

Frist One

← → ⌂ ⓘ localhost:3000

Express

Welcome to Express

My name is LEE, Bo-Yan

My student id is 209410256

index.ejs

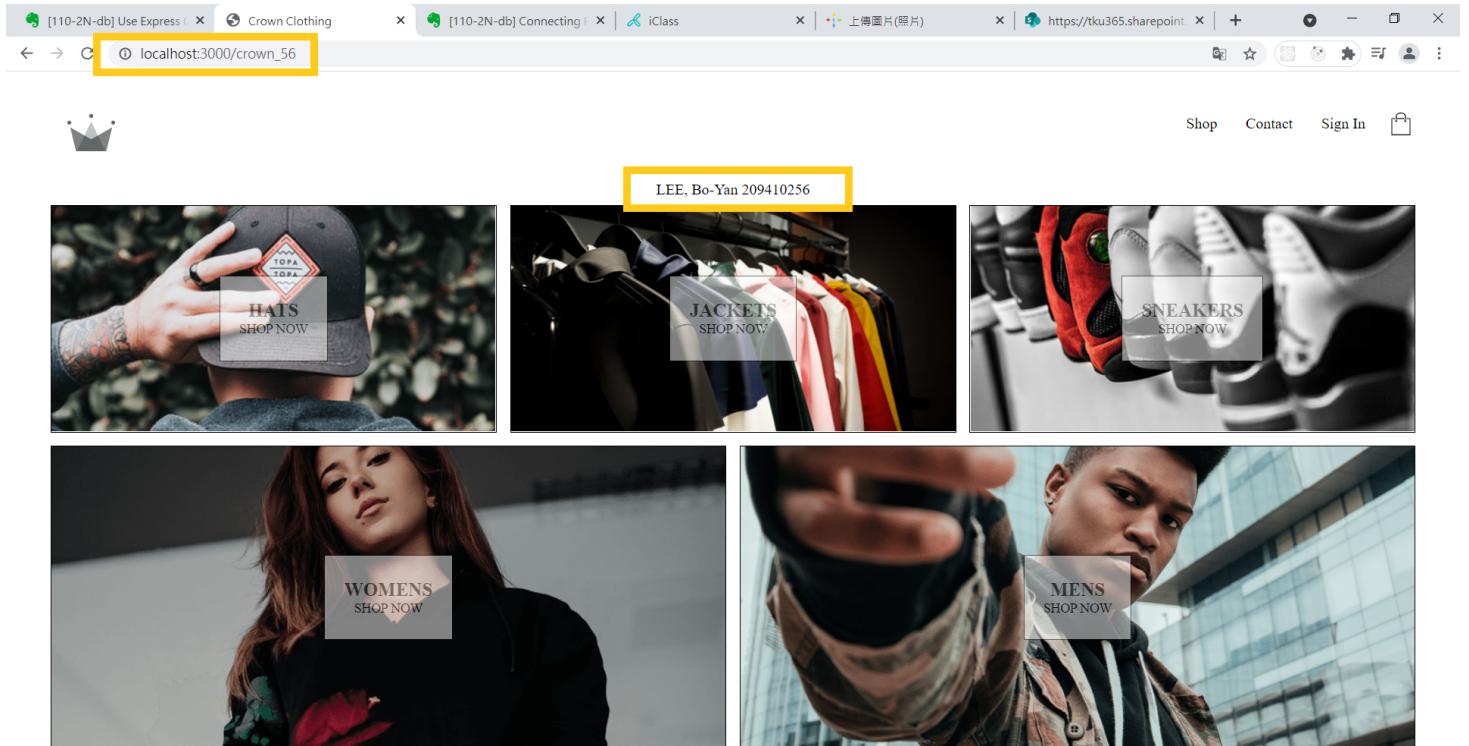
```
<body>
  <h1><%= title %></h1>
  <p>Welcome to <%= title %></p>
  <p></p>
  <h3>My name is <%= name %></h3>
  <h3>My student id is <%= id %></h3>
</body>
```

index.js

```
router.get('/', function (req, res, next) {
  res.render('index', {
    title: 'Express',
    name: 'LEE, Bo-Yan',
    id: '209410256',
  });
});
```

Q2

(local) 將 crown theme 裡面的 index.ejs 複製到 views/crown_xx.ejs，並能在 Chrome 中顯示路由 /crown_xx，如下圖 localhost:3000/crown_xx。標題要有學號及姓名，如 P1 的傳入方式。請繳交 Chrome 截圖，要有 URL
routes/crown_xx.js，相關重點以圖片呈現
views/crown_xx.ejs，相關重點以圖片呈現



```
BillyLee720, 2 weeks ago | 1 author (BillyLee720)
1 var express = require('express');
2 var router = express.Router();
3 /* GET home page. */
4 router.get('/', function (req, res, next) {
5   res.render('crown_56', {
6     title: 'Express',
7     name: 'LEE, Bo-Yan',
8     id: '209410256',
9   });
10 });
11 module.exports = router;
```

views/crown_56.ejs

```
PNG (read-only) 摄取3.PNG (read-only) 摄取.PNG (read-only) crown_56.md M crown_56.ejs X api_56.js app.js
views > crown_56.ejs > html > body > div.header
BillyLee720, 2 weeks ago | author (BillyLee720)
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Crown Clothing</title>
8   <link rel="stylesheet" href=".css/style.css">
9 </head>
10
11 <body>
12 >   <div class="header"> BillyLee720, 2 weeks ago • Initial commit ...
13 </div>
14
15 <div class="homepage">
16   <%= name %>
17   <%= id %>
18   <div class="directory-menu">
19     <div class="menu-item">
20       
21       <a href=".hats.html" class="content">
22         <h1 class="title">HATS</h1>
23         <span class="subtitle">SHOP NOW</span>
24       </a>
25     </div>
26   <div class="menu-item"> ...
27 </div>
28 <div class="menu-item"> ...
29 </div>
30 <div class="large menu-item"> ...
31 </div>
32 <div class="large menu-item"> ...
33 </div>
34 </div>
35 </div>
36
37 <script>
```

Q3

(local) 請將 crown_xx.ejs 裡面 category 五筆資料 PostgreSQL 中，資料庫名稱 crown_xx, table 名稱 category_xx。繳交

The screenshot shows the pgAdmin 4 interface. On the left, the database tree for 'd358o0lk32408p' is visible, with a yellow box highlighting the 'Tables (1)' section under the 'public' schema. A specific table, 'category_56', is selected and also highlighted with a yellow box. The main pane displays a query editor with the following SQL code:

```

1 SELECT * FROM public.category_56
2 ORDER BY id ASC

```

Below the query editor is a data output table with the following data:

	id [PK] integer	name character varying	size character varying	remote_url character varying	local_url character varying	link_url character varying
1	1	hats	[null]	https://i.ibb.co/cvpnL1/hats.png	/img/homepage/hats.png	/crown2_xx/shop2_xx/hats
2	2	jackets	[null]	https://i.ibb.co/px2tCc3/jackets.png	/img/homepage/jackets.png	/crown2_xx/shop2_xx/jackets
3	3	sneakers	[null]	https://i.ibb.co/0jqHpn/sneakers.png	/img/homepage/sneakers.png	/crown2_xx/shop2_xx/sneakers
4	4	womens	large	https://i.ibb.co/GCCdy8t/womens.png	/img/homepage/womens.png	/crown2_xx/shop2_xx/womens
5	5	mens	large	https://i.ibb.co/R70vBrQ/mens.png	/img/homepage/mens.png	/crown2_xx/shop2_xx/mens

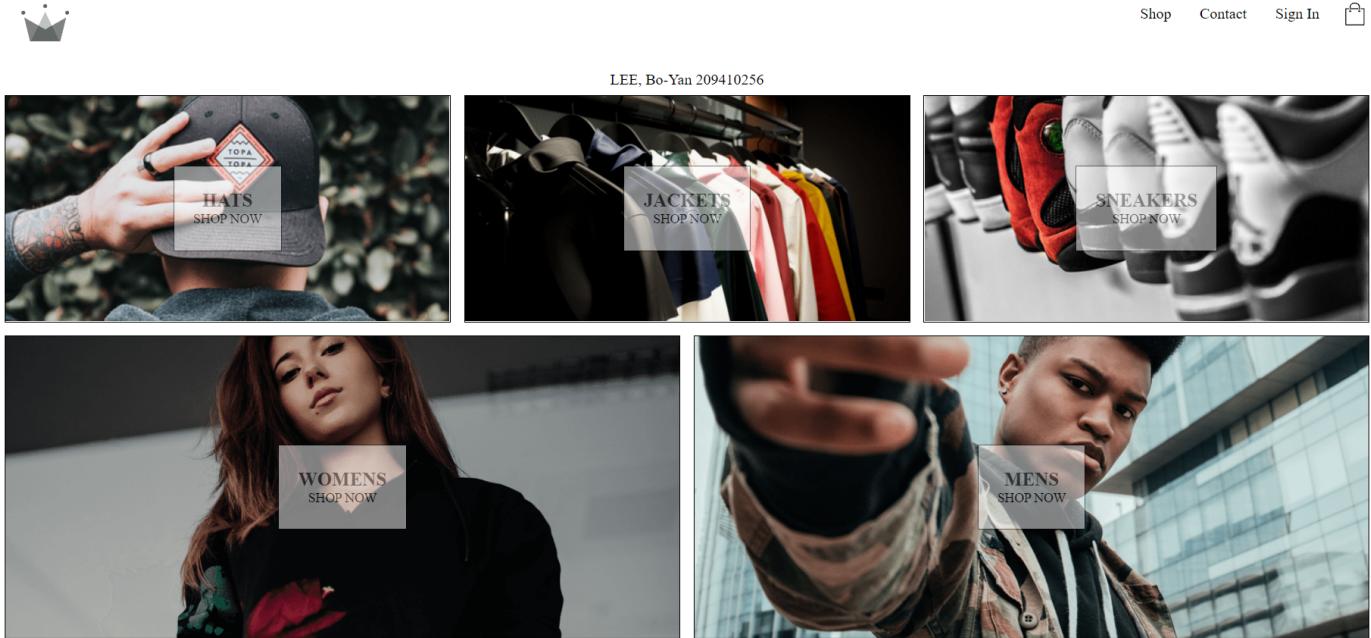
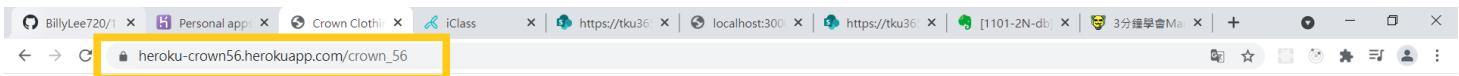
Q4

(heroku) 請將 P2 這一題的結果，在 heroku server 上顯示，路由是 /crown_xx。請繳交 Chrome 截圖 (要看到 URL，上面有 heroku app)

Github Repo URL

Heroku 畫面，要有 app 名稱

[github](#)



A screenshot of the Heroku dashboard. At the top, there's a navigation bar with a user icon, 'Personal', a dropdown arrow, and 'New'. Below it is a search bar with the placeholder 'Filter apps and pipelines'. The main area shows a card for the application 'heroku-crown56', which is a Node.js app running on heroku-20 in the United States. The card includes a small profile picture with 'zzz' on it, the app name, its status, and location. The URL 'heroku-crown56' is highlighted with a yellow box.

Q5

(heroku) 請將 P3 資料庫 category_xx 五筆資料，放到 Heroku Posgres 上。請繳交 Chrome 截圖 (要看到 URL，上面有 heroku database, category_xx, 和 5 筆資料)
Heroku 畫面，要看到 DATABASE_URL 資料
請提供 Heroku DATABASE_URL 資料，並將之拆分提供文字放入 code (``)中

The screenshot shows the pgAdmin 4 interface. On the left, the 'Browser' pane lists various databases and schemas, with 'category_56' highlighted. In the center, the 'Query Editor' pane contains a simple SELECT query:

```
1 SELECT * FROM public.category_56
2 ORDER BY id ASC
```

Below the query, the table data is shown in a grid:

	id [PK] integer	name character varying	size character varying	remote_url character varying	local_url character varying	link_url character varying
1	1	hats	[null]	https://i.ibb.co/cvptnL1/hats.png	/img/homepage/hats.png	/crown2_xx/shop2_xx/hats
2	2	jackets	[null]	https://i.ibb.co/px2tC3/jackets.png	/img/homepage/jackets.png	/crown2_xx/shop2_xx/jackets
3	3	sneakers	[null]	https://i.ibb.co/0jqHpn/sneakers.png	/img/homepage/sneakers.png	/crown2_xx/shop2_xx/sneakers
4	4	womens	large	https://i.ibb.co/GCCdy8t/womens.png	/img/homepage/womens.png	/crown2_xx/shop2_xx/womens
5	5	mens	large	https://i.ibb.co/R70vBrQ/men.png	/img/homepage/mens.png	/crown2_xx/shop2_xx/mens

Config Vars

Config vars change the way your app behaves. In addition to creating your own, some add-ons come with their own.

Config Vars

[Hide Config Vars](#)

DATABASE_URL	postgres://wukdsksghyaxel:b7bb9cf9f8095343aa9c560083f13870184ee174b29be0d68437e77551d77c86@ec2-52-87-123-108.com	Edit	X
NODE_ENV	production	Edit	X
KEY	VALUE	Add	

Database

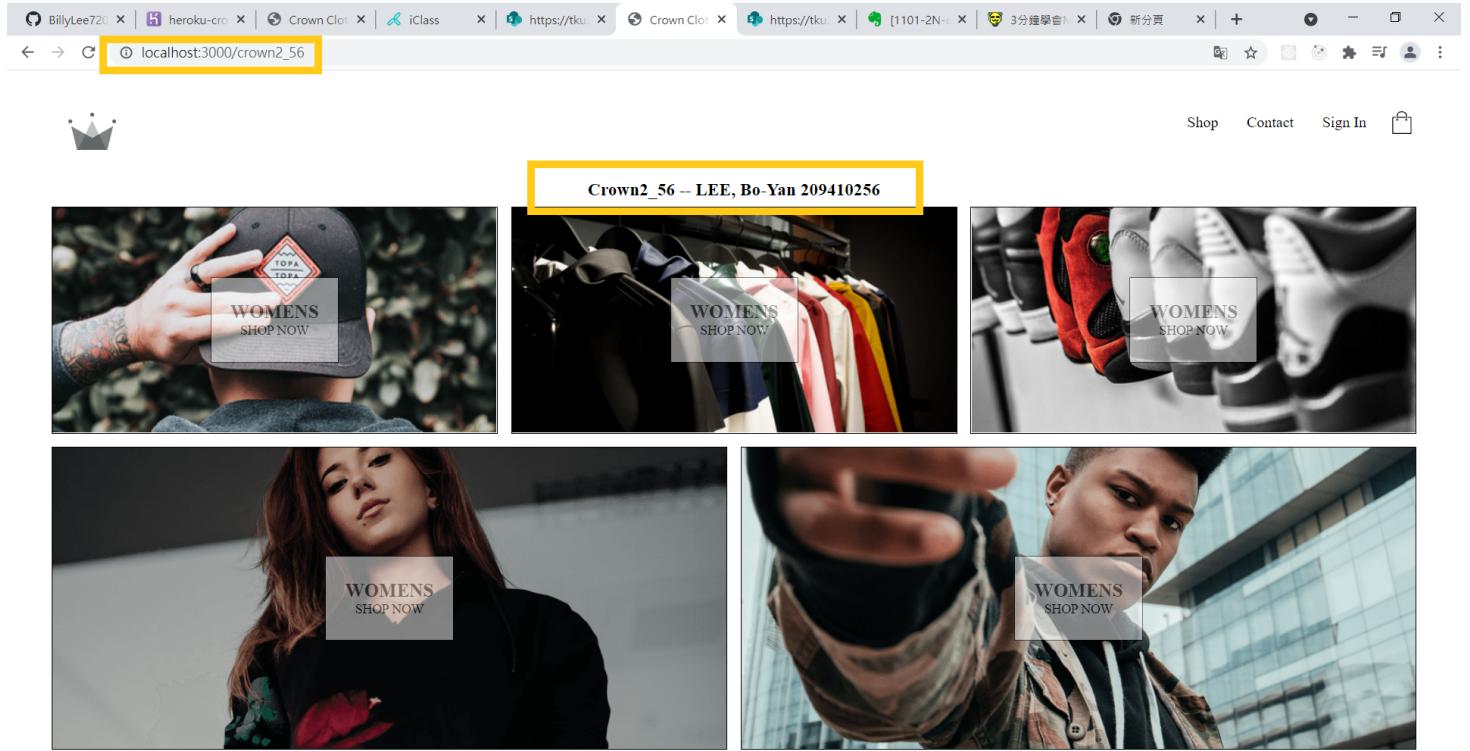
postgres://wukdsksghyaxel:b7bb9cf9f8095343aa9c560083f13870184ee174b29be0d68437e77551d77c86@ec2-52-87-123-108.com

Name : wukdsksghyaxel
password : b7bb9cf9f8095343aa9c560083f13870184ee174b29be0d68437e77551d77c86
Port:5432
Host : ec2-52-87-123-108.compute-1.amazonaws.com
Database : d358o0lk32408p

Q6

(local) 將 views/crown_xx.ejs 複製一份成 crown2_xx.ejs，能夠直接到 PostgreSQL server 取得 category 5 筆資料，套入到 crown2_xx.ejs 中。在 Chrome 中要顯示路由 /crown2_xx 。標題要有 Crown2_xx (from title) 學號 (from id) 及姓名(from name)。請繳交 Chrome 截圖，要有 URL
app.js , 相關重點以圖片標註呈現

routes/crown2_xx.js, 相關重點以圖片標註呈現呈現
views/crown2_xx.ejs , 相關重點以圖片標註呈現呈現



File Edit Selection View Go Run Terminal Help app.js - 1101-db-crown-209410256 - Visual Studio Code

EXPLORER

- 撷取5.PNG (read-only) md
- 撷取4.PNG (read-only) md
- 撷取3.PNG (read-only) md
- 撷取.PNG (read-only) md
- crown_56.md md
- crown2_56.js routes
- api_56.js routes
- crown2_56.ejs views
- app.js

1101-DB-CROWN-209410256

routes

- api_56.js
- crown_56.js
- crown2_56.js
- index.js
- users.js

utils

- database.js

views

- crown_56.ejs
- crown2_56.ejs
- error.ejs
- index.ejs

.gitignore

app.js

package-lock.json

package.json

OUTLINE

TIMELINE

NPM SCRIPTS

```
app.js > ...
BillyLee720, an hour ago | 1 author (BillyLee720)
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 const crown_56_Router = require('./routes/crown_56')
10 const crown2_56_Router = require('./routes/crown2_56')
11 const api_56_Router = require( ... );
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('/','');
26 app.use('/crown_56', crown_56_Router);
27 app.use('/crown2_56', crown2_56_Router);
28
29 app.use('/api_56', api_56_Router);    BillyLee720, an hour ago * add /api_56/category_56
30
31 // catch 404 and forward to error handler
32 app.use(function (req, res, next) {
33   next(createError(404));
34 });
35
36 // error handler
37 app.use(function (err, req, res, next) {
```

BillyLee720, an hour ago * add /api_56/category_56

Ln 29, Col 17 Spaces: 2 UTF-8 CRLF JavaScript Go Live Spell ✓

EXPLORER

OPEN EDITORS

routes > `crown2_56.js`

```

1 var express = require('express');
2 var router = express.Router();
3 const crown2Controller_56 = require('../controllers/crown2Controller_56')
4 /* GET home page. */
5 router.get('/', crown2Controller_56.getCategories);
6
7 module.exports = router;

```

1101-DB-CROWN-209410256

- heroku.PNG
- herokucrown.PNG
- pgadmindata.png
- tempsnip.png
- models
- `Category_56.js`
- node_modules
- public
- routes
- `api_56.js`
- `crown_56.js`
- `crown2_56.js` **selected**
- `index.js`
- `users.js`
- utils
- `database.js`
- views
- `crown_56.ejs`
- `crown2_56.ejs` **selected**
- `error.ejs`

OUTLINE

EXPLORER

OPEN EDITORS

views > `crown2_56.ejs`

```

1 !DOCTYPE html
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6     <title>Crown Clothing</title>
7     <link rel="stylesheet" href="./css/style.css" />
8   </head>
9
10  <body>
11    <div class="header">
12      <a href="/" class="logo-container">...
13      </a>
14      <div class="options">...
15      </div>
16    </div>
17
18    <div class="homepage">
19      <h3>%= title %> -- <%= name %> <%= id %></h3>
20
21    <div class="directory-menu">
22      <% data.forEach( item => { %>
23        <div class="%= item.size %> menu-item">
24          
25          <a href="/womens.html" class="content">
26            <h1 class="title">WOMENS</h1>
27            <span class="subtitle">SHOP NOW</span>
28          </a>
29        </div>
30      <% }) %>
31    </div>
32
33  </div>
34
35  <div class="category-menu">
36    <% data.forEach( item => { %>
37      <div class="category-item">
38        
39        <a href="#" class="content">
40          <h2 class="title">%= item.name %>
41          <span class="subtitle">%= item.size %>
42        </a>
43      </div>
44    <% }) %>
45  </div>
46
47  <div class="product-grid">
48    <% data.forEach( item => { %>
49      <div class="product-item">
50        
51        <a href="#" class="content">
52          <h3>%= item.name %>
53          <span class="price">%= item.price %>
54        </a>
55      </div>
56    <% }) %>
57  </div>
58
59  <div class="bottom-bar">
60    <div class="left">
61      <img alt="Cart icon" />
62      <span>0 items</span>
63    </div>
64    <div class="right">
65      <img alt="Search icon" />
66      <input type="text" placeholder="Search..." />
67    </div>
68  </div>
69
70  <div class="copyright">
71    <p>Copyright © Crown Clothing 2023. All rights reserved.</p>
72  </div>
73
74  </body>
75
76</html>

```

1101-DB-CROWN-209410256

- node_modules
- public
- routes
- `api_56.js`
- `crown_56.js`
- `crown2_56.js` **selected**
- `index.js`
- `users.js`
- utils
- `database.js`
- views
- `crown_56.ejs`
- `crown2_56.ejs` **selected**
- `error.ejs`
- `index.ejs`
- .gitignore
- `app.js`
- package-lock.json
- package.json

Q7

(heroku) 請將 P6 這一題的結果，在 heroku server 上顯示，路由是 /crown2_xx，category 資料必須要從 Heroku Posgres 上取得。請繳交
 Chrome 截圖 (要看到 URL，上面有 heroku app)
 Github Repo URL



Shop Contact Sign In



[Github](#)

Q8

(local) 實作 api 路由 /api_xx/category_xx，能夠直接到 PostgreSQL server 取得 category 5 筆資料並以 json 方式顯示。

請繳交

Chrome 截圖，要有 路由 /api_xx/category_xx

app.js , 相關重點以圖片標註呈現

routes/api_xx.js , 相關重點以圖片標註呈現呈現

controllers/apiCrown2Controller_xx.js , 相關重點以圖片標註呈現呈現

← → C

① localhost:3000/api_56/category_56

```
1 // 20211020221652
2 // http://localhost:3000/api_56/category_56
3
4 [
5   {
6     "id": 1,
7     "name": "hats",
8     "size": null,
9     "remote_url": "https://i.ibb.co/cvpntL1/hats.png",
10    "local_url": "/img/homepage/hats.png",
11    "link_url": "/crown2\_xx/shop2\_xx/hats"
12  },
13  {
14    "id": 2,
15    "name": "jackets",
16    "size": null,
17    "remote_url": "https://i.ibb.co/px2tCc3/jackets.png",
18    "local_url": "/img/homepage/jackets.png",
19    "link_url": "/crown2\_xx/shop2\_xx/jackets"
20  },
21  {
22    "id": 3,
23    "name": "sneakers",
24    "size": null,
25    "remote_url": "https://i.ibb.co/0jqHppn/sneakers.png",
26    "local_url": "/img/homepage/sneakers.png",
27    "link_url": "/crown2\_xx/shop2\_xx/sneakers"
28  },
29  {
30    "id": 4,
31    "name": "womens",
32    "size": "large",
33    "remote_url": "https://i.ibb.co/GCCdy8t/womens.png",
34    "local_url": "/img/homepage/womens.png",
35    "link_url": "/crown2\_xx/shop2\_xx/womens"
36  },
37  {
38    "id": 5,
39    "name": "mens",
40    "size": "large",
41    "remote_url": "https://i.ibb.co/R70vBrQ/mens.png",
42    "local_url": "/img/homepage/mens.png",
43    "link_url": "/crown2\_xx/shop2\_xx/mens"
44 }
```

OPEN EDITORS

- 撷取5.PNG (read-only) md
- 撷取4.PNG (read-only) md
- 撷取3.PNG (read-only) md
- 撷取.PNG (read-only) md
- crown_56.md md M
- localapi.PNG (read-only) md U
- api_56.js routes
- app.js**
- %% crown2_56.ejs views

1101-DB-CROWN-209410256

- node_modules
- public
- routes
 - api_56.js
 - crown_56.js
 - crown2_56.js
 - index.js
 - users.js
- utils
 - database.js
- views
 - %% crown_56.ejs
 - %% crown2_56.ejs
 - %% error.ejs
 - %% index.ejs
 - .gitignore
 - app.js
- package-lock.json
- package.json

OUTLINE

```

js app.js > ...
BillyLee720, an hour ago | 1 author (BillyLee720)
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 const crown_56_Router = require('./routes/crown_56')
10 const crown2_56_Router = require('./routes/crown2_56')
11 const api_56_Router = require('./routes/api_56');

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('/crown_56', crown_56_Router);
27 app.use('/crown2_56', crown2_56_Router);
28 app.use('/users', usersRouter);
29 app.use('/api_56', api_56_Router);

30

31 // catch 404 and forward to error handler
32 app.use(function (req, res, next) {
33   next(createError(404));
34 });

```

EXPLORER

OPEN EDITORS

- 撷取5.PNG (read-only) md
- 撷取4.PNG (read-only) md
- 撷取3.PNG (read-only) md
- 撷取.PNG (read-only) md
- crown_56.md md M
- localapi.PNG (read-only) md U
- api_56.js**
- app.js
- %% crown2_56.ejs views

1101-DB-CROWN-209410256

- pgadmindata.png
- tempsnip.png
- models
 - Category_56.js
- node_modules
- public
- routes
 - api_56.js**
 - crown_56.js
 - crown2_56.js
 - index.js
 - users.js
- utils
 - database.js
- views
 - %% crown_56.ejs
 - %% crown2_56.ejs
 - %% error.ejs
 - %% index.ejs

OUTLINE

TIMELINE

NPM SCRIPTS

```

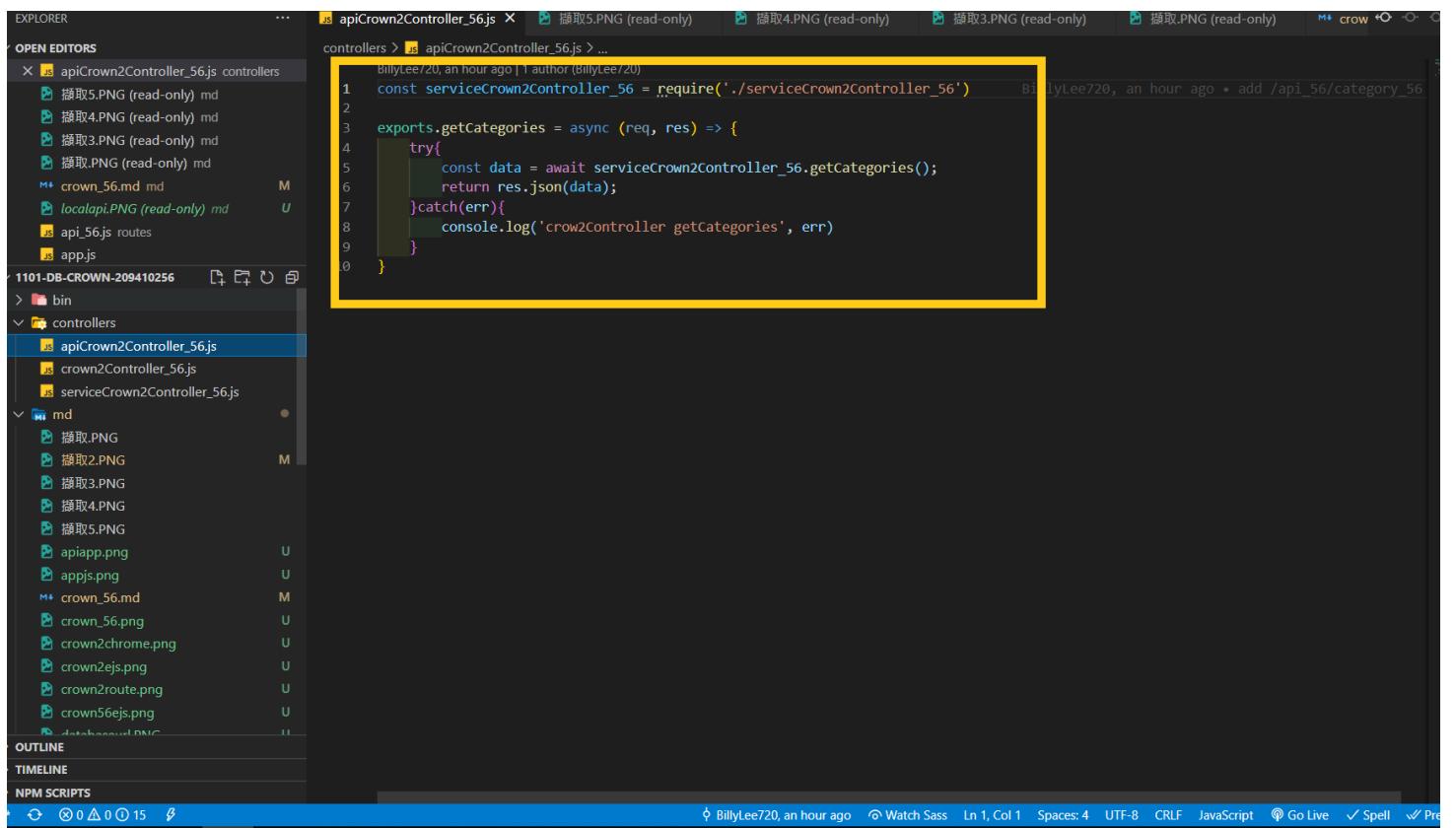
routes > api_56.js >
BillyLee720, an hour ago | 1 author (BillyLee720)
1 var express = require('express');
2 var router = express.Router();
3 const apiCrown2Controller_56 = require('../controllers/apiCrown2Controller_56');

4

5 /* GET home page. */
6 router.get(['/category_56', apiCrown2Controller_56.getCategories]);
7

8 module.exports = router;

```



```
const serviceCrown2Controller_56 = require('./serviceCrown2Controller_56')

exports.getCategoryes = async (req, res) => {
  try{
    const data = await serviceCrown2Controller_56.getCategoryes();
    return res.json(data);
  }catch(err){
    console.log('crown2Controller getCategoryes', err)
  }
}
```

Q9

(heroku) 請將 P8 這一題的結果，在 heroku server 上顯示，路由是 /api_xx/category_xx，category 資料必須要從 Heroku Posgres 上取得。請繳交

Chrome 截圖 (要看到 /api_xx/category_xx，上面有 heroku app)

Github Repo URL

← → C

heroku-crown56.herokuapp.com/api_56/category_56

```
1 // 20211020 22426
2 // https://heroku-crown56.herokuapp.com/api_56/category_56
3
4 [
5   {
6     "id": 1,
7     "name": "hats",
8     "size": null,
9     "remote_url": "https://i.ibb.co/cvpntL1/hats.png",
10    "local_url": "/img/homepage/hats.png",
11    "link_url": "/crown2_xx/shop2_xx/hats"
12  },
13  {
14    "id": 2,
15    "name": "jackets",
16    "size": null,
17    "remote_url": "https://i.ibb.co/px2tCc3/jackets.png",
18    "local_url": "/img/homepage/jackets.png",
19    "link_url": "/crown2_xx/shop2_xx/jackets"
20  },
21  {
22    "id": 3,
23    "name": "sneakers",
24    "size": null,
25    "remote_url": "https://i.ibb.co/0jqHpnP/sneakers.png",
26    "local_url": "/img/homepage/sneakers.png",
27    "link_url": "/crown2_xx/shop2_xx/sneakers"
28  },
29  {
30    "id": 4,
31    "name": "womens",
32    "size": "large",
33    "remote_url": "https://i.ibb.co/GCCdy8t/womens.png",
34    "local_url": "/img/homepage/womens.png",
35    "link_url": "/crown2_xx/shop2_xx/womens"
36  },
37  {
38    "id": 5,
39    "name": "mens",
40    "size": "large",
41    "remote_url": "https://i.ibb.co/R70vBr0/men.png",
42    "local_url": "/img/homepage/mens.png",
43    "link_url": "/crown2_xx/shop2_xx/mens"
44  }
45 ]
```

[GitHub](#)