

P0: Github repo URL

<https://github.com/409411351/1102-db2-409411351.git>

P1: 資料庫截圖，要能顯示 shop_xx, 及所要實作之產品 data

The screenshot shows the PgAdmin interface. On the left, the database browser displays a tree structure of database objects. A red box highlights the 'Columns (4)' section under the 'shop_51' table. The columns listed are id, name, price, and local_url. On the right, the query editor window shows a SELECT query and its results. The results table is also highlighted with a red border and contains the following data:

	id [PK] integer	name character varying	price integer	local_url character varying
1	1	Brown Brim	25	https://i.ibb.co/ZYW3VTp/brown-brim.png
2	2	Blue Beanie	18	https://i.ibb.co/ypkgK0X/blue-beanie.png
3	3	Brown Cowboy	35	https://i.ibb.co/QdJwgmp/brown-cowboy.png
4	4	Grey Brim	25	https://i.ibb.co/RjBLWxB/grey-brim.png
5	5	Green Beanie	18	https://i.ibb.co/YTjW3vF/green-beanie.png
6	6	Palm Tree	14	https://i.ibb.co/rkBDvJX/palm-tree-cap.png
7	7	Red Beanie	18	https://i.ibb.co/bLB646Z/red-beanie.png
8	8	Wolf Cap	14	https://i.ibb.co/1f2nWMM/wolf-cap.png
9	9	Blue Snapback	16	https://i.ibb.co/X2VJP2W/blue-snapback.png

P2: 實作一個 model, Shop_xx，要能從 shop_xx 取的資料，以字串顯示在 console 中

檔案(F) 編輯(E) 選取項目(S) 檢視(V) 移至(G) 執行(R) 終端機(T) 說明(H)

檔案總管

1102-db2-409411351

- bin
- classdemo\md
- hw1
 - hw1_51.md
 - p1.png
 - w02
 - w03
 - w05
 - w06
- models
 - Category_51.js
 - shop_51.js
- node_modules
- public
- routes
- utils
- views
- .gitignore
- app.js
- package-lock.json
- package.json

shop_51.js - 1102-db2-409411351 - Visual Studio Code

```

static async fetchAll(){
  try{
    let results = await db.query('SELECT * from shop_51');
    //console.log('results', results.rows);
    return results.rows;
  }
  catch(err){
    console.log('error',err);
  }
}

//testing
const test = async ()=>{
  let results = await shop_51.fetchAll();
  console.log('test results', JSON.stringify(results));
}

test();
module.exports = shop_51;

```

開源 輸出 線性檢索 命令主控台

```

henry@DESKTOP-FUJ970NG MINGW64 /d/1102-db2-409411351/models (main)
$ node shop_51.js
test results [{"id":1,"name":"Brown Brim","price":25,"local_url":"https://i.ibb.co/ZYw3VTP/brown-brim.png"}, {"id":2,"name":"Blue Beanie","price":18,"local_url":"https://i.ibb.co/yPKgK0X/blue-beanie.png"}, {"id":3,"name":"Brown Cowboy","price":35,"local_url":"https://i.ibb.co/QdJwmpB/brown-cowboy.png"}, {"id":4,"name":"Grey Brim","price":25,"local_url":"https://i.ibb.co/RjBLWk6/grey-brim.png"}, {"id":5,"name":"Green Beanie","price":18,"local_url":"https://i.ibb.co/Tjkh3vF/green-beanie.png"}, {"id":6,"name":"Palm Tree","price":14,"local_url":"https://i.ibb.co/k8DwIX/palm-tree-cap.png"}, {"id":7,"name":"Red Beanie","price":18,"local_url":"https://i.ibb.co/bLB64GZ/red-beanie.png"}, {"id":8,"name":"Wolf Cap","price":14,"local_url":"https://i.ibb.co/f2nWMM/wolf-cap.png"}, {"id":9,"name":"Blue Snapback","price":16,"local_url":"https://i.ibb.co/X2VJP2W/blue-snapback.png"}]
henry@DESKTOP-FUJ970NG MINGW64 /d/1102-db2-409411351/models (main)
$ 

```

> 大圖
> 時間表

P3: 實作本作業所需要的路由，能夠將 shop_xx 資料套用在 ejs 中，請顯示 routes, ejs 程式碼及 Chrome 顯示圖片

YouTube iClass HW1-Crown 實作範例 /crown/ 409411351/1102-db2-409411351/... Crown Clothing Crown Clothing

localhost:3000/crown2_51/shop_51/hats

Shop Contact Sign In

Hats

Brown Brim 25
Blue Beanie 18
Brown Cowboy 35
Grey Brim 25
Green Beanie 18
Palm Tree 14
Red Beanie 18
Wolf Cap 14

Blue Snapback 16

views > crown2_51 > hats.ejs > HTML > body > div.header > div.options > div.cart-icon > img.shopping-icon

```

<% for(let i=0; i<data.length; i++){ %>
  <div class="collection-item">
    
    <div class="collection-footer">
      <span class="name"><%= data[i].name %></span>
      <span class="price"><%= data[i].price %></span>
    </div>
    <button class="custom-button">Add to Cart</button>
  </div>
<% } %>

```

P4: 請將資料庫放入 Heroku PostgreSQL，並能正常顯示 shop_xx, category_xx data。請給予 Heroku DATABASE_URL，截圖要有

Heroku 上 database, shop_xx 名稱及資料。

<https://data.heroku.com/datastores/b44d032e-0531-4b7b-8da9-dbd514b85759#dataclips>

postgres://sqvrdzfinrrlfj:

ca544980eabbbe395305c3f845c4548d2160af7af00cb360ca9f7d588728dff6@

ec2-44-192-245-97.compute-1.amazonaws.com:

5432/

d4047nahm5meu4

The screenshot shows the pgAdmin 4 interface. On the left, the Browser pane displays a tree view of database objects. A red box highlights the 'Tables (2)' section under the 'shop_51' schema, which contains entries for 'category_51' and 'shop_51'. Another red box highlights the 'shop_51' table itself. The main pane shows the Query Editor with the following SQL query:

```
1 SELECT * FROM public.shop_51
2 ORDER BY id ASC
```

The Data Output tab shows the results of the query:

	id [PK] integer	name character varying	price integer	local_url character varying
1	1	Brown Brim	25	https://i.ibb.co/ZYW3VTp/brown-brim.png
2	2	Blue Beanie	18	https://i.ibb.co/ypkgK0X/blue-beanie.png
3	3	Brown Cowboy	35	https://i.ibb.co/QdJWgmp/brown-cowboy.png
4	4	Grey Brim	25	https://i.ibb.co/RjBLWxB/grey-brim.png
5	5	Green Beanie	18	https://i.ibb.co/YTjW3vF/green-beanie.png
6	6	Palm Tree	14	https://i.ibb.co/rKDvJX/palm-tree-cap.png
7	7	Red Beanie	18	https://i.ibb.co/bLB646Z/red-beanie.png
8	8	Wolf Cap	14	https://i.ibb.co/1f2nWMM/wolf-cap.png
9	9	Blue Snapback	16	https://i.ibb.co/X2VJP2W/blue-snapback.png

P5: 在 Heroku 上能正確執行本作業之路由，請提供 Heroku URL

<https://dashboard.heroku.com/apps/crown-1102-db2-409411351/settings>

