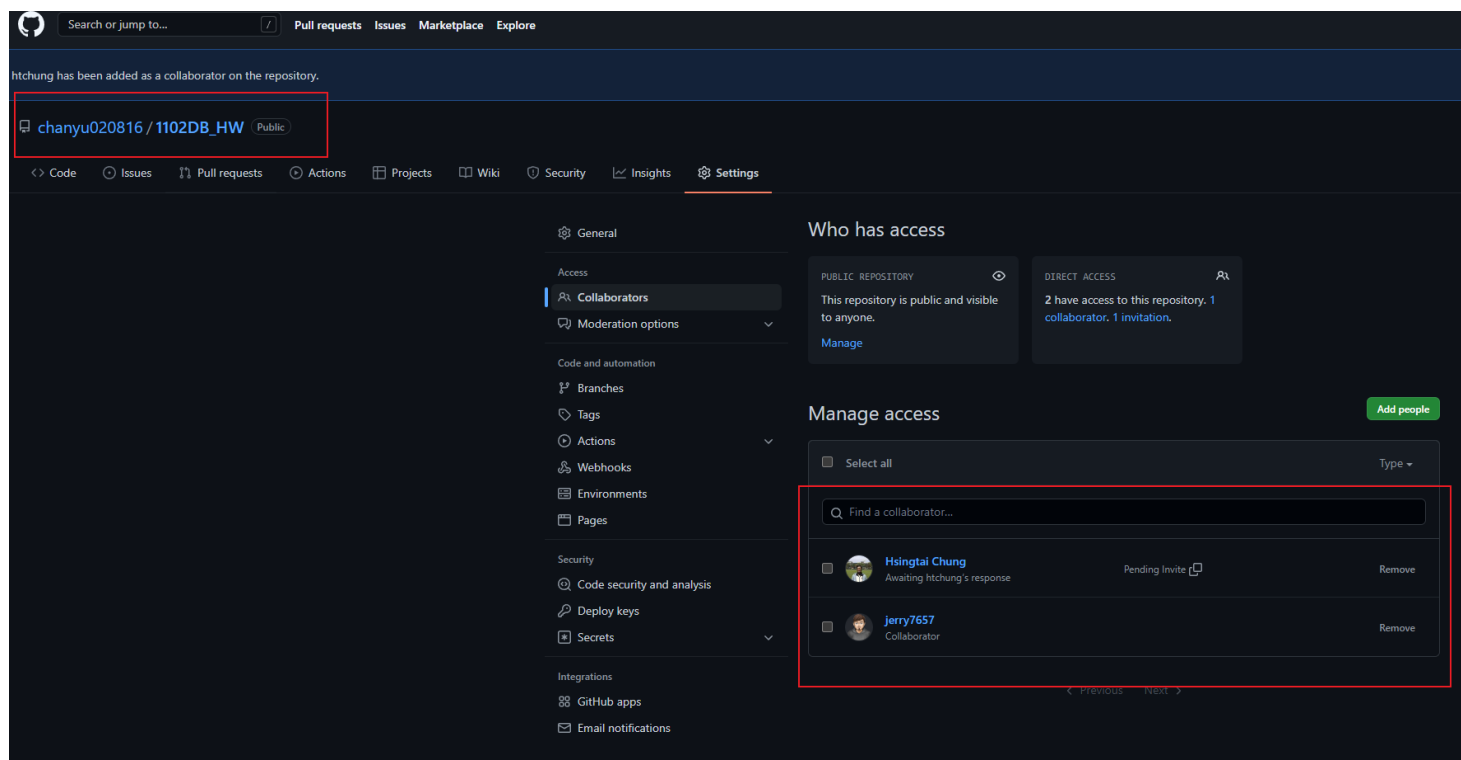


P0: Github repo URL, 請分享給老師 (htchung@gms.tku.edu.tw) 及助教 (jerry7657)



P1: 資料庫截圖，要能顯示 shop_xx, 及所要實作之產品 data

crown2

>

Casts

>

Catalogs

>

Event Triggers

>

Extensions

>

Foreign Data Wrappers

>

Languages

>

Publications

>

Schemas (1)

>

public

>

Aggregates

>

Collations

>

Domains

>

FTS Configurations

>

FTS Dictionaries

>

FTS Parsers

>

FTS Templates

>

Foreign Tables

>

Functions

>

Materialized Views

>

Operators

>

Procedures

>

Sequences

>

Tables (2)

>

category_12

>

Columns

>

Constraints

>

Indexes

>

RLS Policies

>

Rules

>

Triggers

>

jacket_12

>

Trigger Functions

>

Types

>

Views

>

Subscriptions

Property

SQL

Statistics

Dependencies

Dependents

public.jacket_12/crown2/postgres@PostgreSQL 14

Query Editor

Query History

1

SELECT * FROM public.jacket_12

2

ORDER BY id ASC

Data Output

Explain

Messages

Notifications

id

[PK] integer

name

character varying

local_url

character varying

prize

character varying

1

2

3

4

5

1

2

3

4

5

Black Jean Shearling

Blue Jean Jacket

Grey Jean Jacket

Brown Shearling

Tan Trench

https://i.ibb.co/Xzcv...

https://i.ibb.co/mJS...

https://i.ibb.co/N71k...

https://i.ibb.co/s96F...

https://i.ibb.co/M6h...

125

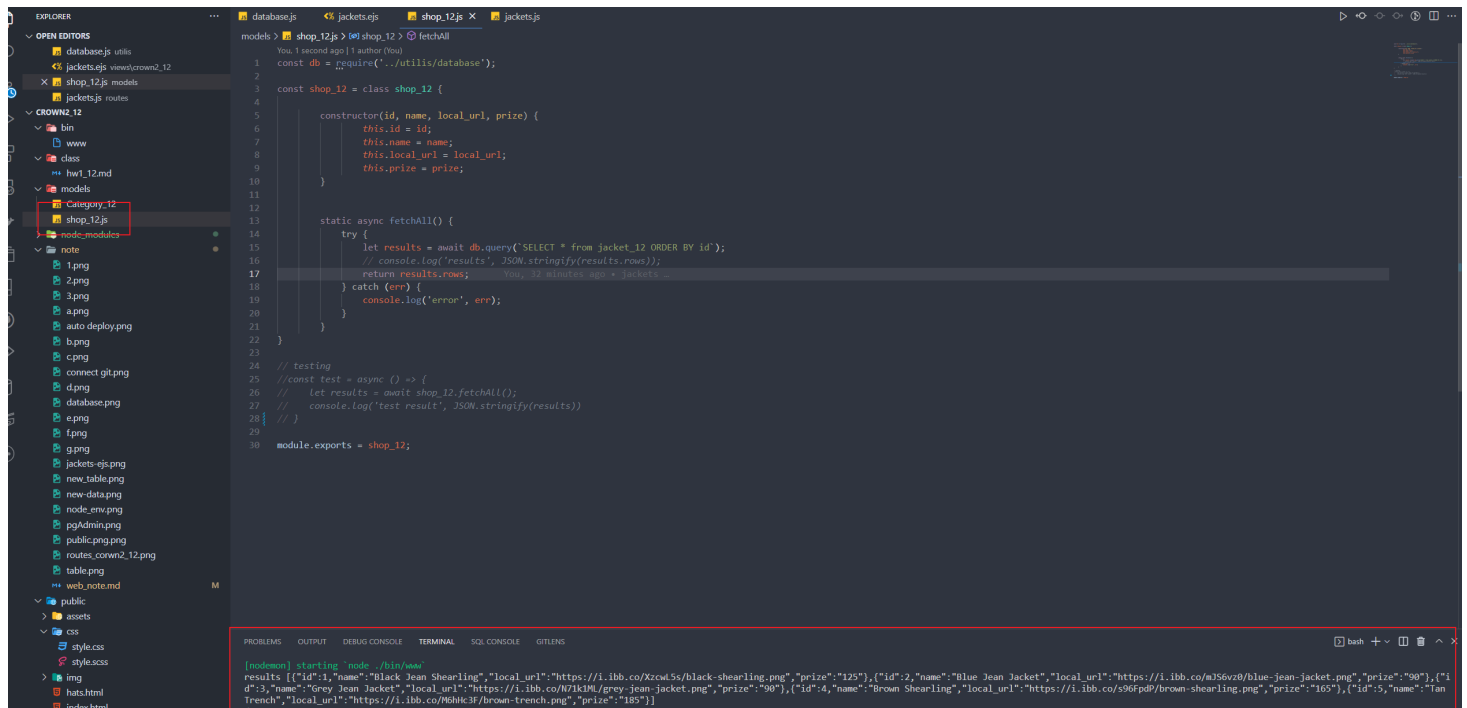
90

90

165

185

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



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

EXPLORER

OPEN EDITORS

database.js utilis

jackets.ejs views\crown2_12

shop_12.js models

jackets.js routes

CROWN2_12

e.png

f.png

g.png

jackets-ejs.png

new_table.png

new-data.png

node_env.png

pgAdmin.png

public.png.png

routes_corwn2_12.png

table.png

web_note.md

public

assets

css

style.css

style.scss

img

hats.html

index.html

jackets.html

mens.html

overview.html

sneakers.html

womens.html

routes

crown2_12.js

index.js

jackets.js

users.js

shop_12

database.js jackets.ejs shop_12.js jackets.js

routes > jackets.js > router.get('/') callback > data

You, 19 minutes ago | 1 author (You)

1 var express = require('express');

2 const shop_12 = require('../models/shop_12');

3 var router = express.Router();

4

5 /* GET home page. */

6 router.get('/', async function(req, res, next) {

7 try {

8 let results = await shop_12.fetchAll();

9 console.log('results', JSON.stringify(results));

10 res.render('crown2_12/jackets', {

11 data: results});

12 }catch(err){

13 console.log('error', err)

14 }

15

16 });

17

18 module.exports = router;

19

EXPLORER

OPEN EDITORS

- database.js utilis
- ✖ % jackets.ejs views\crown2_12
- shop_12.js models
- jackets.js routes
- % index.ejs views\crown2_12

CROWN2_12

- l.png
- g.png
- jackets-ejs.png
- new_table.png
- new-data.png
- node_env.png
- pgAdmin.png
- public.png.png
- routes_corwn2_12.png
- table.png
- web_note.md

public

- assets
- css
 - style.css
 - style.scss
- img
 - hats.html
 - index.html
 - jackets.html
 - mens.html
 - overview.html
 - sneakers.html
 - womens.html
- routes
 - JS crown2_12.js
 - JS index.js
 - JS jackets.js
 - JS users.js
- shop_12
- utilis
 - JS database.js
- views
 - JS crown2_12
 - % index.ejs
 - % jackets.ejs
 - error.ejs

views > crown2_12 > % jackets.ejs > html > body > div.shop-page

```
18 <div class="cart-icon" onclick="cartToggle()" >
19
20 </div>
21 <div class="cart-dropdown">
22 <div class="cart-items">
23 <div class="cart-item">
24 
25 <div class="item-details">
26 <span class="name">Brown Cowboy</span>
27 <span class="price">4 * $35</span>
28 </div>
29 </div>
30 <div class="cart-item">
31
32 </div>
33 </div>
34 <button class="custom-button">GO TO CHECKOUT</button>
35 </div>
36 </div>
37 </div>
38
39 <div class="shop-page">
40 <div class="collection-page">
41 <h1 class="title">Jackets</h1>
42 <div class="items">
43 <% for(let i=0; i<data.length; i++) { %>
44 <div class="collection-item">
45 
46 <div class="collection-footer">
47 <span class="name"><%= data[i].name %></span>
48 <span class="price"><%= data[i].prize %></span>
49 </div>
50 <button class="custom-button">Add to Cart</button>
51 </div>
52 <% } %>
53 </div>
54 </div>
55 </div>
56
57 <script>
58 function cartToggle(){
59 p = document.querySelector('.cart-dropdown');
60 p.classList.toggle("show");
61 }
62 </script>
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

[nodemon] starting `node ./bin/www`

results [{"id":1,"name":"Black Jean Shearling","local_url":"https://i.ibb.co/XzcwL5s/blac
d":3,"name":"Grey Jean Jacket","local url":"https://i.ibb.co/N71k1ML/grey-jean-jacket.png

EXPLORER

OPEN EDITORS

- database.js utilis
- jackets.ejs views\crown2_12
- shop_12.js models
- app.js
- jackets.js routes

CROWN2_12

- www
- class
 - hw1_12.md
 - p0.png
 - p1.png
 - p2.png
 - p3-chrome.png
 - p3-ejs.png
 - p3-routes.png
- models
 - Category_12
 - shop_12.js
- node_modules
- note
- public
 - assets
- css
 - style.css
 - style.scss
- img
 - hats.html
 - index.html
 - jackets.html
 - mens.html
 - overview.html
 - sneakers.html
 - womens.html
- routes
 - crown2_12.js
 - index.js
 - jackets.js
 - users.js
- shop_12
- utilis
 - database.js
- views
 - app.js
 - package-lock.json

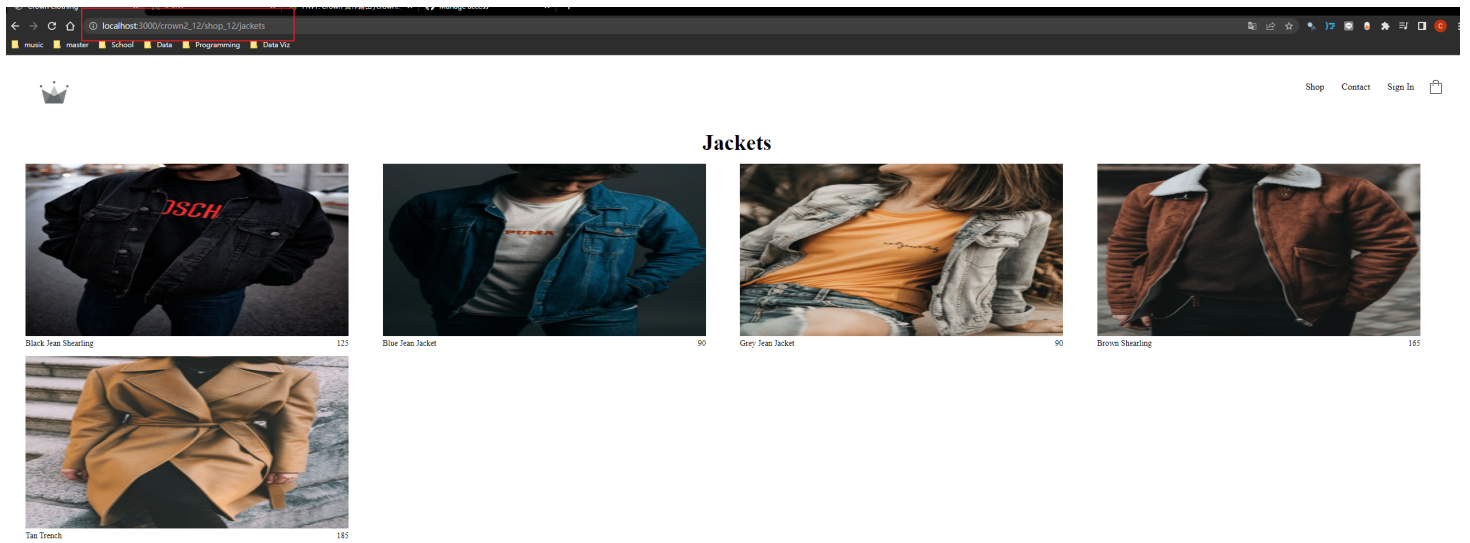
database.js jackets.ejs shop_12.js app.js jackets.ejs

app.js > jackets_Router

```
You, now | 1 author (You)
1  var createError = require('http-errors');
2  var express = require('express');
3  var path = require('path');
4  var cookieParser = require('cookie-parser');
5  var logger = require('morgan');
6
7  var indexRouter = require('./routes/index');
8  var usersRouter = require('./routes/users');
9
10 const crown2_12_Router = require('./routes/crown2_12');
11 const jackets_Router = require('./routes/jackets');
12
13 var app = express();
14
15 // view engine setup
16 app.set('views', path.join(__dirname, 'views'));
17 app.set('view engine', 'ejs');
18
19 app.use(logger('dev'));
20 app.use(express.json());
21 app.use(express.urlencoded({ extended: false }));
22 app.use(cookieParser());
23 app.use(express.static(path.join(__dirname, 'public')));
24
25 app.use('/', indexRouter);
26 app.use('/users', usersRouter);
27 app.use('/crown2_12', crown2_12_Router);
28 app.use('/crown2_12/shop_12/jackets', jackets_Router);
29
30 // catch 404 and forward to error handler
31 app.use(function(req, res, next) {
32   next(createError(404));
33 });
34
35 // error handler
36 app.use(function(err, req, res, next) {
37   // set locals, only providing error in development
38   res.locals.message = err.message;
39   res.locals.error = req.app.get('env') === 'development' ? err : {};
40
41   // render the error page
42   res.status(err.status || 500);
43   res.render('error');
44 });
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL SQL CONSOLE GITLENS

```
[nodemon] starting `node ./bin/www`
results [{"id":1,"name":"Black Jean Shearling","local_url":"https://d":3,"name":"Grey Jean Jacket","local_url":"https://i.ibb.co/N71k1MLTrench","local_url":"https://i.ibb.co/M6hHc3F/brown-trench.png","pri
de v
```



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。

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