

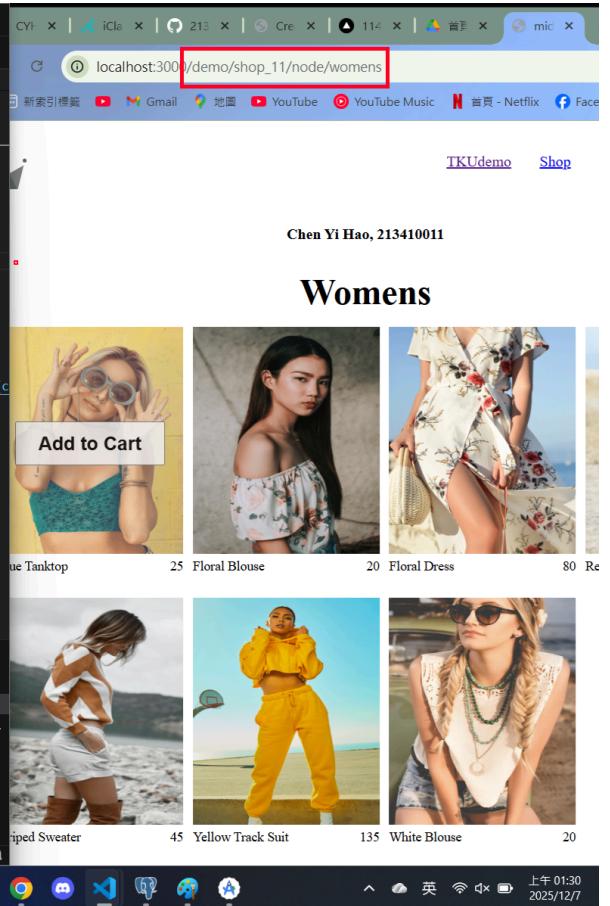
Github URL

Github URL for Vercel

Vercel URL

W12-P1: Refine code of /demo/shop_xx/node

=>



The screenshot shows a browser window displaying a women's clothing store page. The URL in the address bar is `localhost:3001/demo/shop_11/node/womens`. The page title is "Womens". There are several product cards: "Blue Tanktop", "Floral Blouse", "Floral Dress", "Striped Sweater", "Yellow Track Suit", and "White Blouse". An "Add to Cart" button is visible on the first product card.

Code editor view:

```
client_11 > src > app > demo > shop_11 > node > [category] > page.jsx > FetchShopByCategory_11
  1   'use client';
  2
  3   import { useState, useEffect } from 'react';
  4   import { useParams } from 'next/navigation';
  5
  6   import Wrapper from '../../../../../_wrapper/shop_11';
  7   import Product_11 from '../../../../../_components/Product_11';
  8
  9   const FetchShopByCategory_11 = () => [
 10     const [shop_11, setShop_11] = useState([[]]);
 11     const params = useParams();
 12     const category = params.category;
 13     console.log('category', category);
 14
 15     const fetchShopFromNode = async () => {
 16       try {
 17         const response = await fetch(`http://localhost:5000/api/shop_xx/${category}`);
 18         const data = await response.json();
 19         console.log('shop_11 data', data);
 20         if(data.length != 0) {
 21           setShop_11(data);
 22         }
 23       } catch (err) {
 24         console.log(err);
 25       }
 26     };
 27
 28     useEffect(() => {
 29       fetchShopFromNode();
 30     }, [category]);
 31   };
 32
 33   return (
 34     <div>
 35       <h1>Womens</h1>
 36       <div>
 37         <img alt="Product thumbnail" />
 38         <div>
 39           <h3>{product.name}</h3>
 40           <button>Add to Cart</button>
 41         </div>
 42       </div>
 43       <div>
 44         <img alt="Product thumbnail" />
 45         <div>
 46           <h3>{product.name}</h3>
 47           <img alt="Product thumbnail" />
 48         </div>
 49       </div>
 50       <div>
 51         <img alt="Product thumbnail" />
 52         <div>
 53           <h3>{product.name}</h3>
 54           <img alt="Product thumbnail" />
 55         </div>
 56       </div>
 57     </div>
 58   );
 59 }
```

Terminal view:

```
kevin@1vin MINIGM64:/~/w12_next_fullstack_11/client_11 (master)
$ git commit -m "first commit"
[master (root)] kevin@1vin:~/w12_next_fullstack_11/client_11
create mode 100644 src/app/(root)/page.js
create mode 100644 src/app/demo/shop_11/_components/NavbarShopSupa_11.jsx
create mode 100644 src/app/demo/shop_11/_components/Product_11.jsx
create mode 100644 src/app/demo/shop_11/_wrapper/Shop_11.jsx
create mode 100644 src/app/demo/shop_11/node/[category]/page.jsx
```

0d9de76 Hao Yi Chen

Wed Dec 3 20:44:28 2025 +0800

W12-P1: Refine code of /demo/shop_xx/no

W12-P2: Deploy code to Vercel

=> npm run build

The screenshot shows a Windows desktop environment with VS Code open. The terminal tab is active, displaying the command `npm run build` and its execution results:

```
kevin@Elvin MINGW64 /c/2N/w12_next_fullstack_11/client_11
$ npm run build
✓ collecting page data using 7 workers in 620.7ms
[baseline-browser-mapping] The data in this module is over two months old. To ensure accurate Baseline data, please update: `npm i baseline-browser-mapping@latest -D`
✓ generating static pages using 7 workers (8/8) in 503.6ms
✓ finalizing page optimization in 11.4ms
```

A red box highlights the route mapping output:

```
Route (app)
o /
o _/not-found
f /demo/shop_11/node
f /demo/shop_11/[category]
o /demo/shop_11/supa
f /demo/shop_11/supa/[category]
o /exams/midterm
o /exams/midterm/p1_11
o /exams/midterm/p2_11

o (Static) prerendered as static content
f (Dynamic) server-rendered on demand
```

The file tree on the left shows the project structure, including `client_11`, `src`, `app`, and various `.js` files like `Shop_11.jsx`, `layout.js`, `page.js`, `NavbarShopNode_11.jsx`, `NavbarShopSupa_11.jsx`, and `package.json`.

=> npm start

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

- Explorer View:** Shows the project structure under "W12_NEXT_FULLSTACK_11". A red box highlights the "build" folder.
- Code Editor:** Displays the contents of the "package.json" file. The "start" script is highlighted with a red box.

```
1 {
2   "name": "1141_2n_next_chenyihao_11",
3   "version": "0.1.0",
4   "private": true,
5   "scripts": {
6     "dev": "next dev",
7     "build": "next build",
8     "start": "next start"
9   },
10  "dependencies": {
11    "@supabase/supabase-js": "^2.84.0",
12    "next": "16.0.3",
13    "react": "19.2.0",
14    "react-dom": "19.2.0",
15    "sass": "1.54.2",
16    "styled-components": "^6.1.19"
17  },
18  "devDependencies": {
19    "@tailwindcss/postcss": "^4",
20    "babel-plugin-react-compiler": "1.0.0",
21    "tailwindcss": "^4"
22  }
23 }
```
- Terminal:** Shows the command "kevin@Elvin-M-NU64: /c/2N/w12_next_fullstack_11/client_11 \$ npm start" and its output:

```
> 1141_2n_next_chenyihao_11@0.1.0 start
> next start

  ▲ Next.js 16.0.3
  - Local:      http://localhost:3000
  - Network:    http://192.168.1.102:3000

✓ Starting...
✓ Ready in 241ms
category: women
```
- System Tray:** Shows the weather (17°C) and battery status.

=> Vercel, add environment variables

The screenshot shows the Vercel web interface for creating a new project. On the left, the 'Environment Variables' section is highlighted with a red box, showing two entries: `VITE_SUPABASE_URL` with value `https://twcehzwpddvdbmwyktxa.supabase.co` and `VITE_SUPABASE_KEY` with value `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpxVCJ9.eyJpc3MiOiJzdXBhYmFzZSIsInIzI`. On the right, a terminal window shows the deployment process for a project named 'w12_next_fullstack_11'. The logs indicate the project is being built and deployed to the specified URL.

Choose where you want to create the project and give it a name.

Vercel Team Project Name

Elvin Chen's projects Hobby 1141-2n-next-chenyihao-11

Framework Preset

Next.js

Root Directory

./ Edit

> Build and Output Settings

< Environment Variables

Key Value

VITE_SUPABASE_URL https://twcehzwpddvdbmwyktxa.supabase.co

VITE_SUPABASE_KEY eyJhbGciOiJIUzI1NiIsInR5cCI6IkpxVCJ9.eyJpc3MiOiJzdXBhYmFzZSIsInIzI

+ Add More Import.env or paste the .env contents above. Learn more

Deploy

w12_next_fullstack_11

NavarShopNode_11.jsx NavarShopSupa_11.jsx .env

E_SUPABASE_URL=https://twcehzwpddvdbmwyktxa.supabase.co
E_SUPABASE_KEY=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpxVCJ9.eyJpc3MiOiJzdXBhYmFzZSIsInIzI

OUTPUT DEBUG CONSOLE TERMINAL PORTS

git add origin https://github.com/213410011/1141-2n-next-chenyihao-11.
n MINGW64 /c/2N/w12_next_fullstack_11/client_11 (main)
-u origin main
g objects: 154, done.
bjects: 100% (154/154), done.
ression using up to 8 threads
g objects: 100% (144/144), done.
jects: 100% (154/154), 3.65 MiB | 2.01 MiB/s, done.
(delta 11), reused 0 (delta 0), pack-reused 0 (from 0)
olving deltas: 100% (11/11), done.
/github.com/213410011/1141-2n-next-chenyihao-11.git
anch] main -> main
in' set up to track 'origin/main'.
n MINGW64 /c/2N/w12_next_fullstack_11/client_11 (main)

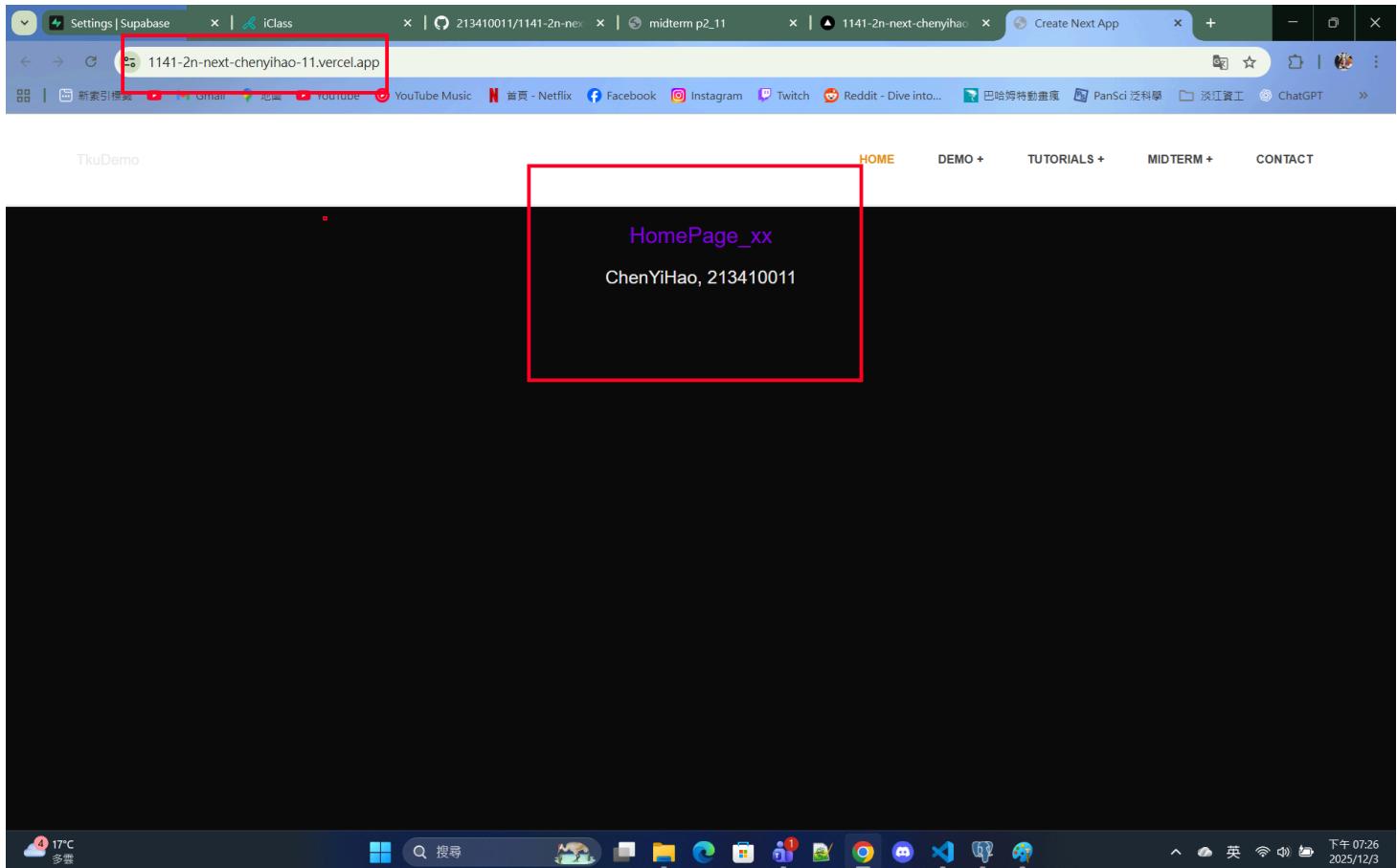
Ln 3, Col 1 Spaces: 4 UTF-8 CRLF { } Dotenv { } Go Live

17°C 多雲

搜尋

下午 07:24
2025/12/3

=> Vercel homepage success



=> Github repo and Vercel URL

[Github Next URL](#)

[Vercel Next URL](#)

The screenshot shows a browser window with multiple tabs open. The active tab is a GitHub repository page for '1141-2n-next-chenyihao-11'. The URL 'github.com/213410011/1141-2n-next-chenyihao-11' is highlighted with a red box. The repository details show a single commit from '213410011' labeled 'first commit'. The 'About' section on the right includes a link to '1141-2n-next-chenyihao-11.vercel.app', which is also highlighted with a red box. The browser's taskbar at the bottom shows various pinned icons.

=> Share to teacher and TA

htchung@gms.tku.edu.tw

sian-0018

The screenshot shows a GitHub repository settings page for a private repository. The left sidebar is titled 'General' and includes sections for Access (with 'Collaborators' highlighted), Code and automation, Security, and Integrations. The main area is titled 'Collaborators and teams' and shows a 'Direct access' section with 2 entities having access. Below this is a 'Manage access' section where two users are listed as pending invites: 'Hsingtai Chung' and 'sian-0018'. Both entries show they are awaiting response and have a trash icon next to them. A red box highlights the 'Collaborators' link in the sidebar and the list of pending invites.

03a4c47 Hao Yi Chen

Wed Dec 3 20:46:50 2025 +0800 W12-P2: Deploy code to Vercel

W12-P3: Implement /demo/shop2_xx/supabase to fetch products from Supabase

=> use SQL to get category2_xx data

The screenshot shows the Supabase dashboard interface. On the left, there's a sidebar with navigation links like 'SHARED', 'FAVORITES', and 'PRIVATE (2)'. Under 'PRIVATE (2)', two snippets are listed: 'category2_11, shop2_11' and 'Blog posts seed table'. The 'category2_11, shop2_11' snippet is selected and contains the following SQL code:

```
59 (29, 'White Blouse', 4, 20, '/images/midterm/womens/white-vest.png', 'https://i.ibb.co/qBcrsJg/white-vest.png'),  
60 (30, 'Camo Down Vest', 5, 325, '/images/midterm/mens/camo-vest.png', 'https://i.ibb.co/x150t3Y/camo-vest.png'),  
61 (31, 'Floral T-shirt', 5, 20, '/images/midterm/mens/floral-shirt.png', 'https://i.ibb.co/qM0750Z/floral-shirt.png'),  
62 (32, 'Black & White Longsleeve', 5, 25, '/images/midterm/mens/long-sleeve.png', 'https://i.ibb.co/55z32tw/long-sleeve.png'),  
63 (33, 'Pink T-shirt', 5, 25, '/images/midterm/mens/pink-shirt.png', 'https://i.ibb.co/RvnwBL8/pink-shirt.png'),  
64 (34, 'Jean Long Sleeve', 5, 40, '/images/midterm/mens/roll-up-jean-shirt.png', 'https://i.ibb.co/VpW4x5t/roll-up-jean-shirt.png'),  
65 (35, 'Burgundy T-shirt', 5, 25, '/images/midterm/mens/polka-dot-shirt.png', 'https://i.ibb.co/mh3VM1f/polka-dot-shirt.png');  
66  
67 select * from category2_11  
68  
69 select * from shop2_11
```

The results section displays a table with the following data:

cid	cname	size	image_url	remote_image_url	link_url
1	hats	NULL	/images/midterm/homepage/hats.png	https://i.ibb.co/cvptnL1/hats.png	/demo/shop_xx/node/hats
2	jackets	NULL	https://i.ibb.co/px2tCc3/jackets.png	/images/midterm/homepage/jackets.png	/demo/shop_xx/node/jackets
3	sneakers	NULL	https://i.ibb.co/0jqHpnP/sneakers.png	/images/midterm/homepage/sneakers.png	/demo/shop_xx/node/sneakers
4	womens	large	https://i.ibb.co/GCCdy8t/womens.png	/images/midterm/homepage/womens.png	/demo/shop_xx/node/womens
5	mens	large	https://i.ibb.co/R70vBrQ/mens.png	/images/midterm/homepage/mens.png	/demo/shop_xx/node/mens

At the bottom of the results table, it says '5 rows (Limited to only 100 rows)' and 'Limit results to: 100 rows'.

=> use SQL to get shop2_xx data

The screenshot shows the Supabase SQL Editor interface. On the left, there's a sidebar with sections like 'SQL Editor', 'Blog posts seed table', 'category2_11, shop2_11', 'PRIVATE', 'category2_11, shop2_11' (which is highlighted with a red box), and 'Blog posts seed table'. Below the sidebar is a search bar and a 'View running queries' button.

In the main area, there are two tabs: 'category2_11, shop2_11' and 'Blog posts seed table'. The 'category2_11, shop2_11' tab is active and contains the following SQL code:

```
56 (26, 'Red Dots Dress', 4, 80, '/images/midterm/womens/red-polka-dot-dress.png', 'https://i.ibb.co/NjBN1ph/red-polka-dot-dress.png'),  
57 (27, 'Striped Sweater', 4, 45, '/images/midterm/womens/striped-sweater.png', 'https://i.ibb.co/KmSKMbH/striped-sweater.png'),  
58 (28, 'Yellow Track Suit', 4, 135, '/images/midterm/womens/yellow-track-suit.png', 'https://i.ibb.co/vlcvwNf/yellow-track-suit.png'),  
59 (29, 'White Blouse', 4, 20, '/images/midterm/womens/white-vest.png', 'https://i.ibb.co/qBcrsJg/white-vest.png'),  
60 (30, 'Camo Down Vest', 5, 325, '/images/midterm/mens/camo-vest.png', 'https://i.ibb.co/xJS0T3Y/camo-vest.png'),  
61 (31, 'Floral T-shirt', 5, 20, '/images/midterm/mens/floral-shirt.png', 'https://i.ibb.co/qMQt5QZ/floral-shirt.png'),  
62 (32, 'Black & White Longsleeve', 5, 25, '/images/midterm/mens/long-sleeve.png', 'https://i.ibb.co/55z32tw/long-sleeve.png'),  
63 (33, 'Pink T-shirt', 5, 25, '/images/midterm/mens/pink-shirt.png', 'https://i.ibb.co/RwnvBL8/pink-shirt.png'),  
64 (34, 'Jean Long Sleeve', 5, 40, '/images/midterm/mens/roll-up-jean-shirt.png', 'https://i.ibb.co/VpW4x5t/roll-up-jean-shirt.png'),  
65 (35, 'Burgundy T-shirt', 5, 25, '/images/midterm/mens/polka-dot-shirt.png', 'https://i.ibb.co/mh3VM1f/polka-dot-shirt.png');  
66  
67 select * from category2_11  
68  
69 select * from shop2_11
```

The results of the second query ('category2_11, shop2_11') are displayed in a table below:

pid	pname	cat_id	price	img_url	remote_img_url
28	Yellow Track Suit	4	135	/images/midterm/womens/yellow-track-suit.png	https://i.ibb.co/vlcvwNf/yellow-track-suit.png
29	White Blouse	4	20	/images/midterm/womens/white-vest.png	https://i.ibb.co/qBcrsJg/white-vest.png
30	Camo Down Vest	5	325	/images/midterm/mens/camo-vest.png	https://i.ibb.co/xJS0T3Y/camo-vest.png
31	Floral T-shirt	5	20	/images/midterm/mens/floral-shirt.png	https://i.ibb.co/qMQt5QZ/floral-shirt.png
32	Black & White Longsleeve	5	25	/images/midterm/mens/long-sleeve.png	https://i.ibb.co/55z32tw/long-sleeve.png
33	Pink T-shirt	5	25	/images/midterm/mens/pink-shirt.png	https://i.ibb.co/RwnvBL8/pink-shirt.png
34	Jean Long Sleeve	5	40	/images/midterm/mens/roll-up-jean-shirt.png	https://i.ibb.co/VpW4x5t/roll-up-jean-shirt.png
35	Burgundy T-shirt	5	25	/images/midterm/mens/polka-dot-shirt.png	https://i.ibb.co/mh3VM1f/polka-dot-shirt.png

At the bottom of the interface, there are system status icons (CPU, RAM, Disk, Network) and a system tray with a weather forecast (17°C), a search bar, and system controls (language, battery, date/time).

=> set foreign key

The screenshot shows the Supabase Table Editor interface for the 'shop2_11' table. On the left, the table structure is displayed with columns: pid (int4), pname (varchar), cat_id (int4), price (float), img_url (text), and remote_img_url (text). A red box highlights the 'Foreign keys' section on the right. In this section, a modal window titled 'Add foreign key relationship to shop2_11' is open. It asks to 'Select a table to reference to' and shows 'category2_11' selected. Below that, it says 'Select columns from public.category2_11 to reference to public.shop2_11' and shows 'cat_id' mapped to 'cid'. A second red box highlights the 'Action if referenced row is updated' dropdown, which is set to 'Cascade'. A third red box highlights the 'Action if referenced row is removed' dropdown, which is set to 'Set NULL'. The status bar at the bottom indicates the date and time as '2025/12/7 上午 01:49'.

=> set RLS policies for public access for all CRUD

The screenshot shows the Supabase dashboard with a modal window titled "Create a new Row Level Security policy". The modal contains the following fields:

- Policy Name:** "public access for all CRUD operations"
- Table on clause:** "public.shop2_11"
- Policy Behavior as clause:** "Permissive"
- Policy Command for clause:** "ALL" (selected)
- Target Roles to clause:** "Defaults to all (public) roles if none selected"

Below these fields is a code editor showing the SQL query generated by the policy configuration:

```
USE OPTIONS ABOVE TO EDIT
1 create policy "public access for all CRUD operation"
2 on "public"."shop2_11"
3 as PERMISSIVE
4 for ALL
5 to public
6 using (
7   true
8 );
```

The word "true" in the code editor is highlighted with a red box. At the bottom of the modal are "Cancel" and "Save policy" buttons.

To the right of the modal, there is a sidebar titled "Templates" containing several pre-defined RLS policies:

- Enable read access for all users**: This policy gives read access to your table for all users via the SELECT operation.
- Enable insert for authenticated users only**: This policy gives insert access to your table for all authenticated users only.
- Enable delete for users based on user_id**: This policy assumes that your table has a column "user_id", and allows users to delete rows which the "user_id" column matches their ID.
- Enable insert for users based on user_id**: This policy assumes that your table has a column "user_id", and allows users to insert rows which the "user_id" column matches their ID.
- Policy with table joins**: Query across tables to build more advanced RLS rules. Assuming 2 tables called teams and members, you can query both tables in the

The browser's address bar shows the URL: supabase.com/dashboard/project/twcehwzwpddvdbmwyktxa/auth/policies?new=true. The system tray at the bottom indicates the date as 2025/12/7 and the time as 上午 01:58.

=> Chrome -- local

Screenshot of a local development environment in Chrome showing a shopping application.

The URL in the address bar is `localhost:3001/demo/shop_11/node/womens`.

The page title is "Womens".

The DevTools console shows the following data structure:

```
[HMR] connected forward-logs-shared.ts:95
category womens forward-logs-shared.ts:95
forward-logs-shared.ts:95
shop_11 data forward-logs-shared.ts:95
Array(7)
  0: {cid: 4, cname: 'womens', size: 'large', image_ur...
  1: {cid: 4, cname: 'womens', size: 'large', image_ur...
  2: {cid: 4, cname: 'womens', size: 'large', image_ur...
  3: {cid: 4, cname: 'womens', size: 'large', image_ur...
  4: {cid: 4, cname: 'womens', size: 'large', image_ur...
  5: {cid: 4, cname: 'womens', size: 'large', image_ur...
  6: {cid: 4, cname: 'womens', size: 'large', image_ur...
length: 7

```

The page displays a grid of women's clothing items:

- Blue Tanktop (Thumbnail)
- 25 Floral Blouse (Thumbnail)
- 20 Floral Dress (Thumbnail)
- 80 Red Dots Dress (Thumbnail)
- Striped Sweater (Thumbnail)
- 45 Yellow Track Suit (Thumbnail)
- 135 White Blouse (Thumbnail)

The bottom status bar shows the date and time: 上午 01:52 2025/12/7.

=> Chrome -- Vercel

The screenshot shows a Chrome browser window with multiple tabs open. The active tab displays a shopping website for women's clothing. The URL in the address bar is highlighted with a red box. The page itself has a header with the text "Chen Yi Hao, 213410011". Below the header is a title "Womens". A grid of five product cards is shown, each featuring a model wearing a different outfit. The products are: "Blue Tanktop" (25), "Floral Blouse" (20), "Floral Dress" (80), "Red Dots Dress" (80), "Striped Sweater" (45), "Yellow Track Suit" (135), and "White Blouse" (20). The bottom of the screen shows the Windows taskbar with various icons and the date/time.