

github程式碼存放點

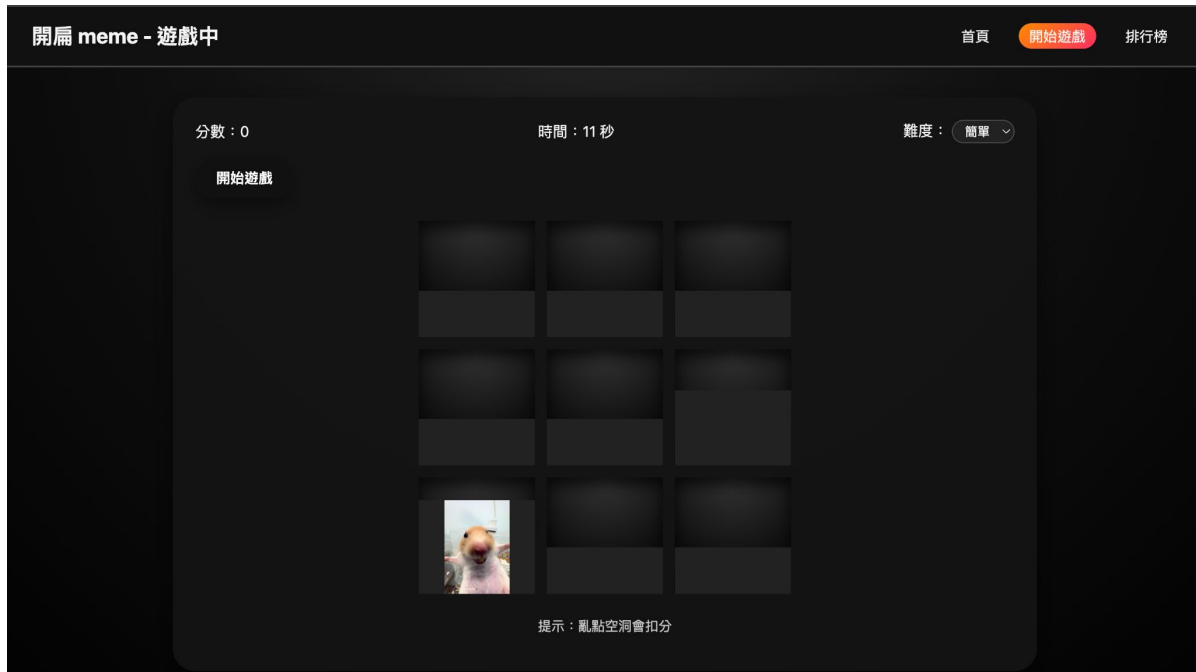
<https://github.com/Nina-cmyk/whack-a-meme-midterm/tree/main>

遊戲核心檔案_game.html

用途：
遊戲頁面（開始遊戲）

內容包含：

- 1、顯示遊戲的 3×3 網格
- 2、顯示分數、時間
- 3、難度選擇器
- 4、開始遊戲的按鈕
- 5、載入 game.js 讓遊戲動起來



1. 文件宣告與標頭區(<head>)

<!DOCTYPE html>: 宣告使用 HTML5 標準

lang="zh-Hant": 告訴瀏覽器此頁面是繁體中文

charset="UTF-8": 設定網頁文字編碼(避免亂碼)

viewport: 確保手機裝置的排版縮放正確

title: 瀏覽器標籤名稱

styles.css: 引入 CSS 樣式(介面、遊戲洞、按鈕...)

這段基本上定義了網頁的語言、樣式與基本資訊。

```
1    <!DOCTYPE html>
2    <html lang="zh-Hant">
3    <head>
4        <meta charset="UTF-8" />
5        <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6        <title>開扁 meme - 遊戲中</title>
7        <link rel="stylesheet" href="styles.css" />
8    </head>
```

2.<header>、<main>、分數、時間、難度、開始遊戲按鈕

<header>

顯示頁面標題: 開扁 meme - 遊戲中

導覽列(首頁／開始遊戲／排行榜)

class="active" 代表當前頁面(會用 CSS 高亮)

<main>

game-panel 是整個遊戲區塊的容器

分數__HTML 負責: 放一個可被抓到的容器放分數、JS

負責: 改這個容器裡的數字

倒數時間__30 是預設時間(30 秒)

難度調整__<select> 代表一個「下拉選單」有

id="difficulty", 給 JS 用的

JS 使用 value 來控制速度

難度 value 意思

簡單 1000 每 1000ms 換一次 → 1 秒一隻

普通 700 每 0.7 秒換一隻 → 比較快

困難 450 每 0.45 秒換一隻 → 超級快

button_按一下 → 呼叫 game.js 的 startGame()

```
9      <body>
10      <header class="site-header">
11      <h1>開扁 meme - 遊戲中</h1>
12      <nav>
13      <a href="index.html">首頁</a>
14      <a href="game.html" class="active">開始遊戲</a>
15      <a href="ranking.html">排行榜</a>
16      </nav>
17      </header>
18
19      <main class="container">
20      <section class="game-panel">
21      <div class="game-info">
22      <div>分數: <span id="score">0</span></div>
23      <div>時間: <span id="time">30</span> 秒</div>
24      <div>
25      難度:
26      <select id="difficulty">
27      <option value="1000">簡單</option>
28      <option value="700">普通</option>
29      <option value="450">困難</option>
30      </select>
31      </div>
32      </div>
33
34      <button id="start-btn" class="btn btn-primary">開始遊戲</button>
```

3. 3×3 遊戲網格、提示文字、頁尾 Footer、載入遊戲邏輯 game.js

hole: 每一格固定是正方形, CSS 做成像洞穴的樣子(陰影、黑色漸層)

meme: JS 用 background-image 方式塞進梗圖(原本位置在洞底部, 加上 .active class 會向上冒出來(由 CSS 控制))

tip_告訴玩家遊戲規則, CSS 控制淡色小提示字

footer_統一網站風格, 放版權或頁腳文字

<script src="game.js"></script> __載入
JavaScript, 讓這個頁面變成好玩的遊戲
內含: 分數計算、計時器、隨機迷因出現邏輯、事件
監聽(點洞判斷加分／扣分)、排行榜寫入
localStorage
* 沒有這支 JS → 網格會出現但無法玩遊戲。

```
36     <div class="game-grid">
37         <!-- 3x3 網格 -->
38         <div class="hole"><div class="meme"></div></div>
39         <div class="hole"><div class="meme"></div></div>
40         <div class="hole"><div class="meme"></div></div>
41         <div class="hole"><div class="meme"></div></div>
42         <div class="hole"><div class="meme"></div></div>
43         <div class="hole"><div class="meme"></div></div>
44         <div class="hole"><div class="meme"></div></div>
45         <div class="hole"><div class="meme"></div></div>
46         <div class="hole"><div class="meme"></div></div>
47     </div>
48
49     <p class="tip">提示：亂點空洞會扣分</p>
50 </section>
51 </main>
52
53 <footer class="site-footer">
54     <p>開扁 meme Whack-a-Meme</p>
55 </footer>
56
57 <script src="game.js"></script>
58 </body>
59 </html>
```

網站主頁_index.html

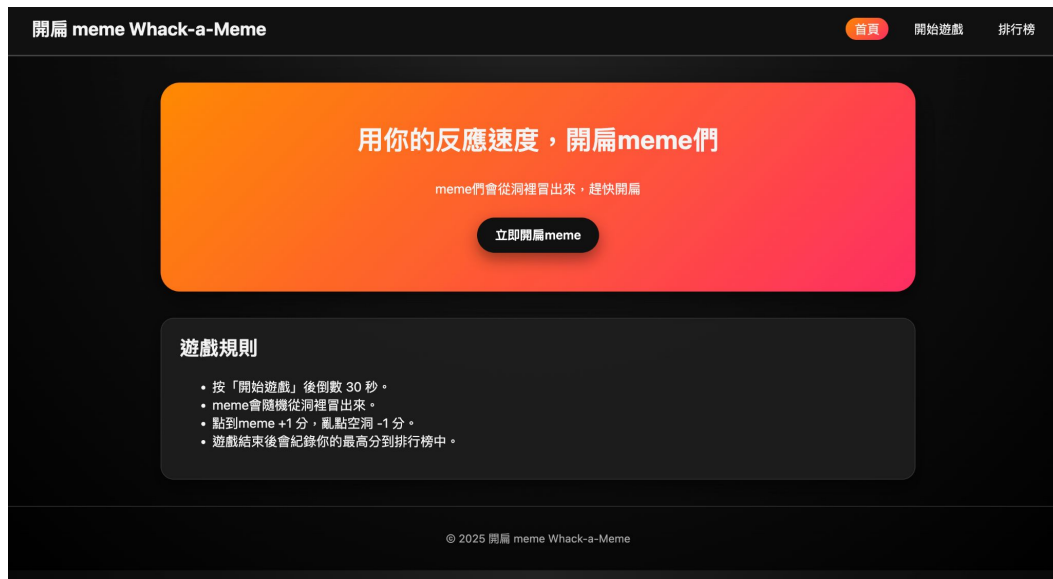
用途：

網站首頁(功能介紹與遊戲規則)

內容包含：

- 1、網站標題
- 2、介紹文字
- 3、顯示遊戲規則
- 4、讓玩家能按按鈕開始遊戲(連到game.html)
- 5、共同 header、footer

這一頁不含遊戲邏輯, 主要負責「介紹 + 導覽」



1、<head>跟前面功能差不多一樣

2、Header: 網站大標題 + 導覽列__顯示網站標題、網站導覽列(切換頁面)

3、Hero 區塊(首頁主視覺)__清楚告訴玩家遊戲主題、提供醒目的 CTA(Call to Action)按鈕 → 開始遊戲

4、遊戲規則區(card)__告知玩家遊戲如何玩、列出重點規則

class="card" → 讓內容看起來像一張卡片
margin-top: 2rem 只是讓它跟 hero 分開一點

5、頁腳 Footer__顯示簡單版權資訊、網站底部結尾, 和 header 一樣, 是三頁共用的固定格式

```
1  <!DOCTYPE html>
2  <html lang="zh-Hant">
3  <head>
4    <meta charset="UTF-8" />
5    <title>開扁 meme Whack-a-Meme</title>
6    <link rel="stylesheet" href="styles.css" />
7  </head>
8  <body>
9    <header class="site-header">
10     <h1>開扁 meme Whack-a-Meme</h1>
11     <nav>
12       <a href="index.html" class="active">首頁</a>
13       <a href="game.html">開始遊戲</a>
14       <a href="ranking.html">排行榜</a>
15     </nav>
16   </header>
17
18   <main class="container">
19     <section class="hero">
20       <h2>用你的反應速度，開扁meme們</h2>
21       <p>meme們會從洞裡冒出來，趕快開扁</p>
22       <a href="game.html" class="btn btn-primary">立即開扁meme</a>
23     </section>
24
25     <section class="card" style="margin-top: 2rem;">
26       <h2>遊戲規則</h2>
27       <ul>
28         <li>按「開始遊戲」後倒數 30 秒。</li>
29         <li>meme會隨機從洞裡冒出來。</li>
30         <li>點到meme +1 分，亂點空洞 -1 分。</li>
31         <li>遊戲結束後會紀錄你的最高分到排行榜中。</li>
32       </ul>
33     </section>
34   </main>
35
36   <footer class="site-footer">
37     <p>© 2025 開扁 meme Whack-a-Meme</p>
38   </footer>
39 </body>
40 </html>
```

排行榜頁面_ranking.html

用途：

顯示玩家的「前五名最高分」排行榜

內容包含：

- 1、動態表格(JS 填資料)
- 2、使用 localStorage 記錄
- 3、顯示玩家名稱與分數

排行榜

[首頁](#)[開始遊戲](#)[排行榜](#)

最高分（前五名）

排名	玩家名稱	分數
1	Player	26
2	00	14
3	Player	0

1、<head>跟前面功能差不多一樣
2、Header 跟前面功能差不多一樣
3、主內容區(排行榜卡片)__<main
class="container">/<section class="card">/<h2>
最高分(前五名)</h2>

4、排行表格(table)__此表格由三部分組成:

排名 第 1~5 名

玩家名稱 玩家輸入的名字

分數 玩家該局分數

5、排行榜資料顯示的位置一開始是空的

JavaScript 會把 <tr> 一行一行塞進來

* <tbody id="leaderboard-body">是排行榜資料
的顯示區塊, 內容完全由 JavaScript 根據
localStorage 的分數自動生成

```
1  <!DOCTYPE html>
2  <html lang="zh-Hant">
3  <head>
4    <meta charset="UTF-8" />
5    <title>排行榜 - 開扁 meme</title>
6    <link rel="stylesheet" href="styles.css" />
7  </head>
8  <body>
9    <header class="site-header">
10     <h1>排行榜</h1>
11     <nav>
12       <a href="index.html">首頁</a>
13       <a href="game.html">開始遊戲</a>
14       <a href="ranking.html" class="active">排行榜</a>
15     </nav>
16   </header>
17
18   <main class="container">
19     <section class="card">
20       <h2>最高分 (前五名)</h2>
21
22       <table class="score-table">
23         <thead>
24           <tr>
25             <th>排名</th>
26             <th>玩家名稱</th>
27             <th>分數</th>
28           </tr>
29         </thead>
30         <tbody id="leaderboard-body">
31           <!-- JS 會自動填入 -->
32         </tbody>
33       </table>
34     </section>
35   </main>
```

1、localStorage.getItem("memeLeaderboard")

→ 從瀏覽器本機儲存抓取排行榜資料

→ 資料格式為 JSON 字串, || "[]" 如果根本沒有排行榜__使用空陣列 "[]" 避免報錯

2、#leaderboard-body 就是 <tbody> 裡的區域

, JS會把 <tr> 插入這個區域, 代表排行榜資料會寫到這裡

3、如果排行榜陣列是空的, 顯示一行尚未遊玩
colspan="3" 表示這一格跨三欄(排名 / 名稱 / 分數)

4、有資料時 → 顯示前五名成績

.map(...): 把陣列每一筆資料變成 <tr>...</tr>

entry : 單筆排行榜資料, 例如 {name:"Jacky",
score: 20}

index : 排名編號(0 開始)

\${index + 1} : 排名 1~5 名

\${entry.name} : 玩家名稱

\${entry.score} : 分數

.join(""): 把所有 <tr> 組成一段完整 HTML

```
37 <script>
38   // 讀取排行榜資料
39   const leaderboard =
40     JSON.parse(localStorage.getItem("memeLeaderboard") || "[]");
41
42   const tbody = document.getElementById("leaderboard-body");
43
44   // 若無資料
45   if (leaderboard.length === 0) {
46     tbody.innerHTML = `
47       <tr>
48         <td colspan="3">尚未遊玩</td>
49       </tr>
50     `;
51   } else {
52     // 顯示前五筆
53     tbody.innerHTML = leaderboard
54       .map(
55         (entry, index) => `
56           <tr>
57             <td>${index + 1}</td>
58             <td>${entry.name}</td>
59             <td>${entry.score}</td>
60           </tr>
61         `
62       )
63       .join("");
64   }
65 </script>
66 </body>
67 </html>
```

網站樣式_styles.css

用途：

所有頁面的共同設計風格

內容包含：

- 1、背景顏色、漸層、字體
- 2、按鈕樣式
- 3、卡片(card)外框、陰影
- 4、遊戲洞穴(hole)美術
- 5、meme圖片顯示大小
- 6、排行榜表格樣式

1. 全域設定__box-sizing: border-box: 讓寬高計算更直覺

, 不會因為 padding/ border 改變尺寸

設定全站字型: Noto Sans TC (中文好看)

設定 深色漸層背景: 上亮下暗, 營造遊戲舞台感

讓 <a> 跟文字風格一致, 不突兀

2. 共同框架: Header / Container / Footer

深色背景、利用 flex 排版標題 + 導覽列

active 連結使用亮色漸層 → 呈現「當前頁面」

/限制最大寬度 960px、居中呈現

保持網頁內容一致的排版美觀

/淺灰文字、上方有分隔線、提供乾淨的頁尾

```
50 .container {
51   max-width: 960px;
52   margin: 2rem auto;
53   padding: 0 1rem;
54 }
55
56 .site-footer {
57   text-align: center;
58   padding: 1rem 0;
59   font-size: 0.85rem;
60   color: #aaa;
61   border-top: 1px solid #333;
62   margin-top: 2rem;
63 }
```

```
1  * {
2    box-sizing: border-box;
3  }
4
5  body {
6    margin: 0;
7    font-family: "Noto Sans TC", system-ui, -apple-system, BlinkMacSystemFont, sans-serif;
8    background: radial-gradient(circle at top, #222 0, #111 40%, #000 100%);
9    color: #f5f5f5;
10  }
11
12  a {
13    color: inherit;
14    text-decoration: none;
15  }
16
17  /* 共同框架 */
18
19  .site-header {
20    background: #111;
21    border-bottom: 2px solid #444;
22    padding: 1rem 2rem;
23    display: flex;
24    justify-content: space-between;
25    align-items: center;
26  }
27
28  .site-header h1 {
29    font-size: 1.4rem;
30    margin: 0;
31  }
32
33  .site-header nav a {
34    margin-left: 1rem;
35    padding: 0.3rem 0.7rem;
36    border-radius: 999px;
37    font-size: 0.95rem;
38    transition: background 0.2s, transform 0.1s;
39  }
40
41  .site-header nav a:hover {
42    background: #333;
43    transform: translateY(-1px);
44  }
45
46  .site-header nav a.active {
47    background: linear-gradient(135deg, #ff8a00, #ff2e63);
48  }
```

1、首頁 Hero 區塊__Hero 是讓首頁變醒目，使用亮色漸層(橘→粉紅)、大圓角+強陰影

2、按鈕樣式__圓角 999px → 膠囊按鈕
鼠標移過會「浮起來」: transform: translateY(-1px)、深色主按鈕

```
65  /* 首頁 hero */
66
67  .hero {
68    text-align: center;
69    padding: 3rem 1rem;
70    background: linear-gradient(135deg, #ff8a00, #ff2e63);
71    border-radius: 1.5rem;
72    box-shadow: 0 12px 35px rgba(0, 0, 0, 0.5);
73  }
74
75  .hero h2 {
76    margin-top: 0;
77    font-size: 2rem;
78  }
79
80  .hero p {
81    max-width: 600px;
82    margin: 0.5rem auto 1.5rem;
83  }
84
85  /* 按鈕 */
86
87  .btn {
88    display: inline-block;
89    padding: 0.7rem 1.4rem;
90    border-radius: 999px;
91    border: none;
92    cursor: pointer;
93    font-weight: 600;
94    font-size: 0.95rem;
95    transition: transform 0.1s, box-shadow 0.1s, background 0.2s;
96  }
97
98  .btn-primary {
99    background: #111;
100    color: #fff;
101    box-shadow: 0 8px 18px rgba(0, 0, 0, 0.4);
102  }
103
104  .btn-primary:hover {
105    transform: translateY(-1px);
106    box-shadow: 0 10px 20px rgba(0, 0, 0, 0.6);
107    background: #000;
108  }
```

卡片:規則說明卡片、排行榜卡片、遊戲框

.game-panel遊戲容器(背景、邊界)

.game-info 分數 + 時間 + 難度項目

.game-grid 3×3 九宮格

.hole 每個洞

.meme meme 圖片(會冒出來)

.meme.active meme 往上冒的動畫

```
110 /* 卡片 */
111
112 .card {
113   background: rgba(30, 30, 30, 0.9);
114   border-radius: 1rem;
115   padding: 1.2rem 1.4rem;
116   box-shadow: 0 8px 20px rgba(0, 0, 0, 0.5);
117   border: 1px solid #333;
118 }
119
120 .card h2,
121 .card h3 {
122   margin-top: 0;
123 }
124
125 /* 遊戲區塊 */
126
127 .game-panel {
128   background: rgba(20, 20, 20, 0.95);
129   border-radius: 1.5rem;
130   padding: 1.5rem;
131   box-shadow: 0 12px 30px rgba(0, 0, 0, 0.5);
132 }
133
134 .game-info {
135   display: flex;
136   flex-wrap: wrap;
137   gap: 1rem;
138   justify-content: space-between;
139   align-items: center;
140   margin-bottom: 1rem;
141 }
142
143 .game-info div {
144   font-size: 1rem;
145 }
146
147 #difficulty {
148   background: #222;
149   border-radius: 999px;
150   border: 1px solid #555;
151   color: #fff;
152   padding: 0.2rem 0.6rem;
153 }
154
155 .game-grid {
156   margin: 1.5rem auto;
157   display: grid;
158   grid-template-columns: repeat(3, 1fr);
159   gap: 0.8rem;
160   max-width: 400px;
161 }
162
163 .hole {
164   position: relative;
165   width: 100%;
166   padding-top: 100%; /* 正方形 */
167   background: radial-gradient(circle at top, #333 0, #111 60%, #000 100%);
168   overflow: hidden;
169   box-shadow: inset 0 8px 15px rgba(0, 0, 0, 0.7);
170   cursor: pointer;
171 }
172
173 .hole:hover {
174   box-shadow: inset 0 10px 20px rgba(0, 0, 0, 0.9);
175 }
176
177 .meme {
178   position: absolute;
179   left: 50%;
180   bottom: -60%;
181   transform: translateX(-50%);
182   width: 100%;
183   height: 100%;
184   background-size: contain; /* 完整顯示 GIF/圖片 */
185   background-repeat: no-repeat;
186   background-position: center;
187   background-color: #222;
188   transition: bottom 0.15s ease-out;
189 }
190
191 .meme.active {
192   bottom: 5%;
193 }
```

1、提示文字

簡單的小提示：視覺淡、但可讀性強

2、排行表格

`.score-table { ... }`

`.score-table th, .score-table td { ... }`

功能：

讓排行榜更整齊、可讀性高

`border-collapse: collapse` → 邊框合併，看起來不會有空隙

顏色深、文字亮

```
195  /* 提示文字 */
196
197  .tip {
198      text-align: center;
199      font-size: 0.9rem;
200      color: #ccc;
201  }
202
203  /* 排行表 */
204
205  .score-table {
206      width: 100%;
207      border-collapse: collapse;
208      margin-top: 1rem;
209  }
210
211  .score-table th,
212  .score-table td {
213      border: 1px solid #444;
214      padding: 0.6rem;
215      text-align: center;
216  }
217
218  .score-table th {
219      background: #222;
220  }
```

game.js

game.js 是整個 Whack-a-Meme 小遊戲的邏輯程式

負責：

控制迷因什麼時候出現、出在哪個洞

計算分數、時間倒數

根據選擇的難度調整速度

遊戲結束後，把分數寫進排行榜（localStorage）

HTML 是「骨架」

CSS 是「外觀」

game.js 就是「行為（互動）」

1、梗圖列表(遊戲素材)__這是一個放遊戲中 meme 圖片的陣列

2、DOMContentLoaded – 等 HTML 載入後才執行, 確保所有 HTML 元素存在後, JS 才開始抓元素、綁事件

3、抓取 DOM 元素:

.hole → 9 個洞

.meme → 9 個對應的 meme 元素

score / time / start-btn → 遊戲資訊

difficulty → 難度(速度)

4、遊戲狀態變數__紀錄分數、倒數計時、計時器 ID、遊戲是否正在進行

5、小工具:隨機取得陣列元素

6、resetMemes() – 收回所有 meme

7、showRandomMeme() – 隨機讓一個洞冒出 meme

```
1 // 梗圖列表
2 const MEME_IMAGES = [
3   "eatingcat.jpg", // 你好 我吃一點
4   "laolao.jpg",    // 老師 葉菜 撈撈
5   "fakerx2.jpg",   // Faker 們
6   "studyingdog.jpg", // 當你廢了兩三個月卻突然要你讀書
7   "dancingcat.gif", // 跳舞貓
8   "stresscat.gif",  // AAAAA STRESSS
9   "bananacat.gif", // 香蕉貓
10  "oiaoiacat.gif",  // OIAOIIA 貓
11  "laughingmouse.jpeg", // 鼠笑
12  "germanecat.gif"  // 德國貓
13 ];
14
15 document.addEventListener("DOMContentLoaded", () => {
16   const holes = document.querySelectorAll(".hole");
17   if (!holes.length) return;
18
19   const memes = document.querySelectorAll(".meme");
20   const scoreEl = document.getElementById("score");
21   const timeEl = document.getElementById("time");
22   const startBtn = document.getElementById("start-btn");
23   const difficultySelect = document.getElementById("difficulty");
24
25   let score = 0;
26   let timeLeft = 30;
27   let gameTimer = null;
28   let memeTimer = null;
29   let gameRunning = false;
30
31   const randomItem = list =>
32     list[Math.floor(Math.random() * list.length)];
33
34   function resetMemes() {
35     memes.forEach(m => {
36       m.classList.remove("active");
37       m.style.backgroundImage = "";
38     });
39   }
40
41   function showRandomMeme() {
42     resetMemes();
43     const hole = randomItem(holes);
44     const meme = hole.querySelector(".meme");
45     meme.style.backgroundImage = `url("${randomItem(MEME_IMAGES)}")`;
46     meme.classList.add("active");
47   }
```

1、startGame() – 開始遊戲：重設分數 & 時間、禁用開始按鈕、根據難度設定 meme 出現頻率、每隔 speed 毫秒出現一個新 meme、每秒倒數時間、時間到呼叫 endGame()

2、endGame() – 結束遊戲 + 儲存排行榜：
讀取排行榜

localStorage.getItem("memeLeaderboard")

加入當前分數 leaderboard.push(...)

排序(最高分在前)

sort((a,b)=>b.score-a.score)

只保留前五高分 slice(0,5)

存回 localStorage setItem(...)

```
49  function startGame() {
50      if (gameRunning) return;
51      gameRunning = true;
52
53      score = 0;
54      timeLeft = 30;
55      scoreEl.textContent = score;
56      timeEl.textContent = timeLeft;
57      startBtn.disabled = true;
58
59      const speed = Number(difficultySelect.value) || 800;
60
61      showRandomMeme();
62      memeTimer = setInterval(showRandomMeme, speed);
63
64      gameTimer = setInterval(() => {
65          timeLeft -= 1;
66          timeEl.textContent = timeLeft;
67
68          if (timeLeft <= 0) endGame();
69      }, 1000);
70  }
71
72  function endGame() {
73      clearInterval(gameTimer);
74      clearInterval(memeTimer);
75      gameRunning = false;
76      startBtn.disabled = false;
77
78      resetMemes();
79
80      alert(`時間到！你的分數是：${score}`);
81
82      const playerName =
83          prompt("輸入你的名字（將記錄在排行榜）:", "Player") || "Player";
84
85      // ---- Top5 排行榜整合 ----
86      const leaderboard =
87          JSON.parse(localStorage.getItem("memeLeaderboard") || "[]");
88
89      leaderboard.push({ name: playerName, score });
90
91      leaderboard.sort((a, b) => b.score - a.score);
92
93      const top5 = leaderboard.slice(0, 5);
94
95      localStorage.setItem("memeLeaderboard", JSON.stringify(top5));
96  }
```

1、玩家點擊洞穴事件：

如果點到正在冒出的 meme → +1 分

如果點到空洞 → -1 分

點到成功後，把 meme 收回（避免重複得分）

2、綁定開始遊戲按鈕：

玩家按下按鈕 → 開始遊戲

總結JS：

這段 JavaScript 控制整個 Whack-a-Meme 的遊戲流程，包括隨機出現迷因、計時、計分、難度調整以及 localStorage 的前五名排行榜，使遊戲能完整運作。

```
98     alert("本次分數已加入排行榜！");
99   }
100
101   holes.forEach(hole => {
102     hole.addEventListener("click", () => {
103       if (!gameRunning) return;
104
105       const meme = hole.querySelector(".meme");
106
107       if (meme.classList.contains("active")) {
108         score += 1;
109         meme.classList.remove("active");
110         meme.style.backgroundImage = "";
111       } else {
112         score -= 1;
113       }
114
115       scoreEl.textContent = score;
116     });
117   });
118
119   startBtn.addEventListener("click", startGame);
120 });
```

遊戲迷因素材

用途：

遊戲內會隨機出現的迷因圖片／GIF

在 `game.js` 裡會從 MEME_IMAGES 陣列中隨機選取

包含 GIF + JPG

