

(100%) JavaScript 網頁設計 1N 期末上機 考-2 -- 開放考試

2024-6-13 at E201

Note:

1. 請不要發揮同學愛，作弊雙方除了本次考試 0 分外，平常分數另扣 20 分，情節嚴重者會送校。
2. iClass 上請繳交 md_xx.zip，還有 f2_xx.zip 2 個檔
3. 請直接將答案寫在 f2_xx.md 上，老師實作的圖片放在 f2_htc.pdf 上，請依照老師所給的圖片來實作並標註
4. 請用老師提供的 f2_xx code 來實作 P2, P3, P4, P5 等 4 題
5. 跟期中考相關的檔案及目錄名稱有 xx 時，必須要改成學號後 2 碼，沒有修改時，會視違犯情況扣分。跟期中考題無關的檔案及目錄名稱有 xx，則不必修改。
6. 每一張圖片要有機房左側背景，圖片上要有你的學號(或後兩碼)，圖片標註要跟老師所標註的類似。違者會依情節扣分。
7. 有標註要 git log 的題目，必須要做，這是我們上課練習的一部分。
8. 請自評分數，將每一題的？填入分數，沒有填者，不會批改，以 0 分計算。

Your (Name, ID): (?, ?)

f2 開放網路題目 (50%)

- P1-1 (10%): ? 分
- P1-2 (10%): ? 分
- P1-3 (10%): ? 分
- P1-4 (10%): ? 分
- P2 (20%): ? 分
- P3 (15%): ? 分
- P4 (15%): ? 分
- P5 (10%): ? 分

總分: ? 分

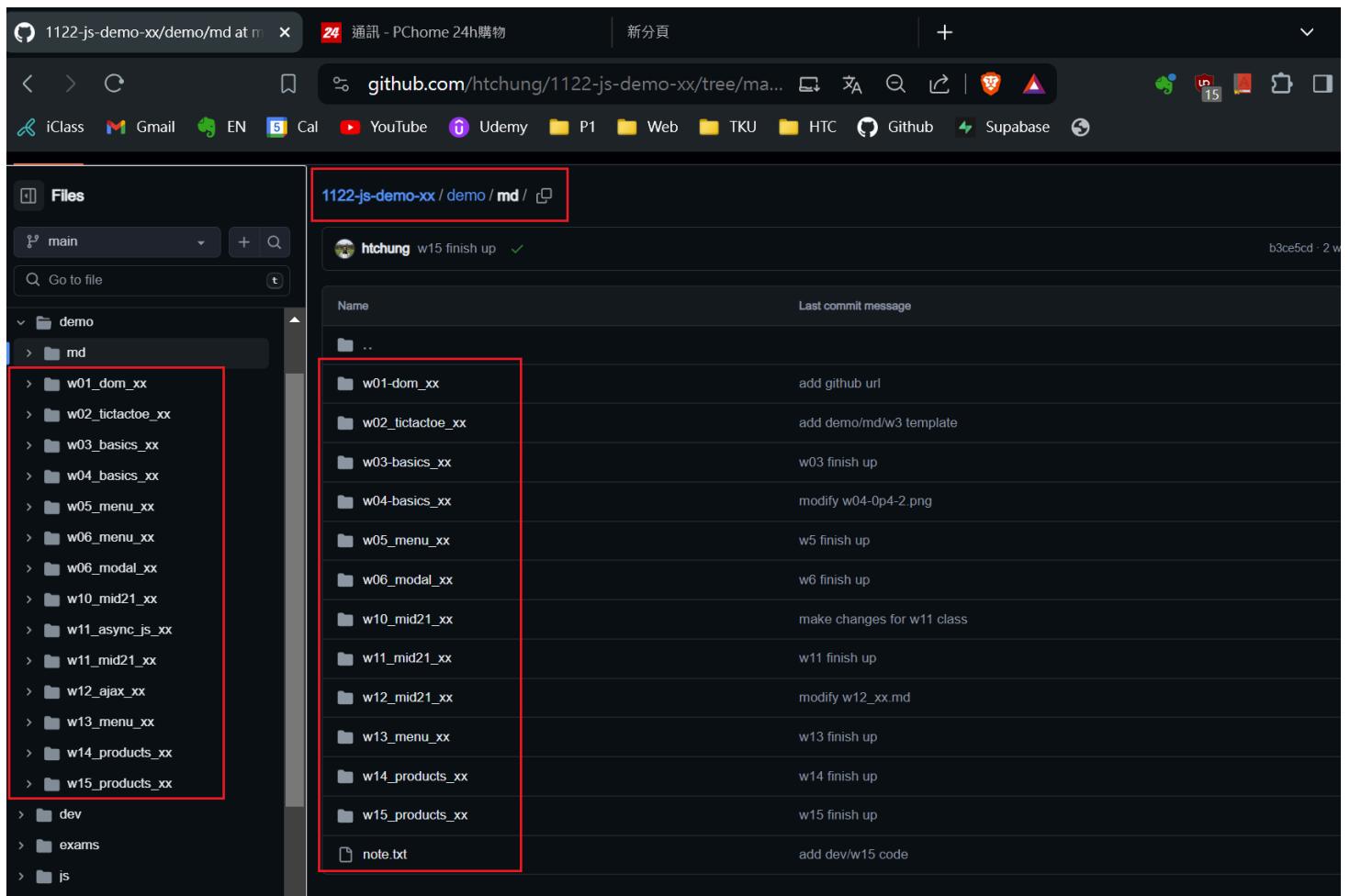
(40%) P1: Show Info of Your Github Repo, Client Supa Github, Vercel as follows

(10%) P1-1: Show Your Github Repo for all md and demo

=> My Github repo URL

<https://github.com/htchung/1122-js-demo-xx/>

=> 下圖左邊框顯示所有周次的 demo，右邊框顯示所有周次的 md



Your Answer

=> My Github repo URL

[My Github repo URL for demo](#)

=> 下圖左邊框顯示所有的 demo，右邊框顯示所有的 md



=> git log

(10%) P1-2: Show all your git logs from W10 to W15

=> My Github URL for git logs · 標註本學期 git log 次數

My Github URL for git logs

The screenshot shows a GitHub repository page for 'htchung / 1122-js-demo-xx'. The repository is private. The commit history lists 98 commits. A red box highlights the '98 Commits' link in the top right corner of the commit list. The commits are organized into several branches and folders, with descriptions and dates for each commit.

Commit	Description	Date
ff8699a	htchung add exams/final1 for Final-1 exam	last week
	add megamenu for class demo	3 months ago
	add megamenu for class demo	3 months ago
	w15 finish up	2 weeks ago
	add exams/final1 for Final-1 exam	last week
	add exams/final1 for Final-1 exam	last week
	W15-P4: Put w15-demo into project-demo, and make it run ...	2 weeks ago
	W15-P4: Put w15-demo into project-demo, and make it run ...	2 weeks ago
	gitlog.txt	2 months ago
	index.html	3 weeks ago
	megamenu_xx.html	3 months ago

=> git log for w15 and w14

The screenshot shows a GitHub commit history for the repository `htchung/1122-js-demo-xx`. The commits are organized into two main sections: W15 and W14.

W15 Commits:

- W15-P4: Put w15-demo into project-demo, and make it run correctly**
hitchung committed 2 weeks ago · 1 / 1
- W15-P3: Implement company filter functions, test for each company using direct filtering of the products containing company data**
hitchung committed 2 weeks ago · 1 / 1
- W15-P2: Implement company filter functions, test for each company**
hitchung committed 2 weeks ago · 1 / 1
- W15-P1: Create company (filter) menu using scss**
hitchung committed 2 weeks ago · 1 / 1

W15 Sub-tasks:

- add md/w15 template**
hitchung committed 2 weeks ago · 1 / 1
- add dev/w15 code**
hitchung committed 2 weeks ago · 1 / 1

Commits on May 23, 2024:

w14 finish up
hitchung committed 3 weeks ago · 1 / 1

W14 Commits:

- 14-P4: Do products_supabase_xx.js to get data from Supabase**
hitchung committed 3 weeks ago · 1 / 1
- W14-P3: Use SQL to create schemas and data for company_xx and products_xx**
hitchung committed 3 weeks ago · 1 / 1
- W14-P2: Do products_localJson_xx.js to get local json data (/api/productsData.json)**
hitchung committed 3 weeks ago · 1 / 1
- W14-P1: Do products_api_xx.js to get data from an api url**
hitchung committed 3 weeks ago · 1 / 1

=> git log for w13 and w12

The screenshot shows a GitHub repository's commit history for two weeks, with specific commits highlighted.

W13 commits (Top Section):

- w13 finish up
- W13-P4: Create company_xx and store_xx tables, and insert all data
- W13-P3: Use menuSupa.html, menuSupa.js to get 3 menu data from Supabase
- W13-P2: Create 3 menu data from Supabase
- W13-P1: Get 3 menu data from /api/data_xx.json
- add test.html, test.js to test company_xx, store_xx in Supabase
- add dev/w14 store code
- add md/w13 template, dev/w13 code

W12 commits (Bottom Section):

- Commits on May 9, 2024
- modify w12_xx.md
- w12 finish up
- W12-P5: Do 3_json_people_async_await
- W12-P4: Do 3_json_people_fetch_api
- W12-P3: Do 3_json_people_xhr
- W12-P2: Do 2_add_btn
- W12-P1: Do 1_simple-text
- add md/w12 template

=> git log for w11 and w10

github.com/htchung/1122-js-demo-xx/commit...

Commits on May 2, 2024

- W11-P3: Answer for '5-prommisse-demo'
 - htchung committed last month · ✓ 1 / 1
- W11-P2: Answer p3_xx in mid21
 - htchung committed last month · ✓ 1 / 1
- W11-P1: Answer p2_xx in mid21
 - htchung committed last month · ✓ 1 / 1

make changes for w11 class

- htchung committed last month · ✓ 1 / 1

add md/w11 template, demo/w11 & w12 code

- htchung committed last month · ✓ 1 / 1

Commits on Apr 25, 2024

w10 finish up

- htchung committed 2 months ago · ✓ 1 / 1

- W10-P4: Show products retrieved from Supabase
 - htchung committed 2 months ago · ✓ 1 / 1
- W10-P3: Get 9 products data from Supabase, and show in console
 - htchung committed 2 months ago · ✓ 1 / 1
- W10-P2: Get a photo from unsplash, and put it into storage in Supabase
 - htchung committed 2 months ago · ✓ 1 / 1
- W10-P1: In Supabase, create product_xx table with 8 product data
 - htchung committed 2 months ago · ✓ 1 / 1

add api course materials from 1112, and put them into dev

- htchung committed 2 months ago · ✓ 1 / 1

add demo/w10 template

Your Answer

=> My Github URL for git logs · 標註本學期 git log 次數



=> git log for w15 and w14



=> git log for w13 and w12



=> git log for w11 and w10



=> git log

(10%) P1-3: 佈署 w15 product demo 到 Vercel，並顯示如下資訊

w15 product demo in Vercel

=> 顯示在 Vercel 上的 demo

The screenshot shows a web browser window with the URL `1122-js-demo-xx.vercel.app/#` in the address bar. The page content is a product demo application. On the left, there's a sidebar titled "Company" with options: All, Ikea (selected), Marcos, Caressa, and Liddy. The main area displays two products: "High-Back Bench" and "Wooden Table". The "High-Back Bench" card includes an image, a title, a price (\$9.99), and a "remoteImg" URL. The "Wooden Table" card also includes an image, a title, a price (\$19.99), and a "remoteImg" URL. On the right, the DevTools console shows an error message: "Expression not available" for `companyName_ikea`. Below it, a stack trace for "Ikea products" is shown, pointing to `products_sup_xx.js:35`. The stack trace details the creation of an array of three objects, each representing a product from the "Ikea" company.

```
companyName_ikea
Ikea products
  ▾ Array(3) 1
    ▾ 0:
      companyId: 1
      company_xx: {id: 1, name: 'Ikea', email: 'ikea@gmail.com', phone: '0978543210'}
      created_at: "2024-05-23T12:32:47.006356"
      id: 3
      localImg: "./images/product-3.jpg"
      price: 9.99
      remoteImg: "https://ifafquhvqbvtzaghf.supabase.co/storage/v1/object/p
      title: "High-Back Bench"
    ▾ 1: {id: 4, created_at: '2024-05-23T12:32:47.006356', title: 'Wooden Table'}
    ▾ 2: {id: 7, created_at: '2024-05-23T12:32:47.006356', title: 'Albany Sectio
length: 3
[[Prototype]]: Object
```

Your Answer

w15 product demo in Vercel

=> 顯示在 Vercel 上的 demo



=> git log

(10%) P1-4: Supabase SQL 基本操作

請用 SQL 產生 blog_f2_xx table，並匯入 6 筆資料，第一筆資料的 id 必須是你學號的最後一個數字，結果請從 Vercel 上顯示

=> 執行 SQL 指令

The screenshot shows the Supabase dashboard with the URL `supabase.com/dashboard/project/sldvcsgcmh...`. The table `tku_xx` is selected. A red box highlights the SQL code in the query editor:

```
1 CREATE TABLE blog_f2_xx (
2   id int NOT NULL PRIMARY KEY,
3   title varchar(255),
4   description varchar(255),
5   category varchar(255),
6   local_img varchar(255),
7   remote_img varchar(255)
8 );
9
10 INSERT INTO blog_f2_xx (id, title, description, category, local_img, remote_img)
11 VALUES
12 (7, 'Sun Rise From The Mountain', 'Lorem ipsum dolor sit amet consectetur adipisicing elit.', 'travel', './images/photo-7.jpg', ''),
13 (8, 'Serene Lake With Trees', 'Lorem ipsum dolor sit amet consectetur adipisicing elit.', 'travel', './images/photo-8.jpg', ''),
14 (9, 'Rocks Of Queen Head In Yehliu Taiwan', 'Lorem ipsum dolor sit amet consectetur adipisicing elit.', 'travel', './images/photo-9.jpg', ''),
15 (1, 'Seven Reasons Why Coffee Is Awesome', 'Lorem ipsum dolor sit amet consectetur adipisicing elit.', 'lifestyle', './images/photo-1.jpg', ''),
16 (2, 'Travel To Paris', 'Lorem ipsum dolor sit amet consectetur adipisicing elit.', 'travel', './images/photo-2.jpg', ''),
17 (3, 'Coffee Brings Friendship', 'Lorem ipsum dolor sit amet consectetur adipisicing elit.', 'lifestyle', './images/photo-3.jpg', '')
```

The results section shows "Success. No rows returned".

=> 用 SQL 指令取得 blog_f2_xx 所有資料

```
select * from blog_f2_xx;
```

The screenshot shows the Supabase dashboard with the URL `supabase.com/dashboard/project/sldvcsgcmh...`. The table `tku_xx` is selected. The SQL Editor contains the following code:

```
10 INSERT INTO blog_f2_xx (id, title, description, category, local_img, remote_img)
11 VALUES
12 (7, 'Sun Rise From The Mountain', 'Lorem ipsum dolor sit amet consectetur adipisicing elit.', 'travel', './images/photo-7.jpg', ''),
13 (8, 'Serene Lake With Trees', 'Lorem ipsum dolor sit amet consectetur adipisicing elit.', 'travel', './images/photo-8.jpg', ''),
14 (9, 'Rocks Of Queen Head In Yehliu Taiwan', 'Lorem ipsum dolor sit amet consectetur adipisicing elit.', 'travel', './images/photo-9.jpg', ''),
15 (1, 'Seven Reasons Why Coffee Is Awesome', 'Lorem ipsum dolor sit amet consectetur adipisicing elit.', 'lifestyle', './images/photo-1.jpg', ''),
16 (2, 'Travel To Paris', 'Lorem ipsum dolor sit amet consectetur adipisicing elit.', 'travel', './images/photo-2.jpg', ''),
17 (3, 'Coffee Brings Friendship', 'Lorem ipsum dolor sit amet consectetur adipisicing elit.', 'lifestyle', './images/photo-3.jpg', '')
```

Below the editor, a red box highlights the query `select * from blog_f2_xx;`. The results section shows a table with the following data:

id	title	description	category	local_img	remote_img
7	"Sun Rise From The Mountain"	"Lorem ipsum dolor sit amet consectetur adipisicing elit."	"travel"	"./images/photo-7.jpg"	"./images/photo-7.jpg"
8	"Serene Lake With Trees"	"Lorem ipsum dolor sit amet consectetur adipisicing elit."	"travel"	"./images/photo-8.jpg"	"./images/photo-8.jpg"
9	"Rocks Of Queen Head In Yehliu Taiwan"	"Lorem ipsum dolor sit amet consectetur adipisicing elit."	"travel"	"./images/photo-9.jpg"	"./images/photo-9.jpg"
1	"Seven Reasons Why Coffee Is Awesome"	"Lorem ipsum dolor sit amet consectetur adipisicing elit."	"lifestyle"	"./images/photo-1.jpg"	"./images/photo-1.jpg"
2	"Travel To Paris"	"Lorem ipsum dolor sit amet consectetur adipisicing elit."	"travel"	"./images/photo-2.jpg"	"./images/photo-2.jpg"
3	"Coffee Brings Friendship"	"Lorem ipsum dolor sit amet consectetur adipisicing elit."	"lifestyle"	"./images/photo-3.jpg"	"./images/photo-3.jpg"

Your Answer

=> 執行 SQL 指令



=> blog_f2_xx table 顯示



=> git log

以下 P2, P3, P4, P5 題目請透過老師提供的 theme-crown 來實作，如有 xx 需改成學號後兩碼。

(20%) P2: 實作靜態畫面，點選上方選單 Shop_xx, logo 及主頁面的某 category，可以正常切換

請以學號最後一碼除以 5 取餘數 (value) · 取得 category

value =1, category = 'hats'

value =2, category = 'jackets'

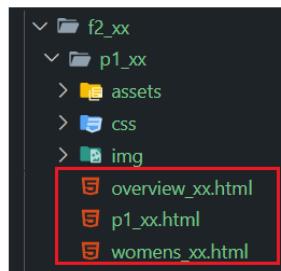
value =3, category = 'sneakers'

value =4, category = 'womens'

value =5, category = 'mens'

本題老師選擇 value = 4 (womens) 來實作

=> 目錄結構及檔名如下



=> 執行 p1_xx.html 顯示頁面如下，請框住你需要實作的 category 圖片，本題老師 demo 是 womens

A screenshot of a web browser displaying a shopping website. The URL bar shows the address: 127.0.0.1:5502/exams/final2-w17/f2_xx_answer/f2_xx/p1_xx/p1_xx.html. The page features a navigation bar with links for Cal, iClass, Gmail, YouTube, EN, P1, Web, Udemy, GitHub, and Supabase. On the left is a user profile icon with the name 'htchung, 987612354'. On the right are buttons for 'Shop_xx', 'Contact', 'Sign In', and a shopping cart icon. Below the navigation, there are five product categories displayed in a grid:

- HATS** SHOP NOW (highlighted with a red box)
- JACKETS** SHOP NOW
- SNEAKERS** SHOP NOW
- WOMENS** SHOP NOW (highlighted with a red box)
- MENS** SHOP NOW

The background of the website shows a person wearing a cap and a woman in a hoodie.

=> 顯示本功能修改了那些 code



=> 點選你需要實作之 category 圖片，要能顯示該 category 下的所有產品圖片，本題顯示 womens 下的 7 張圖片

htchung, 987612354

Shop_xx Contact Sign In

Womens

Blue Tanktop	25	Floral Blouse	20	Floral Dress	80	Red Dots Dress	80

=> 顯示本功能修改了那些 code



=> 點選 logo 回到 p1_xx.html

The screenshot displays a web application interface. At the top, there's a navigation bar with various icons and links. The main content area features a grid of five product categories, each with a small image and a 'SHOP NOW' button. The categories are: HATS, JACKETS, SNEAKERS, WOMENS, and MENS. Each category image has a red box around it, likely indicating an interactive element.

=> 顯示本功能修改了那些 code



=> 點選 Shop_xx 顯示 2 個 category 圖片，每個 category 4 張圖片

老師 demo，使用的學號後 2 碼是 54，4 屬 womens, 5 屬 mens，因此先顯示 womens 4 張圖片，再顯示 mens 4 張圖片，如果兩個數字一樣，則取下一個數字，如果是 55，就取 mens (5) 及 hats (1)

htchung, 987612354

Womens

Blue Tanktop

Floral Blouse

Floral Dress

Red Dots Dress

Mens

=> 顯示本功能修改了那些 code



Your Answer

=> 目錄結構及檔名



=> 執行 p1_xx.html 顯示頁面如下，請框住你需要實作的 category 圖片，本題老師 demo 是 womens



=> 顯示本功能修改了那些 code



=> 點選你需要實作之 category 圖片，要能顯示該 category 下的所有產品圖片，本題顯示 womens 下的 7 張圖片



=> 顯示本功能修改了那些 code



=> 點選 logo 回到 p1_xx.html



=> 顯示本功能修改了那些 code



=> 點選 Shop_xx 顯示 2 個 category 圖片，每個 category 4 張圖片

老師 demo，使用的學號後 2 碼是 54，4 屬 womens, 5 屬 mens，因此先顯示 womens 4 張圖片，再顯示 mens 4 張圖片



=> 顯示本功能修改了那些 code



=> git log

請將 p1_xx 目錄下所有檔案，複製到 p2_xx 準備從 Supabase 將資料放入 womens_xx.html (或其他你需要做的 category), overview_xx.html

(15%) P3: 請將 P2 題所需要的資料用 SQL 來建立

本說明是以 womens 7 張圖片及 mens 6 張圖片來說明

=> 用 SQL create table 建立 store_f2_xx，其中必須有一個欄位要建立 category_id，其資料設定為 1~5 (如 P2 題之設定)，用 SQL insert into 匯入 womens 7 張圖片資料。create table 跟 insert into 要一次執行，能建立及匯入資料。

category_f2_xx SQL 指令老師會提供，請先執行，再進行本題

htc.tku@gmail.com's Org Free / tku_xx / Enable branching

```

1 CREATE TABLE store_f2_xx (
2     id int NOT NULL PRIMARY KEY,
3     name varchar(255),
4     category_id int,
5     price real,
6     local_img varchar(255),
7     remote_img varchar(255) ...
8 );
9
10 INSERT INTO store_f2_xx (id, name, category_id, price, local_img, remote_img)
11 VALUES
12 (1, 'Blue Tanktop', 4, 25, '/img/womens/blue-tank.png', 'https://i.ibb.co/7CQVJNm/blue-tank.png'),
13 (2, 'Floral Blouse', 4, 20, '/img/womens/floral-blouse.png', 'https://i.ibb.co/4W2DGKm/floral-blouse.png'),
14 (3, 'Floral Dress', 4, 80, '/img/womens/floral-skirt.png', 'https://i.ibb.co/KV18Ysr/floral-skirt.png'),
15 (4, 'Red Dots Dress', 4, 80, '/img/womens/red-polka-dot-dress.png', 'https://i.ibb.co/N3BN1bh/red-polka-dot-dress.png'),
16 (5, 'Striped Sweater', 4, 45, '/img/womens/blue-tank.png', 'https://i.ibb.co/KmSkMbH/striped-sweater.png'),
17 (6, 'Yellow Track Suite', 4, 135, '/img/womens/floral-blouse.png', 'https://i.ibb.co/v1cvwNf/yellow-track-suit.png'),
18 (7, 'White Blouse', 4, 20, '/img/womens/floral-skirt.png', 'https://i.ibb.co/qBcrsJg/white-vest.png');
19
20 select * from store_f2_xx where category_id = 4;

```

Results Chart Export 100 rows Source: Primary Database role postgres Run selected

Success. No rows returned

=> 並下 SQL 指令搜尋 womens 資料

htc.tku@gmail.com's Org Free / tku_xx / Enable branching

```

10 INSERT INTO store_f2_xx (id, name, category_id, price, local_img, remote_img)
11 VALUES
12 (1, 'Blue Tanktop', 4, 25, '/img/womens/blue-tank.png', 'https://i.ibb.co/7CQVJNm/blue-tank.png'),
13 (2, 'Floral Blouse', 4, 20, '/img/womens/floral-blouse.png', 'https://i.ibb.co/4W2DGKm/floral-blouse.png'),
14 (3, 'Floral Dress', 4, 80, '/img/womens/floral-skirt.png', 'https://i.ibb.co/KV18Ysr/floral-skirt.png'),
15 (4, 'Red Dots Dress', 4, 80, '/img/womens/red-polka-dot-dress.png', 'https://i.ibb.co/N3BN1bh/red-polka-dot-dress.png'),
16 (5, 'Striped Sweater', 4, 45, '/img/womens/blue-tank.png', 'https://i.ibb.co/KmSkMbH/striped-sweater.png'),
17 (6, 'Yellow Track Suite', 4, 135, '/img/womens/floral-blouse.png', 'https://i.ibb.co/v1cvwNf/yellow-track-suit.png'),
18 (7, 'White Blouse', 4, 20, '/img/womens/floral-skirt.png', 'https://i.ibb.co/qBcrsJg/white-vest.png');
19
20 select * from store_f2_xx where category_id = 4;

```

Results Chart Export 100 rows Source: Primary Database role postgres Run selected

id	name	category_id	price	local_img	remote_img
1	"Blue Tanktop"	4	25	"/img/womens/blue-tank.png"	"https://i.ibb.co/7CQVJNm/blue-tank.png"
2	"Floral Blouse"	4	20	"/img/womens/floral-blouse.png"	"https://i.ibb.co/4W2DGKm/floral-blouse.png"
3	"Floral Dress"	4	80	"/img/womens/floral-skirt.png"	"https://i.ibb.co/KV18Ysr/floral-skirt.png"
4	"Red Dots Dress"	4	80	"/img/womens/red-polka-dot-dress.png"	"https://i.ibb.co/N3BN1bh/red-polka-dot-dress.png"
			45	"/img/womens/blue-tank.png"	"https://i.ibb.co/KmSkMbH/striped-sweater.png""
			135	"/img/womens/floral-blouse.png"	"https://i.ibb.co/v1cvwNf/yellow-track-suit.png"
7	"White Blouse"	4	20	"/img/womens/floral-skirt.png"	"https://i.ibb.co/qBcrsJg/white-vest.png"

Your Answer

=> 用 SQL create table 建立 store_f2_xx



=> 用 SQL insert into 匯入 womens 7 張圖片資料，並下 SQL 指令搜尋 womens 資料

標題 womens 請改成你所需要實作的 category



=> 用 SQL insert into 匯入 mens 6 張圖片資料，並下 SQL 指令搜尋 mens 資料

標題 mens 請改成你所需要實作的 category，顯示如上 womens 截圖，先執行 insert into，再執行 select 篩選指令



=> git log

(15%) P4: 請將 womens.html 內的 7 張圖片，改成是自 Supabase 取得並套用。

=> 顯示 womens_xx.html，圖片顯示的區塊是空的；顯示 Supabase API keys

```
womens_xx.html
12 <img src= "./assets/crown.svg" />
13 </a>
14 <div>
15 <h2 style="margin-top: 2rem">hchung,
16 987612345</h2>
17 <div class="options"> ...
18 </div>
19 <div class="shop-page">
20 <div class="collection-page">
21 <h1 class="title">Womens</h1>
22 <div class="items"></div>
23 </div>
24 <script>
25 function cartToggle() {
26   p = document.querySelector('.cart-dropdown');
27   p.classList.toggle('show');
28 }
29 </script>
30 <script
31 type="module"
32 src="https://cdn.jsdelivr.net/npm/@supabase/
33 supabase-js"
34 ></script>

clientSupabase_xxjs
1 const SUPABASE_URL = 'https://sldvcsbcmhhgtrjjccxd.supabase.co';
2 const SUPABASE_ANON_KEY =
3 'eyJhbGciOiJIUzI1NiIsInR5cI6IkpxVCJ9.
4 eyJpc3MiOiJzdXBhYmFzZSIsInIzIi6InNzHZjc2djbwZ3Rya
5 mpjYhkLiwiem9sZSI6Imfub24iLCJpYXQiOjE2ODMwODE3MjQsIm
6 V4cC16MTk5ODY1NzcyNj0.
7 JsxiJygoCFy05Fz6Ng--phtkMFEBKTACBbG732aX2bI';
```

=> Supabase API key 必須和上一小題相符合

The screenshot shows the Supabase dashboard under the 'API Settings' tab. The 'Project URL' is displayed as <https://sldvcsgcmhhgtrjjccxd.supabase.co>. Below it, the 'Project API keys' section shows an 'anon' key labeled 'public'. A red box highlights the 'Project URL' and the 'anon' key area.

=> Chrome 顯示可以呈現 womens 7 張圖片，要用 console.log 呈現有自 Supabase 取的資料

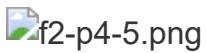
The screenshot shows a web browser displaying a shopping page for women's clothing. The top navigation bar includes a user profile with the name 'htchung, 987612354'. The main content area shows four items: 'Blue Tanktop', 'Floral Blouse', 'Floral Dress', and 'Red Dots Dress'. The 'Blue Tanktop' item has a red box around its image and an 'Add to Cart' button. To the right, the browser's developer tools are open, specifically the 'Console' tab. It shows an error message: 'Uncaught (in promise) Error: Could not establish connection. Receiving end does not exist.' Below the error, a 'store f2_xx data' object is shown as an array of 7 objects, each representing a product with fields like category_id, id, local_img, name, price, and remote_img. A red box highlights this data structure.

Your Answer

=> 顯示 womens_xx.html，圖片顯示的區塊是空的；顯示 Supabase API keys



=> Supabase API key 必須和上一小題相符合



=> Chrome 顯示可以呈現 **womens** 7 張圖片，要用 **console.log** 呈現有自 Supabase 取的資料

標題 **womens** 請改成你所需要實作的 category



=> 顯示根本題有關的 **js/p2_xx.js code**



=> **git log**

(10%) P5: 請將 **overview_xx.html** 內的 **womens** 4 張圖片 及 **mens** 4 張圖片，
換成是從 **Supabase** 取得並套用。

Hint: 可取得 **womens** 所有資料後再用 **slice** 陣列指令取得前四筆，總共要自 **Supabase** 取兩次，第一次
取的 **womens** 資料，第二次取得 **mens** 資料

=> 顯示 overview_xx.html 中顯示 Women 圖片之區塊是空的

The screenshot shows the VS Code interface with the following details:

- EXPLORER:** Shows the project structure under "1122-JS-DEMO-XX".
- CODE EDITOR:** Displays the content of "overview_xx.html".
 - A red box highlights the section from `

` to `

`.
 - A red box highlights the script block starting with `function cartToggle(){`.
 - A red box highlights the script tag `<script type="module" src=".js/p2_overview_xx.js">`.

=> Chrome 顯示可以呈現 womens 4 張圖片，要用 console.log 呈現自 Supabase 取的資料

The screenshot shows a browser window with the following elements:

- Address Bar:** Shows the URL: 127.0.0.1:5502/exams/final2-w17/f2_xx_answer/f2_xx/p2_xx/overview_xx.html
- Header:** Shows a user profile with the name "htchung, 987612354".
- Content:** Displays a grid of four women's clothing items:
 - Blue Tanktop
 - Floral Blouse
 - Floral Dress
 - Red Dots Dress
- Console:** Shows the Supabase data being logged.

```
Live reload enabled.
overview_xx.html:101
Uncaught (in promise)
Error: Could not establish connection. Receiving end does not exist.
p2_overview_xx.js:17
store_f2_xx_cid data 4
  ▼ Array(7) ④
    ▷ 0: {id: 1, name: 'Blue Tanktop', category_id: 4, price: 25, local_img: '...'}
    ▷ 1: {id: 2, name: 'Floral Blouse', category_id: 4, price: 20, local_img: '...'}
    ▷ 2: {id: 3, name: 'Floral Dress', category_id: 4, price: 80, local_img: '...'}
    ▷ 3: {id: 4, name: 'Red Dots Dress', category_id: 4, price: 80, local_img: '...'}
    ▷ 4: {id: 5, name: 'Stripped Sweater', category_id: 4, price: 45, local_im...
    ▷ 5: {id: 6, name: 'Yellow Track Suite', category_id: 4, price: 135, local_...
    ▷ 6: {id: 7, name: 'White Blouse', category_id: 4, price: 20, local_img: '...
    ▷ length: 7
    ▷ [[Prototype]]: Array(0)
```

=> Chrome 顯示可以呈現 mens 4 張圖片，要用 console.log 呈現自 Supabase 取的資料

```
store_f2_xx cId data 4 [Object, Object, Object, Object]
store_f2_xx cId data 5 [Object, Object, Object, Object, Object, Object]
```

The screenshot shows a web page with a grid of four women's dresses at the top. Below this is a section titled 'Mens' containing four men's shirts. On the right, the browser's developer tools are open, specifically the JavaScript console. It displays two arrays of objects, each representing a category of products. The first array has 4 items, and the second has 5 items. Each item contains properties like id, name, category_id, and price.

Category	Product Name	Category ID	Price
Women	Blue Tanktop	25	
	Floral Blouse	20	
	Floral Dress	80	
	Red Dots Dress	80	
Mens	Camo Down Vest	325	
	Floral T-shirt	20	
	Black & White Longsleeve	25	
	Pink T-shirt	25	

Your Answer

=> 顯示 overview_xx.html 中顯示 Women 圖片之區塊是空的

標題 womens 請改成你所需要實作的 category



=> Chrome 顯示可以呈現 womens 4 張圖片，要用 console.log 呈現有自 Supabase 取的資料



=> 顯示跟取得 womens 有關的 js/p2_xx.js code



=> 顯示 overview_xx.html 中顯示 Mens 圖片之區塊是空的

標題 Mens 請改成你所需要實作的 category



=> Chrome 顯示可以呈現 Mens 4 張圖片，要用 console.log 呈現有自 Supabase 取的資料



=> 顯示跟取得 mens 有關的 js/p2_xx.js code



=> git log