

[Github URL](#)

[Github URL for Vercel](#)

[Vercel URL](#)

Video: W15-P1: Create tables category2_xx, shop2_xx, user2_xx, cart2_xx, and put 5 products into a cart for the user of your id

=> pgAdmin4, show SQL command to get the needed info

The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer shows the database structure. The 'Tables (4)' folder is expanded, showing 'cart2_02', 'category2_02', 'shop2_02', and 'user2_02'. The 'Columns (4)' folder for 'user2_02' is also expanded, showing 'uid', 'uname', 'email', and 'password'. The main pane shows a SQL query in the 'Query' tab. The query is:

```
select C.user_id, U.uname as user_name, C.product_id, S.pname as product_name, C.quantity, S.price,
from cart2_02 as C, user2_02 as U, shop2_02 as S
where U.uid = '213410102' and C.user_id = U.uid and C.product_id = S.pid
```

The 'Data Output' tab shows the results of the query. The results are displayed in a table with 5 rows and 7 columns. The columns are: user_id, user_name, product_id, product_name, quantity, price, and img_url. The data is as follows:

	user_id	user_name	product_id	product_name	quantity	price	img_url
1	213410102	李國蘋	1	Brown Brim	2	25	/images/midterm/hats/brown-brim.png
2	213410102	李國蘋	10	Black Jean Shearling	3	125	/images/midterm/jackets/black-shearling.png
3	213410102	李國蘋	15	Adidas NMD	1	220	/images/midterm/sneakers/adidas-nmd.png
4	213410102	李國蘋	23	Blue Tanktop	2	25	/images/midterm/womens/blue-tank.png
5	213410102	李國蘋	30	Camo Down Vest	2	325	/images/midterm/mens/camo-vest.png

The status bar at the bottom indicates 'Total rows: 5' and 'Query complete 00:00:00.061'. The system tray at the bottom shows the date and time: '日圖/新臺幣 -0.64%' and '下午 03:17 2025/12/28'.

=> sql code

The screenshot shows a VS Code editor with a dark theme. The Explorer sidebar on the left displays a file tree for a project named '1141_2N_NEXT_APPLE_02'. The 'data' folder is expanded, and a red box highlights the files 'blog_02.sql', 'cart2_02.sql', 'category2_02.sql', 'shop2_02.sql', and 'user2_02.sql'. The 'cart2_02.sql' file is selected, and its content is visible in the main editor. The code consists of a PostgreSQL table creation statement for 'cart2_02' and an 'INSERT INTO' statement with 10 rows of data. Below the insert, a 'select' query is partially visible, filtering by 'U.uid = '213410102'' and joining tables 'cart2_02', 'user2_02', and 'shop2_02'. The status bar at the bottom indicates the current position is 'Ln 25, Col 73' and the file encoding is 'UTF-8'.

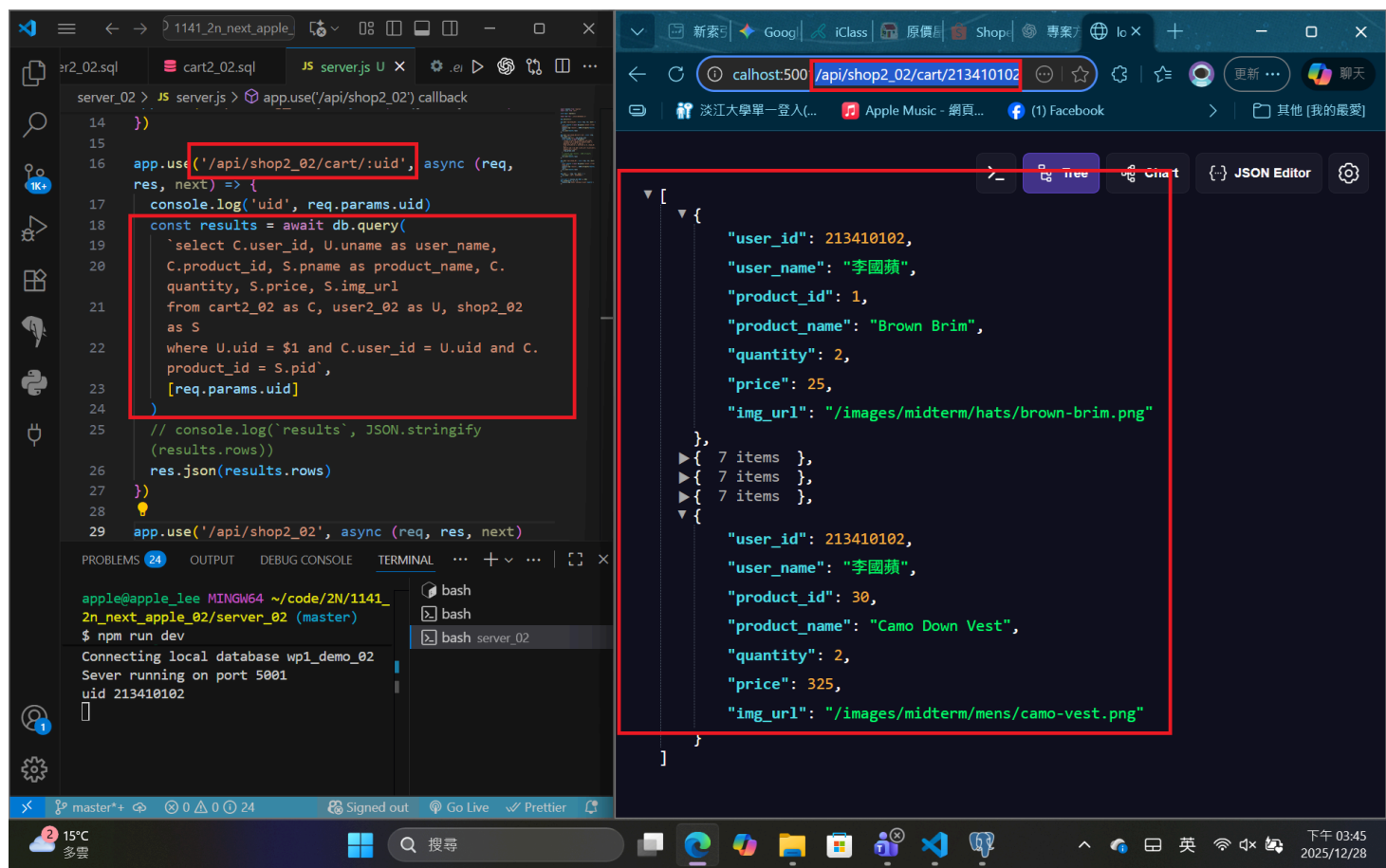
```
1 CREATE TABLE cart2_02 (  
2   cid int NOT NULL PRIMARY KEY,  
3   user_id int,  
4   product_id int,  
5   quantity int,  
6   total int DEFAULT 0,  
7   added_at timestamp DEFAULT CURRENT_TIMESTAMP  
8 );  
9  
10 INSERT INTO cart2_02 (cid, user_id, product_id, quantity, total, added_at)  
11 VALUES  
12 (1, 213410102, 1, 2, 0, '2025-12-24 18:50:50'),  
13 (2, 213410102, 10, 3, 0, '2025-12-24 18:50:50'),  
14 (3, 213410102, 15, 1, 0, '2025-12-24 18:50:50'),  
15 (4, 213410102, 23, 2, 0, '2025-12-24 18:50:50'),  
16 (5, 213410102, 30, 2, 0, '2025-12-24 18:50:50'),  
17 (6, 1, 1, 2, 0, '2025-12-24 18:50:50'),  
18 (7, 2, 10, 3, 0, '2025-12-24 18:50:50'),  
19 (8, 3, 15, 1, 0, '2025-12-24 18:50:50'),  
20 (9, 4, 23, 2, 0, '2025-12-24 18:50:50'),  
21 (10, 4, 30, 2, 0, '2025-12-24 18:50:50')  
22  
23 select C.user_id, U.username as user_name, C.product_id, S.pname as product_name, C.quantity, S.price, S.img_url  
24 from cart2_02 as C, user2_02 as U, shop2_02 as S  
25 where U.uid = '213410102' and C.user_id = U.uid and C.product_id = S.pid
```

f43d4bd apple550678

2025-12-28 15:20:35 +0800

Video: W15-P1: Create tables category2_

Video: W15-P2: Implement route /api/shop_xx/cart/:uid to get the info as the SQL in W15-P1



3286bd9 apple550678

2025-12-28 15:48:13 +0800

Video: W15-P2: Implement route /api/shop

Video: W15-P3: Use localStorage

=> Implement setLocalStorage

The screenshot displays a development environment with a code editor on the left and a browser window on the right.

Code Editor (Left): The file `page.jsx` is open, showing the `GroceryPage_02` component. The `setLocalStorage` function is defined as follows:

```
const setLocalStorage = (items) => {
  localStorage.setItem('list', JSON.stringify(items))
}
```

The `addItem` function calls `setLocalStorage(newItems)` to save the updated list.

Browser Window (Right): The browser is at `localhost:3000/demo/grocery_02`. The application, titled "Grocery Bud", shows a form with an "Add Item" button and a list of items: "Apple" and "Eggs", each with a "Delete" button.

The browser's developer tools (Application tab) show the state of the application. The `list` property in the `localStorage` storage is highlighted, showing the following JSON array:

```
[{"name": "Apple", completed: false, id: "4zDH4uhM0DUu7j-o2MUuJ"}, {"name": "Eggs", completed: false, id: "1m_zqlfoe_7iDANibAlYV"}]
```

=> Implement getLocalStorage

The screenshot displays a development environment with three main components:

- Code Editor (Left):** Shows the implementation of `getLocalStorage` and `setLocalStorage` functions in `page.jsx`.

```
11 const getLocalStorage = () => {  
12   if (typeof window !== 'undefined') {  
13     let list = localStorage.getItem('list')  
14     if (list) {  
15       list = JSON.parse(list)  
16     } else {  
17       list = []  
18     }  
19     return list  
20   }  
21   return [] // Return default for server-side  
22 }  
  
24 const setLocalStorage = (items) => {  
25   localStorage.setItem('list', JSON.stringify  
26     (items))  
27 }  
  
29 const GroceryPage_02 = () => {  
30   const [items, setItems] = useState([])  
31  
32   useEffect(() => {  
33     const storedList = getLocalStorage()  
34     if (storedList.length > 0) {  
35       setItems(storedList)  
36     }  
37   }, [])  
38 }
```
- Web Application (Middle):** A browser window showing a "Grocery Bud" interface. It includes an "Add Item" button and a list of items: "Apple" and "Eggs", each with a "Delete" button.
- Browser Storage (Right):** The browser's storage inspector shows the data stored in `localStorage`.

Key	Value
list	[{"name": "Apple", "completed": false}, {"name": "Eggs", "completed": false}]

59bba84 apple550678

2025-12-28 16:44:18 +0800

Video: W15-P3: Use localStorage