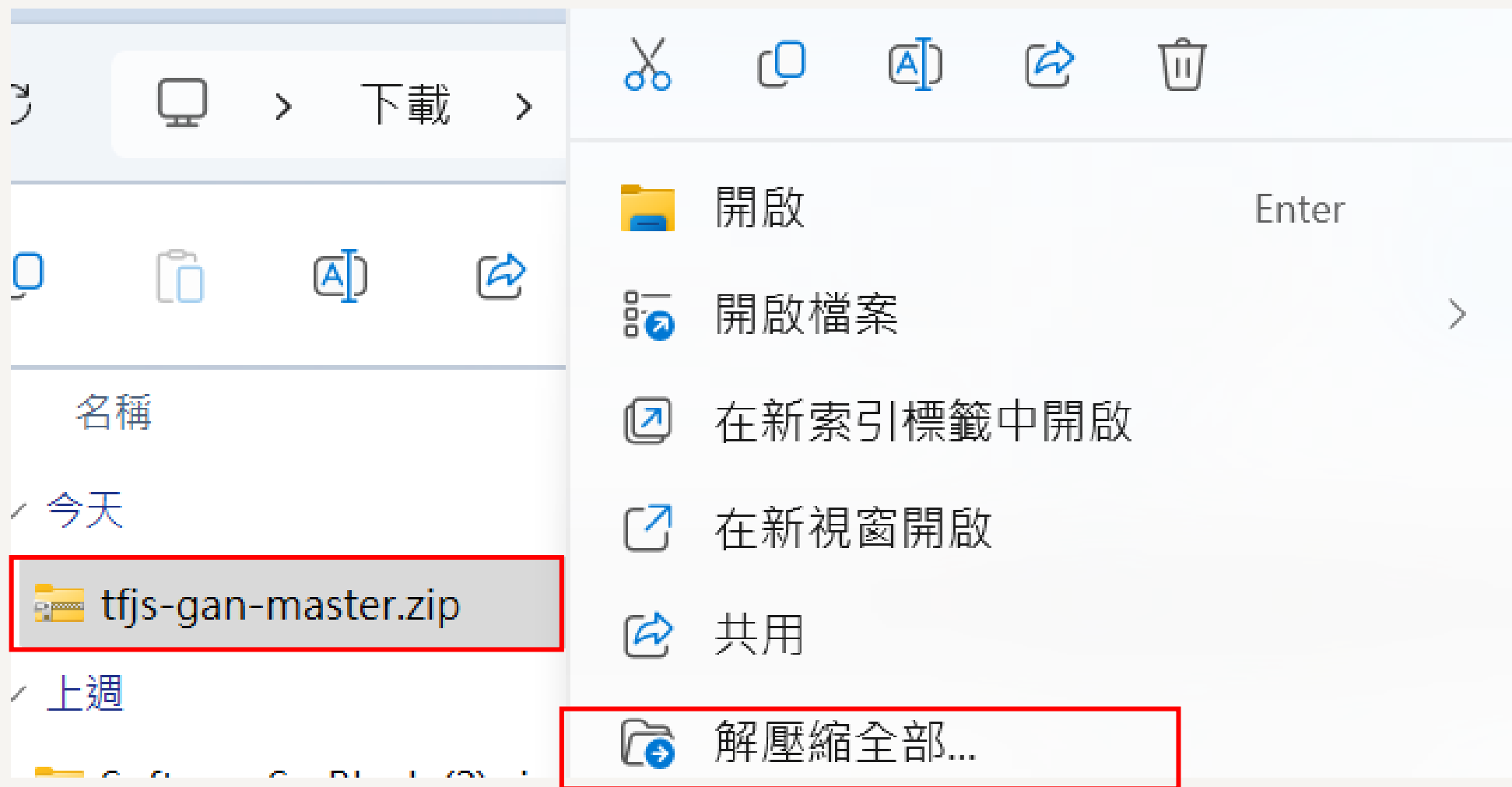
Simple GAN example using tensorflow JS core

[mwdchang.github.io/tfjs-gan/](#)

[neural-network](#) [generative-adversarial-network](#) [tensorflow-js](#)

[Readme](#) [Activity](#) [60 stars](#) [5 watching](#) [19 forks](#)

[Report repository](#)



# 至tfjs-gan-master目錄下複製全部

tfjs-gan-master

← → ↑ ↺ 下載 > tfjs-gan-master > tfjs-gan-master > 搜尋 tfjs-gan-master

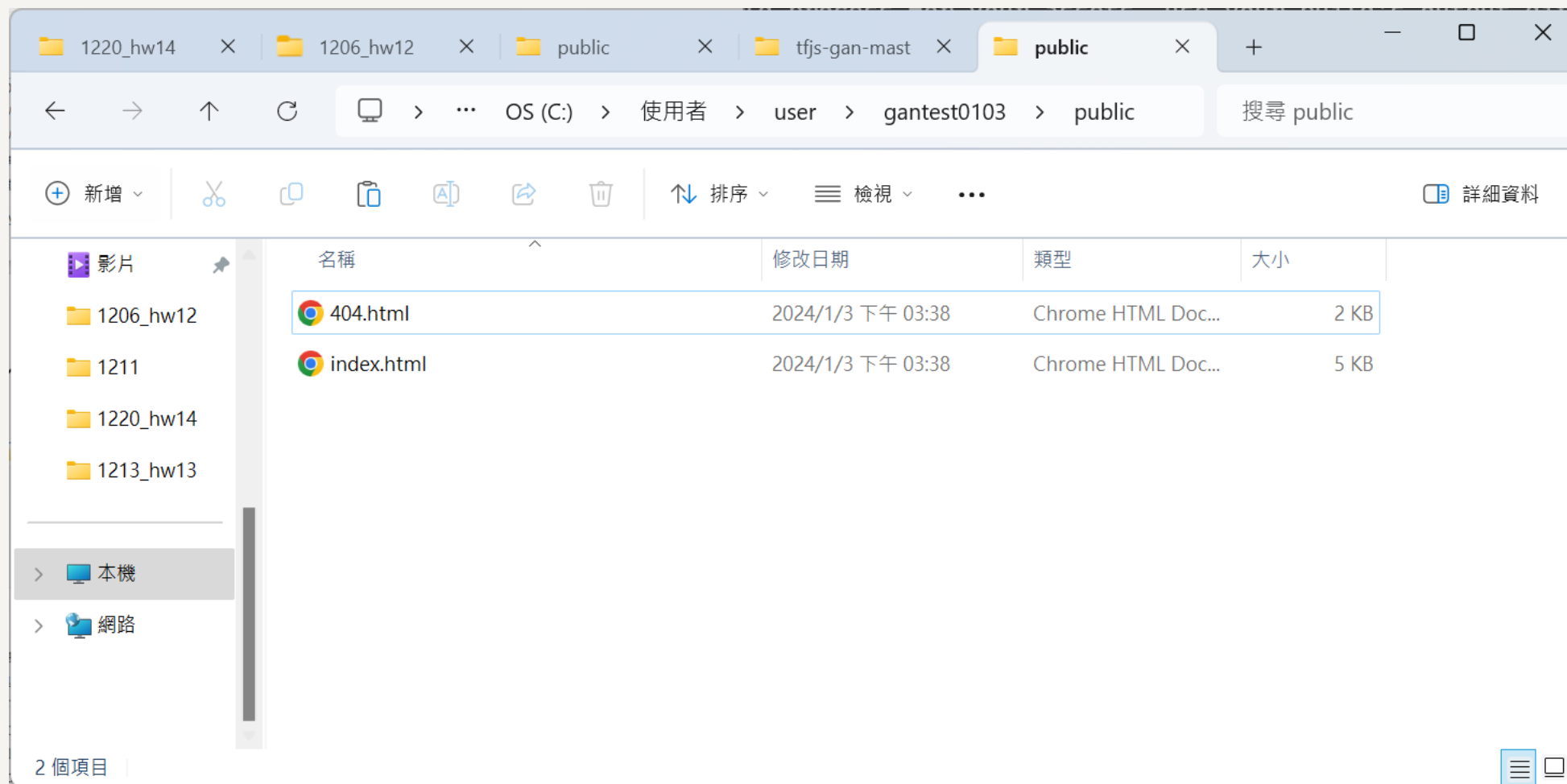
新增 剪貼 複製 移動 刪除 排序 檢視 詳細資料

名稱	修改日期	類型	大小
mnist_images.png	2024/1/3 下午 03:39	PNG 檔案	10,440 KB
mnist_labels_uint8	2024/1/3 下午 03:39	檔案	635 KB
styles.css	2024/1/3 下午 03:39	CSS 來源檔案	1 KB
weights.js	2024/1/3 下午 03:39	JavaScript 來源檔案	1,626 KB
data.js	2024/1/3 下午 03:39	JavaScript 來源檔案	6 KB
gan.js	2024/1/3 下午 03:39	JavaScript 來源檔案	4 KB
image-util.js	2024/1/3 下午 03:39	JavaScript 來源檔案	3 KB
index.html	2024/1/3 下午 03:39	Chrome HTML Doc...	5 KB
README.md	2024/1/3 下午 03:39	Markdown 來源檔案	1 KB
images	2024/1/3 下午 03:39	檔案資料夾	

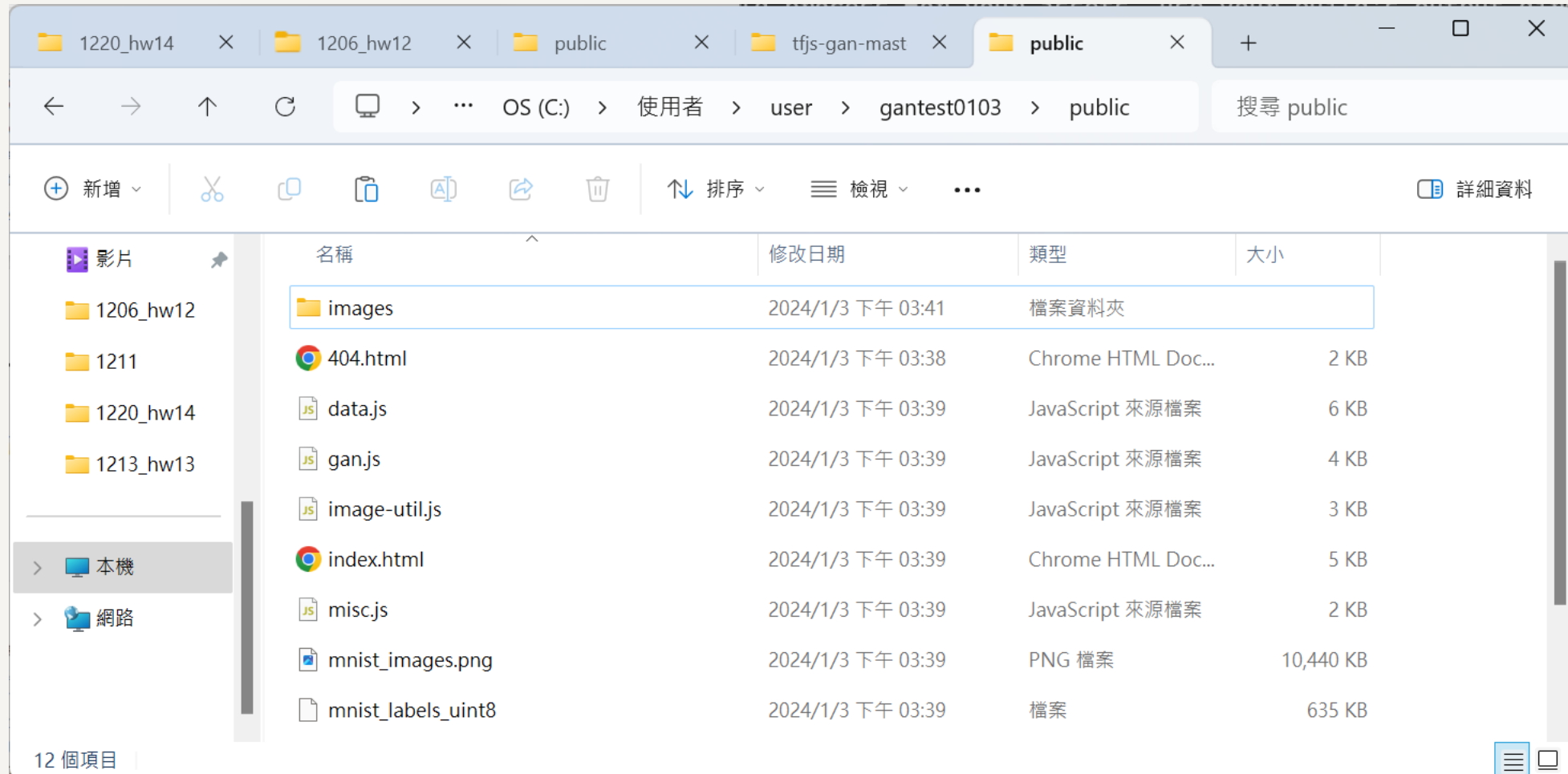
11 個項目 | 已選取 11 個項目



# 至gantest1029目錄下複製全部



# Result



# Firebase deploy & deploy complete

```
C:\Users\user\gantest0103\public>firebase deploy
```

```
=== Deploying to 'aiot2-b948f'...
```

```
i  deploying hosting
```

```
i  hosting[aiot2-b948f]: beginning deploy...
```

```
i  hosting[aiot2-b948f]: found 14 files in public
```

```
+  hosting[aiot2-b948f]: file upload complete
```

```
i  hosting[aiot2-b948f]: finalizing version...
```

```
+  hosting[aiot2-b948f]: version finalized
```

```
i  hosting[aiot2-b948f]: releasing new version...
```

```
+  hosting[aiot2-b948f]: release complete
```


```
+  Deploy complete!
```

```
Project Console: https://console.firebase.google.com/project/aiot2-b948f/overview
```

```
Hosting URL: https://aiot2-b948f.web.app
```

```
C:\Users\user\gantest0103\public>
```


### 目前版本



 lindahui0204@gmail.com  
 2024/1/3 下午3:42 (世界標準時間)

75f960

### 舊有版本



 lindahui0204@gmail.com  
 2024/1/3 下午3:42 (世界標準時間)

75f960
版本儲存空間設定
瞭解詳情 →

### 網域

[aiot2-b948f.web.app](#)


預設

[aiot2-b948f.firebaseio.com](#)


預設

新增自訂網域
查看全部 2 個網域 →

# 連至網站（完成布署）

## Simple MNIST GAN using TensorflowJS

Hand-written digit generation using Generative Adversarial Network. A simple TensorflowJS implementation that clocks in at less than 100 lines of code.

Early stages:



Getting better:



Later still:



Click **Train** to train for (an additional) 5) epochs. Click **Load weights** to restore pre-trained weights for the *Generator*. Click **Sample image** to generate a sample output using the current weights. The network should start to converge after 15-20 epochs.

Train

Load weights

Sample image

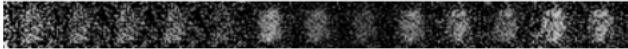
Fork me on GitHub

# Result

## Simple MNIST GAN using TensorflowJS

Hand-written digit generation using Generative Adversarial Network. A simple TensorflowJS implementation that clocks in at less than 100 lines of code.

Early stages:



Getting better:



Later still:



Click **Train** to train for (an additional) 5) epochs. Click **Load weights** to restore pre-trained weights for the *Generator*. Click **Sample image** to generate a sample output using the current weights. The network should start to converge after 15-20 epochs.

Train

Load weights

Sample image



Fork me on GitHub

Elements

Console

Sources

>>

2

top

Filter

2 hidden

Default levels

No Issues

► 6), byteLength: 4, byteOffset: 0, Length: 1, Symbol(Symbol.toStringTag): 'Float32Array']

i 1350

discriminator cost

► 6), byteLength: 4, byteOffset: 0, Length: 1, Symbol(Symbol.toStringTag): 'Float32Array']

generator cost

► 6), byteLength: 4, byteOffset: 0, Length: 1, Symbol(Symbol.toStringTag): 'Float32Array']

i 1374

discriminator cost

► 6), byteLength: 4, byteOffset: 0, Length: 1, Symbol(Symbol.toStringTag): 'Float32Array']

generator cost

done...

>

Console

What's New

Issues

Highlights from the Chrome 120 update

# Homework 15-1

- 比較傳統卷積運算與depthwise separable convolution參數量的比較。

## 傳統卷積

在傳統卷積中，每個輸入通道都有自己的卷積核（filter），並且這些卷積核在所有的輸入通道上共享。卷積層的輸出是將這些卷積核的產生的特徵圖相加。

如果有  $C_{\text{in}}$  個輸入通道，每個卷積核的大小是  $K \times K$ ，則傳統卷積的參數量（weights）為：

$$\text{參數量} = C_{\text{in}} \times C_{\text{out}} \times K \times K$$

其中  $C_{\text{out}}$  是卷積層的輸出通道數。



## 深度可分離卷積

深度可分離卷積分為兩個步驟：深度卷積和逐點卷積。首先進行深度卷積，對每個輸入通道進行單獨的卷積操作，然後進行逐點卷積，將深度卷積的輸出進行逐點卷積，產生最終的輸出。

如果有  $C_{in}$  個輸入通道，深度卷積的卷積核大小是  $K \times K$ ，逐點卷積的卷積核大小是  $1 \times 1$ ，輸出通道數是  $C_{out}$ ，則深度可分離卷積的參數量為：

$$\text{參數量} = C_{in} \times K \times K + C_{in} \times C_{out} \times 1 \times 1$$

這裡，深度卷積的部分通常比傳統卷積的參數量小，因為每個輸入通道有自己的卷積核。

## 結論

深度可分離卷積通常具有比傳統卷積更少的參數，這使得它在模型中使用時能夠減少計算成本並加速訓練，同時在許多應用中仍能保持相當好的性能。

高效卷積計算結構 - Depthwise Separable Convolution

<https://blog.yeshuanova.com/2018/02/depthwise-separable-convolution/>