

Your Github URL

Your Vercel URL

W14-P1: show first welcome page

=>

```

index.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><%= title %></title>
7   </head>
8   <body>
9     <h1><%= title %></h1>
10    <h2>Welcome to <%= title %></h2>
11    <h3><%= name %>, <%= id %></h3>
12    <footer>
13      <p>&copy; 2025 TKU</p>
14    </footer>
15  </body>
16</html>

```

```

server_14.js
1 import express from "express";
2 const app_14 = express();
3
4 app_14.use(express.static("public"));
5 app_14.set("view engine", "ejs");
6
7 app_14.use("/", (req, res, next) => {
8   res.render("index", {
9     title: "Express Server",
10    name: "Liang-Ting Lin",
11    id: "913410014"
12  });
13}
14
15 const port = process.env.PORT || 5000;
16
17 app_14.listen(port, () => {
18   console.log(`Server is running on port ${port}`);
19 });
20
21

```

4554eaf%09zero2005x%09Thu May 22 19:50:26 2025 +0800 W14-P1: show first welcome page

W14-P2: Show static products page

=> show render page with data passed into ejs page

```

1 import express from "express";
2
3 const app_14 = express();
4
5 app_14.use(express.static("public"));
6 app_14.set("view engine", "ejs");
7
8 app_14.use("/product_14/static", (req, res, next) => {
9   console.log("static");
10  res.render('product_14/static', {
11    title: "Get Product - Static",
12    name: "Liang-Ting Lin",
13    id: "913410014",
14  });
15});
16
17 app_14.use("/", (req, res, next) => {
18  res.render('index', {
19    title: "Express Server",
20    name: "Liang-Ting Lin",
21    id: "913410014",
22  });
23});
24
25 const port = process.env.PORT || 5000;
26
27 app_14.listen(port, () => {
28   console.log(`Server is running on port ${port}`);
29 });
30

```

Get Product - Static  
Liang-Ting Lin, 913410014

1-Emperor Bed  
\$21.99

=> show how the product-1.jpg can be accessed fro public directory as root directory

```

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     <link rel="stylesheet" href="/style.css" />
7   </head>
8   <body>
9     <div style="text-align: center; margin-top: 2rem">
10       <h2>%= title %</h2>
11       <h4 style="margin-top: -1.5rem">%= name %, <%= id %</h4>
12     </div>
13     <div class="products-container">
14       <div class="single-product">
15         
21       </div>
22       <div class="single-product">
23         
29     </div>
30   </div>
31   <div style="text-align: center; margin-top: 2rem">
32     <h3>Buy Now</h3>
33     <form>
34       <input type="button" value="Buy Now" />
35     </form>
36   </div>
37 </body>
38 </html>

```

Get Product - Static  
Liang-Ting Lin, 913410014

1-Emperor Bed  
\$21.99

c19ed13%09zero2005x%09Thu May 22 19:54:09 2025 +0800 W14-P2: Show static products page

W14-P3: Create database js\_demo\_xx, table product\_xx, and insert 12 data, write test code to get all products

=> Create database js\_demo\_xx, table product\_xx, and insert 12 data

The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer sidebar displays the database structure, including servers, databases (js\_demo\_14), and tables (product\_14). The main area shows a query editor with the following SQL query:

```
select * from public.product_14
```

The Data Output tab displays the results of the query, showing 12 rows of product data. The columns are id, title, price, category, img, and remote\_img. The data includes various furniture items like Emperor Bed, Accent Chair, High-Back Bench, Wooden Table, Dining Table, Entertainment Center, Albany Sectional, Sofa Set, Utopia Sofa, Modern Bookshelf, Albany Table, and Leather Sofa.

	<b>id</b> [PK] integer	<b>title</b> character varying (255)	<b>price</b> real	<b>category</b> character varying (255)	<b>img</b> text	<b>remote_img</b> text
1	1	Emperor Bed	21.99	Liddy	./images/product-1.jpg	https://loxlxwyirrqkjrruqbq.supabase.co/storage/v1/object/public/htchung-1n-14/p
2	2	Accent Chair	25.99	Caressa	./images/product-2.jpg	https://loxlxwyirrqkjrruqbq.supabase.co/storage/v1/object/public/htchung-1n-14/p
3	3	High-Back Bench	9.99	Ikea	./images/product-3.jpg	https://loxlxwyirrqkjrruqbq.supabase.co/storage/v1/object/public/htchung-1n-14/p
4	4	Wooden Table	19.99	Ikea	./images/product-4.jpg	https://loxlxwyirrqkjrruqbq.supabase.co/storage/v1/object/public/htchung-1n-14/p
5	5	Dining Table	69.99	Caressa	./images/product-5.jpg	https://loxlxwyirrqkjrruqbq.supabase.co/storage/v1/object/public/htchung-1n-14/p
6	6	Entertainment Center	25.99	Liddy	./images/product-6.jpg	https://loxlxwyirrqkjrruqbq.supabase.co/storage/v1/object/public/htchung-1n-14/p
7	7	Albany Sectional	10.99	Ikea	./images/product-7.jpg	https://loxlxwyirrqkjrruqbq.supabase.co/storage/v1/object/public/htchung-1n-14/p
8	8	Sofa Set	69.99	Liddy	./images/product-8.jpg	https://loxlxwyirrqkjrruqbq.supabase.co/storage/v1/object/public/htchung-1n-14/p
9	9	Utopia Sofa	69.99	Liddy	./images/product-9.jpg	https://loxlxwyirrqkjrruqbq.supabase.co/storage/v1/object/public/htchung-1n-14/p
10	10	Modern Bookshelf	8.99	Marcos	./images/product-10.jpg	https://loxlxwyirrqkjrruqbq.supabase.co/storage/v1/object/public/htchung-1n-14/p
11	11	Albany Table	79.99	Marcos	./images/product-11.jpg	https://loxlxwyirrqkjrruqbq.supabase.co/storage/v1/object/public/htchung-1n-14/p
12	12	Leather Sofa	39.95	Liddy	./images/product-12.jpg	https://loxlxwyirrqkjrruqbq.supabase.co/storage/v1/object/public/htchung-1n-14/p

At the bottom right of the pgAdmin interface, there is a message: "Table rows counted: 12".

=> connect js\_demo\_14 database

The screenshot shows both the VS Code terminal and the pgAdmin 4 interface. In the VS Code terminal, the user has run the command "node ./database.js" and is connected to the database:

```
demo > w14 fullstack_14 > server_14 > utils > JS database.js > ...
1 import pkg from "pg";
2 const { Pool } = pkg;
3
4 let pool = new Pool({
5   host: "localhost",
6   port: 5432,
7   user: "postgres",
8   password: "user",
9   database: "js_demo_14",
10 });
11
12 console.log("Connecting to the database...", pool.options);
13
14 export default pool;
15
```

In the pgAdmin 4 interface, the object explorer shows the database structure. The main query editor contains the same SQL query as before:

```
select * from public.product_14
```

The Data Output tab shows the same 12 rows of product data. At the bottom right of the pgAdmin interface, there is a message: "Table rows counted: 12".

=> get all product data

The screenshot shows the VS Code interface with the title bar "1132-1N-DEMO-14". The Explorer sidebar on the left shows a project structure for "1132-1N-DEMO-14" containing files like "demo", "w04\_basics\_14", "w05\_basics\_14", "w06\_prep\_test\_14", "w10\_mid1\_m3\_14", "w10\_product\_superbase\_14", "w11\_async\_js\_14", "w12\_ajax\_async\_14", "w13\_fetch\_async\_14", "w13\_product\_api\_14", "w14\_fullstack\_14", "client\_14", "server\_14", "node\_modules", "public", "routers/api", "JS apiProductRouter\_14.js", "utils", "data", "JS database.js", "JS db-test.js", "views", ".env", "package-lock.json", "package.json", "JS server\_14.js", "exams", and "mid1\_14". The terminal tab is active, displaying a Node.js script for connecting to a database:

```

import db from "./database.js";
const testDB = async () => {
  try {
    const results = await db.query("SELECT * FROM product_14");
    console.log("results", JSON.stringify(results.rows));
  } catch (error) {
    console.error("Error testing the database connection", error);
  }
};

```

The status bar at the bottom indicates "Ln 5, Col 64" and "Spaces: 4".

c53ddd%09zero2005x%09Thu May 22 20:58:47 2025 +0800 W14-P3: Create database js\_demo\_xx, table product\_xx, and insert 12 data, write test code to get all products

W14-P4: implement route /api/product\_xx to return json data

The screenshot shows the VS Code interface with the title bar "1132-1N-DEMO-14". The Explorer sidebar on the left shows a project structure for "1132-1N-DEMO-14" containing files like "demo", "w14\_fullst...", "server\_14", "node\_modules", "public", "routers/api", "utils", "data", "JS database.js", "JS db-test.js", "views", ".env", "package-lock.json", "package.json", "JS server\_14.js", "exams", "mid1\_14", "quiz1\_14", "rename.sh", "js", "md", "w01\_dom\_14", "w02\_dom\_14", "w03\_tictoatoe\_14", "w04\_basic\_14", "w05\_basic\_14", and "w06\_prep\_14". The terminal tab is active, displaying code for an Express.js application:

```

import express from "express";
import db from "../../utils/database.js";
const router = express.Router();

router.get("/", async (req, res, next) => {
  try {
    const results = await db.query("SELECT * FROM product_14");
    res.json(results.rows);
  } catch (error) {
    console.error("Error testing the database connection", error);
  }
});

export default router;

```

To the right, a browser window shows the URL "localhost:5000/api/product\_14" with a JSON response:

```

[{"id": 1, "title": "Emperor Bed", "price": 21.99, "category": "Liddy", "img": "./images/product-1.jpg", "remote_img": "https://loxlxwyirrqkjrrugbog.supabase.co/storage/v1/object/public/hutchung-in-14/product_14/product-1.jpg"}, {"id": 2, "title": "Accent Chair", "price": 25.99, "category": "Caressa", "img": "./images/product-2.jpg", "remote_img": "https://loxlxwyirrqkjrrugbog.supabase.co/storage/v1/object/public/hutchung-in-14/product_14/product-2.jpg"}, {"id": 3, "title": "High-Back Bench", "price": 9.99, "category": "Ikea", "img": "./images/product-3.jpg", "remote_img": "https://loxlxwyirrqkjrrugbog.supabase.co/storage/v1/object/public/hutchung-in-14/product_14/product-3.jpg"}, {"id": 4, "title": "Wooden Table", "price": 19.99, "category": "Ikea", "img": "./images/product-4.jpg", "remote_img": "https://loxlxwyirrqkjrrugbog.supabase.co/storage/v1/object/public/hutchung-in-14/product_14/product-4.jpg"}, {"id": 5, "title": "Dining Table", "price": 69.99, "category": "Caressa", "img": "./images/product-5.jpg", "remote_img": "https://loxlxwyirrqkjrrugbog.supabase.co/storage/v1/object/public/hutchung-in-14/product_14/product-5.jpg"}, {"id": 6, "title": "Entertainment Center", "price": 25.99, "category": "Liddy", "img": "./images/product-6.jpg", "remote_img": "https://loxlxwyirrqkjrrugbog.supabase.co/storage/v1/object/public/hutchung-in-14/product_14/product-6.jpg"}, {"id": 7, "title": "Albany Sectional", "price": 10.99, "category": "Ikea", "img": "./images/product-7.jpg", "remote_img": "https://loxlxwyirrqkjrrugbog.supabase.co/storage/v1/object/public/hutchung-in-14/product_14/product-7.jpg"}, {"id": 8, "title": "Sofa Set", "price": 69.99, "category": "Liddy", "img": "./images/product-8.jpg", "remote_img": "https://loxlxwyirrqkjrrugbog.supabase.co/storage/v1/object/public/hutchung-in-14/product_14/product-8.jpg"}, {"id": 9, "title": "Utopia Sofa", "price": 69.99, "category": "Ikea", "img": "./images/product-9.jpg", "remote_img": "https://loxlxwyirrqkjrrugbog.supabase.co/storage/v1/object/public/hutchung-in-14/product_14/product-9.jpg"}, {"id": 10, "title": "Modern Bookshelf", "price": 10.99, "category": "Marcos", "img": "./images/product-10.jpg", "remote_img": "https://loxlxwyirrqkjrrugbog.supabase.co/storage/v1/object/public/hutchung-in-14/product_14/product-10.jpg"}, {"id": 11, "title": "Albany Table", "price": 79.99, "category": "Marcos", "img": "./images/product-11.jpg", "remote_img": "https://loxlxwyirrqkjrrugbog.supabase.co/storage/v1/object/public/hutchung-in-14/product_14/product-11.jpg"}, {"id": 12, "title": "Leather Sofa", "price": 39.95, "category": "Liddy", "img": "./images/product-12.jpg", "remote_img": "https://loxlxwyirrqkjrrugbog.supabase.co/storage/v1/object/public/hutchung-in-14/product_14/product-12.jpg"}]

```

The status bar at the bottom indicates "Ln 19, Col 3" and "Spaces: 2".

=> whole json at /api/product\_xx

```

[{"id": 1, "title": "Emperor Bed", "price": 21.99, "category": "Liddy", "img": "./images/product-1.jpg", "remote_img": "https://loxlxwyirrakjrrugbog.supabase.co/storage/v1/object/public/htchung-1n-14/product_14/product-1.jpg"}, {"id": 2, "title": "Accent Chair", "price": 25.99, "category": "Caressa", "img": "./images/product-2.jpg", "remote_img": "https://loxlxwyirrakjrrugbog.supabase.co/storage/v1/object/public/htchung-1n-14/product_14/product-2.jpg"}, {"id": 3, "title": "High-Back Bench", "price": 9.99, "category": "Ikea", "img": "./images/product-3.jpg", "remote_img": "https://loxlxwyirrakjrrugbog.supabase.co/storage/v1/object/public/htchung-1n-14/product_14/product-3.jpg"}]
  
```

1133d36%09zero2005x%09Thu May 22 21:12:39 2025 +0800 W14-P4: implement route /api/product\_xx to return json data

## W14-log: git logs for W14

Commit	Author	Date	Message
1133d36	zero2005x	May 22, 2025	W14-P4: implement route /api/product_xx to return json data
c53dddc	zero2005x	May 22, 2025	W14-P3: Create database js_demo_xx, table product_xx, and insert 12 data, write test code to get all products
c19ed13	zero2005x	May 22, 2025	W14-P2: Show static products page
4554eaf	zero2005x	May 22, 2025	W14-P1: show first welcome page
a40dd23	zero2005x	May 15, 2025	W13-P4: Get Products from local json and from API