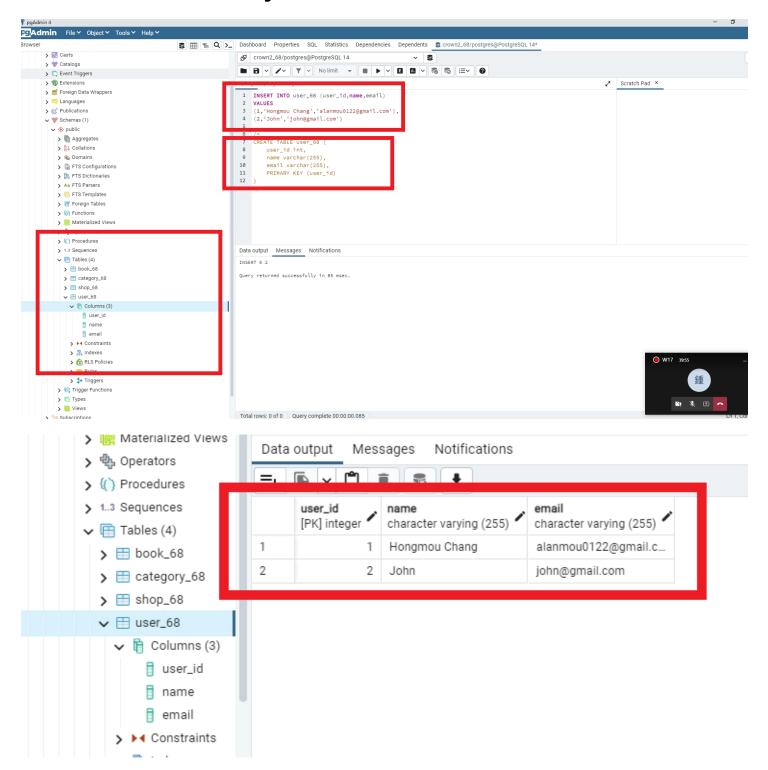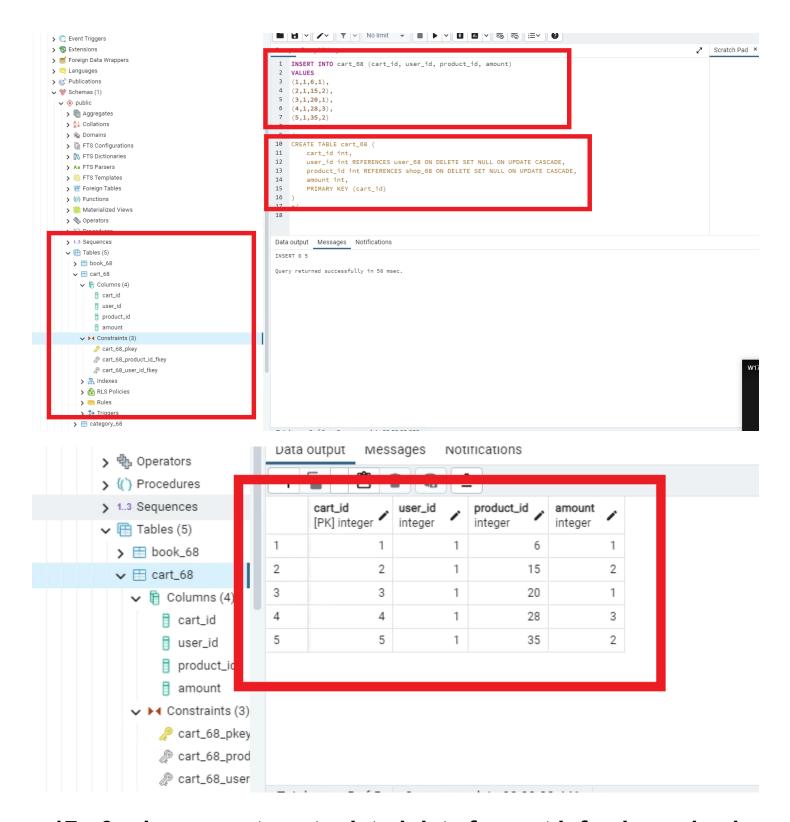# w17-p1: create user_xx in postgreSQL server, enter two data, the first one is about your

w17-p2: create cart_xx in postgreSQL server, enter five data, one for each category

```
1  INSERT INTO cart_68 (cart_id, user_id, product_id, amount)
2  VALUES
3  (1,1,6,1),
4  (2,1,15,2),
5  (3,1,20,1),
6  (4,1,28,3),
7  (5,1,35,2)

10  CREATE TABLE cart_68 (
11      cart_id int,
12      user_id int REFERENCES user_68 ON DELETE SET NULL ON UPDATE CASCADE,
13      product_id int REFERENCES shop_68 ON DELETE SET NULL ON UPDATE CASCADE,
14      amount int,
15      PRIMARY KEY (cart_id)
16  )
17  */
18
```

Data output Messages Notifications

INSERT 0 5

Query returned successfully in 50 msec.

cart_id [PK] integer	user_id integer	product_id integer	amount integer
1	1	6	1
2	1	15	2
3	1	20	1
4	1	28	3
5	1	35	2

w17-p3: give query to get related data for cart info shown in ejs

Query Query History

```
1  SELECT cart_id, U.name AS user_name,S.name AS product_name,price,amount
2  FROM cart_68 as C, user_68 AS U, shop_68 AS S
3  WHERE C.user_id = U.user_id AND C.product_id = S.id
```

Data output   Messages   Notifications

	cart_id integer	user_name character varying (255)	product_name character varying (45)	price real	amount integer
1	1	Hongmou Chang	Brown Brim	25	1
2	2	Hongmou Chang	Black Jean Shearling	125	2
3	3	Hongmou Chang	Adidas NMD	220	1
4	4	Hongmou Chang	Blue Tanktop	25	3
5	5	Hongmou Chang	Camo Down Vest	325	2

w17-p4: insert data in order to violate the key constraints, foreign key constraints

public
> Aggregates
> Collations
> Domains
> FTS Configurations
> FTS Dictionaries
> Aa FTS Parsers
> FTS Templates
> Foreign Tables
> Functions
> Materialized Views
> Operators
> Procedures
> 1..3 Sequences
∨ Tables (5)
  > book_68
  ∨ cart_68
    ∨ Columns (4)
      cart_id
      user_id
      product_id
      amount
    ∨ Constraints (3)
      cart_68_pkey
      cart_68_product_id_fkey
      cart_68_user_id_fkey

Query   Query History

```sql
1   INSERT INTO cart_68 (cart_id, user_id, product_id, amount)
2   VALUES
3   (1,3,6,1)
4
5   /*
6   INSERT INTO cart_68 (cart_id, user_id, product_id, amount)
7   VALUES
8   (1,1,6,1),
9   (2,1,15,2),
10  (3,1,20,1),
11  (4,1,28,3),
12  (5,1,35,2)
13  */
14
15  /*
16  CREATE TABLE cart_68 (
```

Data output   Messages   Notifications

ERROR: duplicate key value violates unique constraint "cart_68_pkey"
DETAIL: Key (cart_id)=(1) already exists.
SQL state: 23505

Database Navigation Tree

- ⌄ ◈ public
  - › Aggregates
  - › Collations
  - › Domains
  - › FTS Configurations
  - › FTS Dictionaries
  - › Aa FTS Parsers
  - › FTS Templates
  - › Foreign Tables
  - › Functions
  - › Materialized Views
  - › Operators
  - › Procedures
  - › 1..3 Sequences
  - ⌄ Tables (5)
    - › book_68
    - ⌄ cart_68
      - ⌄ Columns (4)
        - cart_id
        - user_id
        - product_id
        - amount
      - ⌄ Constraints (3)
        - cart_68_pkey
        - cart_68_product_id_fkey
        - cart_68_user_id_fkey

Query

```
1   INSERT INTO cart_68 (cart_id, user_id, product_id, amount)
2   VALUES
3   (6,3,6,1)
4
5   /*
6   INSERT INTO cart_68 (cart_id, user_id, product_id, amount)
7   VALUES
8   (1,1,6,1),
9   (2,1,15,2),
10  (3,1,20,1),
11  (4,1,28,3),
12  (5,1,35,2)
13  */
14
15  /*
16  CREATE TABLE cart_68 (
```

Data output   Messages   Notifications

```
ERROR:  insert or update on table "cart_68" violates foreign key constraint
"cart_68_user_id_fkey"
DETAIL:  Key (user_id)=(3) is not present in table "user_68".
SQL state: 23503
```

```
1  INSERT INTO cart_68 (cart_id, user_id, product_id, amount)
2  VALUES
3  (6,1,5,1)
4
5  /*
6  INSERT INTO cart_68 (cart_id, user_id, product_id, amount)
7  VALUES
8  (1,1,6,1),
9  (2,1,15,2),
10 (3,1,20,1),
11 (4,1,28,3),
12 (5,1,35,2)
13 */
14
15 /*
16 CREATE TABLE cart_68 (
```

Data output   Messages   Notifications

ERROR: insert or update on table "cart_68" violates foreign key constraint
"cart_68_product_id_fkey"
DETAIL: Key (product_id)=(5) is not present in table "shop_68".
SQL state: 23503

w17-p5.1: test ON UPDATE CASCASE

- > () Procedures
- > 1..3 Sequences
- ∨ ▦ Tables (5)
  - > ▦ book_68
  - ∨ ▦ cart_68
    - ∨ 🗄 Columns (4)
      - 🗄 cart_id
      - 🗄 user_id
      - 🗄 product_id
      - 🗄 amount
    - ∨ ▶◀ Constraints (3)
      - 🔑 cart_68_pkey
      - 🔑 cart_68_product_id_fkey
      - 🔑 cart_68_user_id_fkey
    - > 🔠 Indexes
    - > 🔒 RLS Policies
    - > 🈯 Rules
    - > ⥇ Triggers
  - > ▦ category_68
  - > ▦ shop_68
  - > ▦ user_68
- > 📲 Trigger Functions
- > 🗔 Types
- > 🔳 Views
- > 🥱 Subscriptions

Data output Messages Notifications

	user_id [PK] integer	name character varying (255)	email character varying (255)
1	1	Hongmou Chang	alanmou0122@gmail.c…
2	**5**	john	john@gmail.com

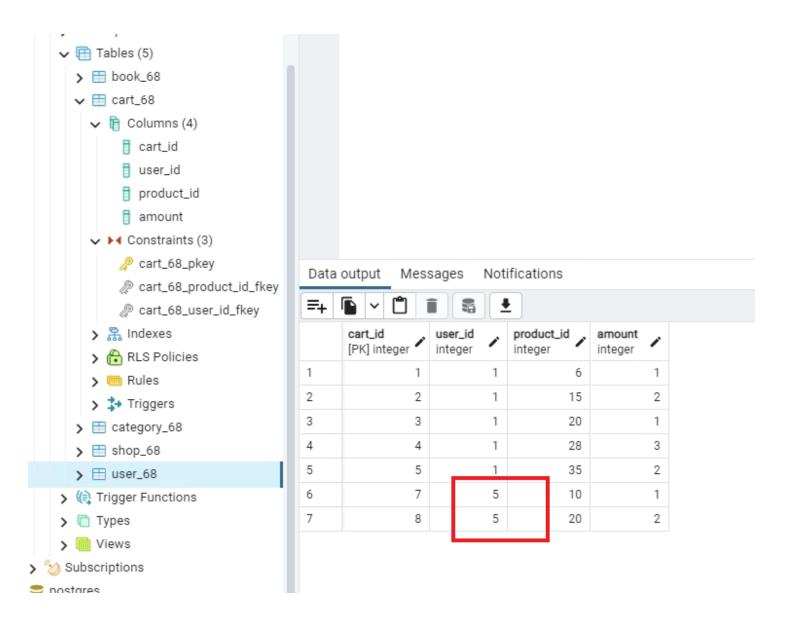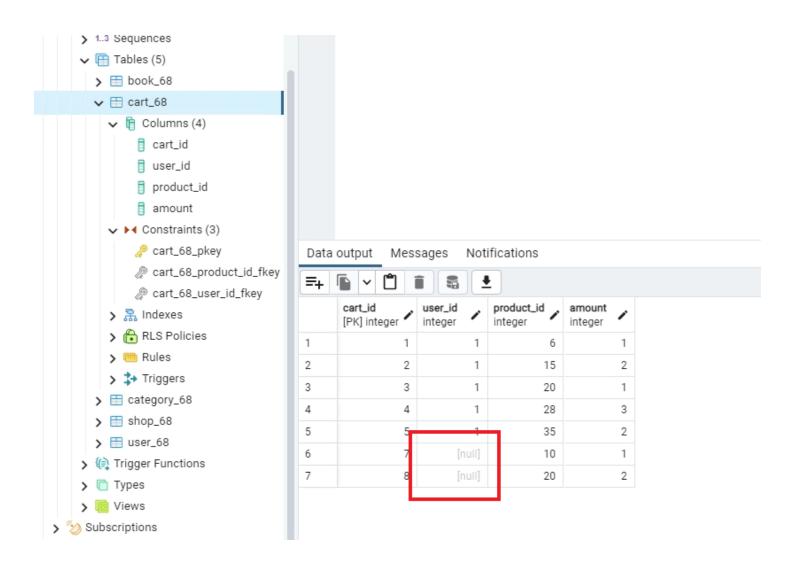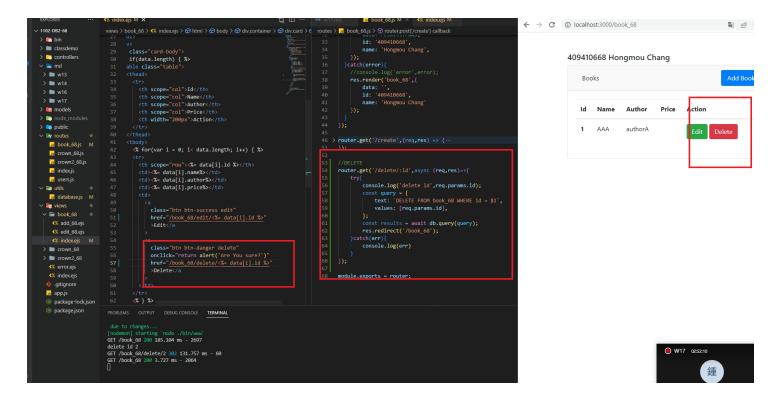## w17-p5.2: test ON DELETE SET NULL

(刪掉第二個 user 那步驟忘記截)

w17-p6: implement /book_xx/delete/:id

w17-p7: implement /book_xx/update

```
16      <div class="container" style="margin-top: 50px">
17          <div class="card">
18              <div class="card-header">Edit Book</div>
19              <div class="card-body">
20                  <form action="/book_68/update" method="post">
21                      <div class="form-group">
22                          <input
23                              type="hidden"
24                              class="form-control"
25                              name="id"
26                              value="<%= id %>"
27                              autocomplete="off"
28                          />
29                      </div>
30                      <div class="form-group">…
39                      </div>
40                      <div class="form-group">
41                          <label>Author:</label>
42                          <input
43                              type="text"
44                              class="form-control"
45                              name="author"
46                              value="<%= author %>"
47                              autocomplete="off"
48                          />
49                      </div>
50                      <div class="form-group">
51                          <label>Price:</label>
52                          <input
53                              type="text"
54                              class="form-control"
55                              name="price"
56                              value="<%= price %>"
57                              autocomplete="off"
58                          />
59                      </div>
```

```
61              };
62              const results = await db.query(query);
63              data = results.rows;
64              res.render('book_68/edit_68',{
65                  id: data[0].id,
66                  name: data[0].name,
67                  author: data[0].author,
68                  price: data[0].price,
69              });
70          }catch(err){
71              console.log(err);
72          }
73      });
74
75      //UPDATE
76      router.post('/update',async (req,res) => {
77          try{
78              const query = {
79                  text: `UPDATE book_68 SET name = $1, author = $2, price = $3 WHERE id = $4`,
80                  values: [req.body.name, req.body.author, req.body.price, req.body.id],
81              };
82              await db.query(query);
83              res.redirect('/book_68');
84          }catch(err){
85              console.log(err);
86          }
87      });
88
89      //DELETE
90      router.get('/delete/:id',async (req,res)=>{…
102     });
103
104     module.exports = router;
```

```
[nodemon] restarting due to changes...
[nodemon] starting `node ./bin/www`
GET /book_68 200 115.361 ms - 2067
id 1
GET /book_68/edit/1 200 8.741 ms - 1952
POST /book_68/update 302 24.542 ms - 60
GET /book_68 200 3.046 ms - 2067
id 1
GET /book_68/edit/1 200 115.949 ms - 1952
POST /book_68/update 302 5.223 ms - 60
GET /book_68 200 2.271 ms - 2068
```