University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 4

Date: April 5, 2023

Group Number: 94

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Hansen Dan	84620178	u0c4h	hdan2580@gmail.com
Celine Liu	20153755	t3z6w	zijingliu2021@outlook.com
Bhavye Thukral	80045370	t3l0m	bhavyeth@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Description

The objective of this project is to create an inventory tracking system that will enable a business to monitor the state of each package and the quantity of goods on hand. Users will be able to examine package contents, present location, and transit status through the company's internal distribution network using the system.

Data on goods can be managed and viewed with ease thanks to the system's user interface. With the help of the interface, users will be able to look for particular goods or packages, update information about each package, see how many of each item are currently on hand, and monitor the progress of packages as they travel through the supply chain.

Real-time updates on package and inventory progress will also be provided by the system. Users will thus have access to the most recent data regarding the location and status of packages, the number of items in stock, and other relevant information.

Overall, this inventory monitoring system will give a strong tool for managing inventory and following packages through the internal distribution network.

Schema Differences

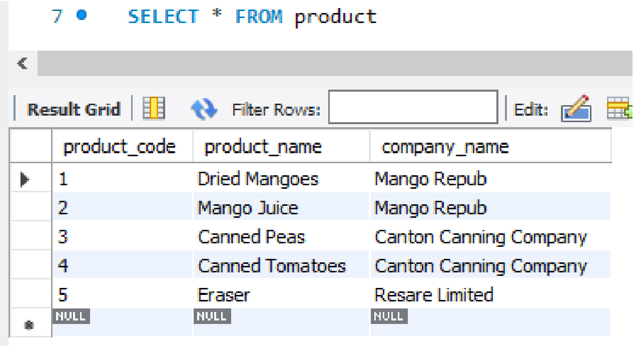
The following minor changes have been made to the schema from milestone 2:

- Renamed the table "Contains" as "Has" to avoid using the SQL reserved words.
- The data type for license_plate attribute was changed from CHAR[6] to CHAR[7] to accommodate the convenience of having a space in the middle, for example, “HIJ 120”. This change was applied to all associated tables (Transportation, Package, Internal_Fleet, External_Fleet, Travels_to).

Schema Screenshots

Schema:

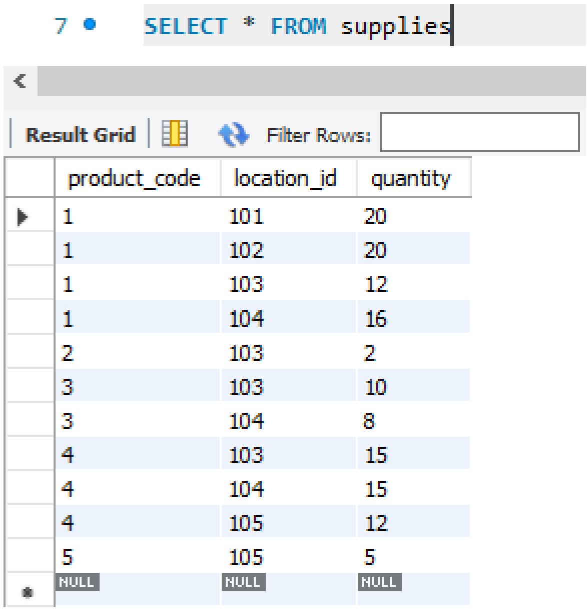
- Product(product_code: integer, product_name, company_name: char[30])



```
SELECT * FROM product
```

product_code	product_name	company_name
1	Dried Mangoes	Mango Repub
2	Mango Juice	Mango Repub
3	Canned Peas	Canton Canning Company
4	Canned Tomatoes	Canton Canning Company
5	Eraser	Resare Limited
NULL	NULL	NULL

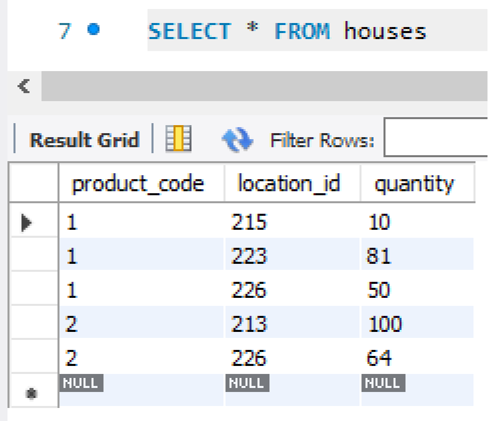
- Supplies(product_code: integer, location_id: integer, quantity: integer)



```
SELECT * FROM supplies
```

product_code	location_id	quantity
1	101	20
1	102	20
1	103	12
1	104	16
2	103	2
3	103	10
3	104	8
4	103	15
4	104	15
4	105	12
5	105	5
NULL	NULL	NULL

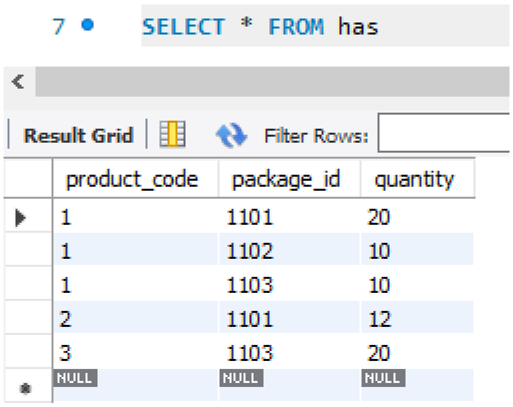
- Houses(product_code: integer, location_id: integer, quantity: integer)



```
SELECT * FROM houses
```

product_code	location_id	quantity
1	215	10
1	223	81
1	226	50
2	213	100
2	226	64
NULL	NULL	NULL

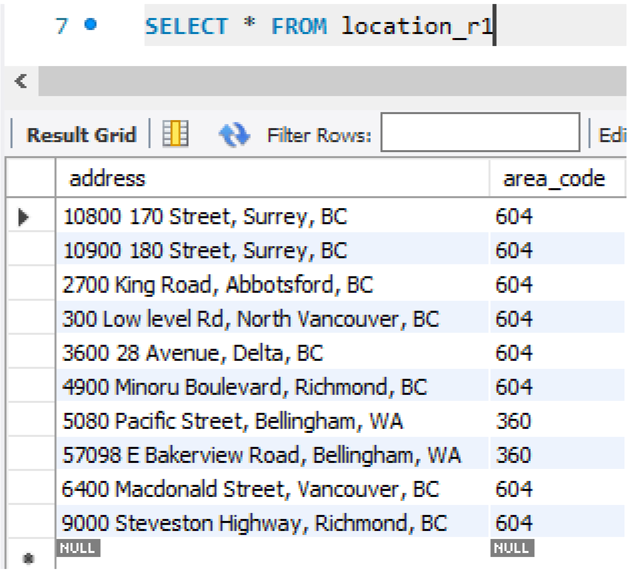
- Has(product_code: integer, package_id: integer, quantity: integer)



```
SELECT * FROM has
```

product_code	package_id	quantity
1	1101	20
1	1102	10
1	1103	10
2	1101	12
3	1103	20
NULL	NULL	NULL

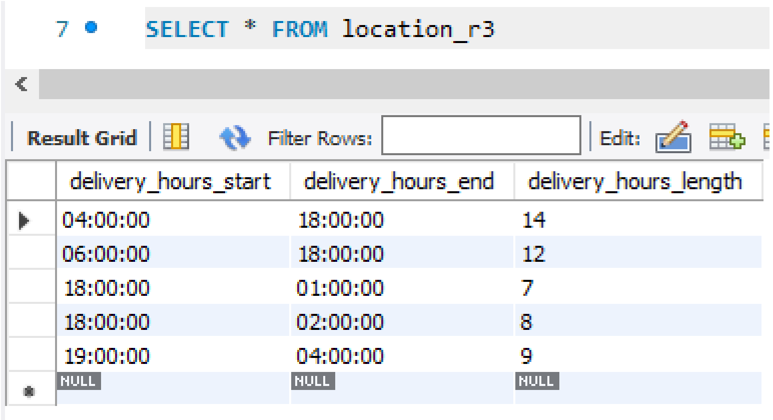
- Location_R1(address, area_code)



```
SELECT * FROM location_r1
```

address	area_code
10800 170 Street, Surrey, BC	604
10900 180 Street, Surrey, BC	604
2700 King Road, Abbotsford, BC	604
300 Low level Rd, North Vancouver, BC	604
3600 28 Avenue, Delta, BC	604
4900 Minoru Boulevard, Richmond, BC	604
5080 Pacific Street, Bellingham, WA	360
57098 E Bakerview Road, Bellingham, WA	360
6400 Macdonald Street, Vancouver, BC	604
9000 Steveston Highway, Richmond, BC	604
NULL	NULL

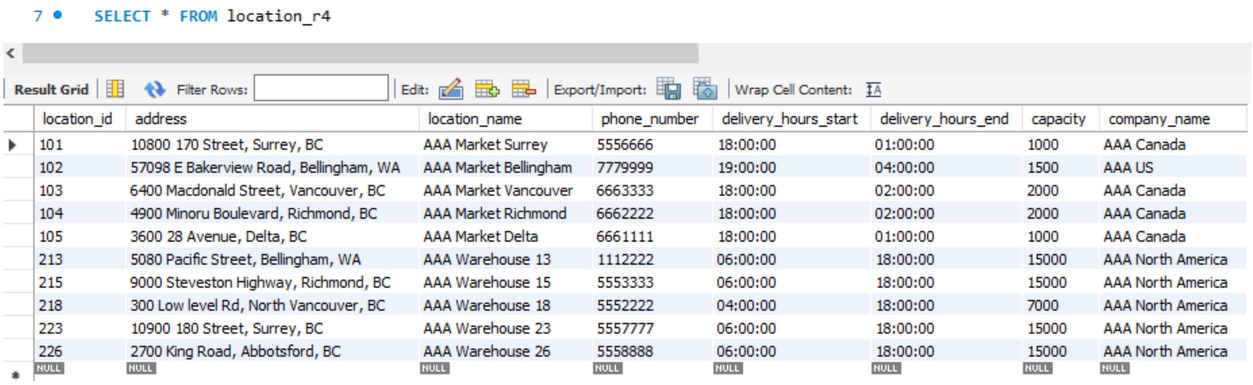
- Location_R3(delivery_hours_start, delivery_hours_end, delivery_hours_length)



```
SELECT * FROM location_r3
```

delivery_hours_start	delivery_hours_end	delivery_hours_length
04:00:00	18:00:00	14
06:00:00	18:00:00	12
18:00:00	01:00:00	7
18:00:00	02:00:00	8
19:00:00	04:00:00	9
NULL	NULL	NULL

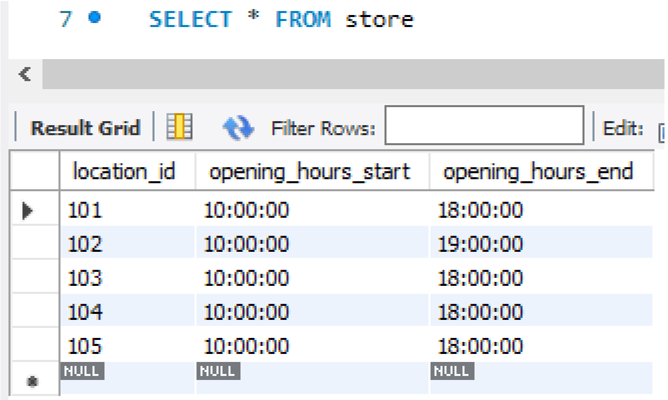
- Location_R4(**address**, location_id, location_name, phone_number, delivery_hours_start, delivery_hours_end, capacity, **company_name**)



```
SELECT * FROM location_r4
```

location_id	address	location_name	phone_number	delivery_hours_start	delivery_hours_end	capacity	company_name
101	10800 170 Street, Surrey, BC	AAA Market Surrey	5556666	18:00:00	01:00:00	1000	AAA Canada
102	57098 E Bakerview Road, Bellingham, WA	AAA Market Bellingham	7779999	19:00:00	04:00:00	1500	AAA US
103	6400 Macdonald Street, Vancouver, BC	AAA Market Vancouver	6663333	18:00:00	02:00:00	2000	AAA Canada
104	4900 Minoru Boulevard, Richmond, BC	AAA Market Richmond	6662222	18:00:00	02:00:00	2000	AAA Canada
105	3600 28 Avenue, Delta, BC	AAA Market Delta	6661111	18:00:00	01:00:00	1000	AAA Canada
213	5080 Pacific Street, Bellingham, WA	AAA Warehouse 13	1112222	06:00:00	18:00:00	15000	AAA North America
215	9000 Steveston Highway, Richmond, BC	AAA Warehouse 15	5553333	06:00:00	18:00:00	15000	AAA North America
218	300 Low level Rd, North Vancouver, BC	AAA Warehouse 18	5552222	04:00:00	18:00:00	7000	AAA North America
223	10900 180 Street, Surrey, BC	AAA Warehouse 23	5557777	06:00:00	18:00:00	15000	AAA North America
226	2700 King Road, Abbotsford, BC	AAA Warehouse 26	5558888	06:00:00	18:00:00	15000	AAA North America
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

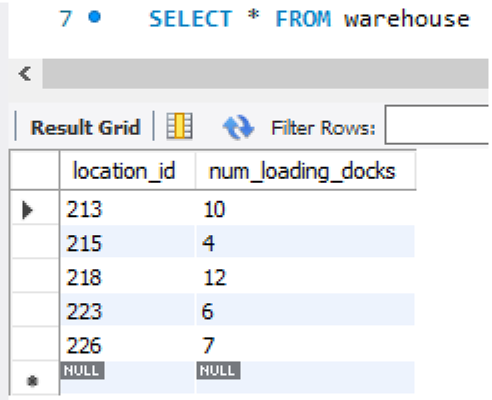
- Store(**location_id**: integer, opening_hours_start: time, opening_hours_end: time)



```
SELECT * FROM store
```

location_id	opening_hours_start	opening_hours_end
101	10:00:00	18:00:00
102	10:00:00	19:00:00
103	10:00:00	18:00:00
104	10:00:00	18:00:00
105	10:00:00	18:00:00
NULL	NULL	NULL

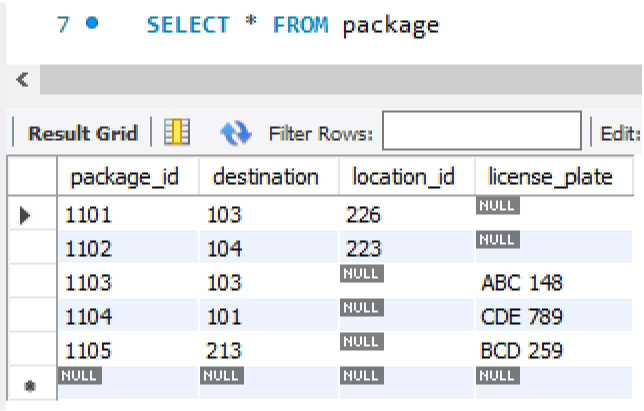
- Warehouse(**location_id**: integer, num_loading_docks: integer)



```
SELECT * FROM warehouse
```

location_id	num_loading_docks
213	10
215	4
218	12
223	6
226	7
NULL	NULL

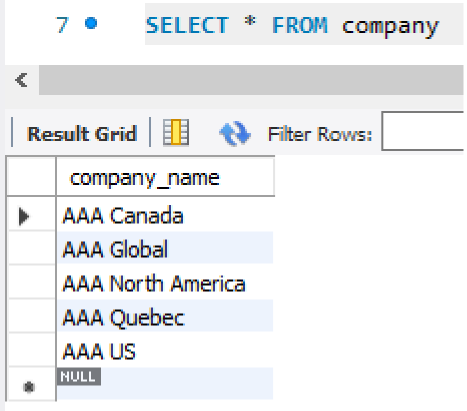
- Package(package_id, destination: integer, **location_id**: integer, **license_plate:** char[7])



```
SELECT * FROM package
```

package_id	destination	location_id	license_plate
1101	103	226	NULL
1102	104	223	NULL
1103	103	NULL	ABC 148
1104	101	NULL	CDE 789
1105	213	NULL	BCD 259
NULL	NULL	NULL	NULL

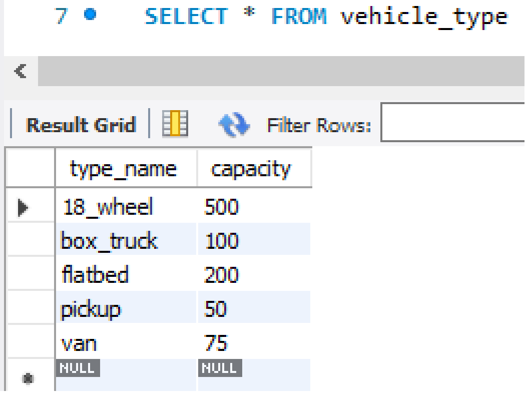
- Company(company_name: char[30])



```
SELECT * FROM company
```

company_name
AAA Canada
AAA Global
AAA North America
AAA Quebec
AAA US
NULL

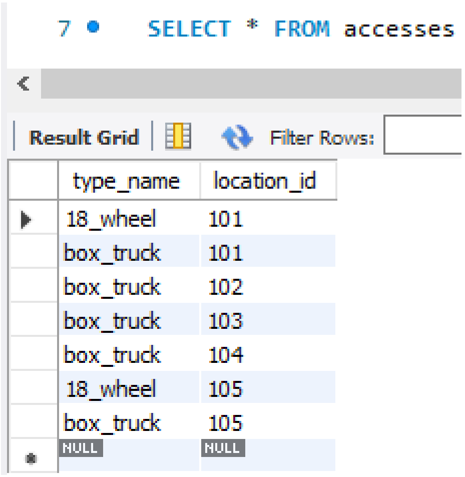
- Vehicle_type(type_name: char[30], capacity: integer)



```
SELECT * FROM vehicle_type
```

type_name	capacity
18_wheel	500
box_truck	100
flatbed	200
pickup	50
van	75
NULL	NULL

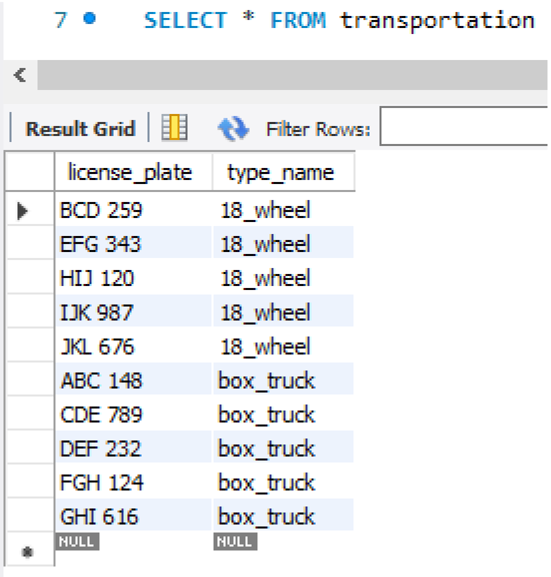
- Accesses(type_name: char[30], **location_id**: integer)



```
SELECT * FROM accesses
```

type_name	location_id
18_wheel	101
box_truck	101
box_truck	102
box_truck	103
box_truck	104
18_wheel	105
box_truck	105
NULL	NULL

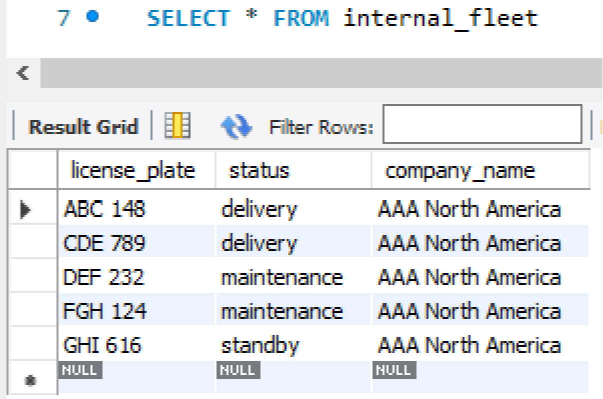
- Transportation(license_plate: char[7], **type_name**: char[30])



```
SELECT * FROM transportation
```

license_plate	type_name
BCD 259	18_wheel
EFG 343	18_wheel
HIJ 120	18_wheel
IJK 987	18_wheel
JKL 676	18_wheel
ABC 148	box_truck
CDE 789	box_truck
DEF 232	box_truck
FGH 124	box_truck
GHI 616	box_truck
NULL	NULL

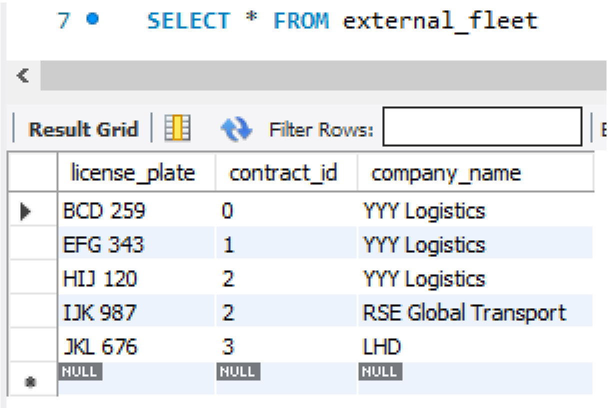
- Internal_Fleet(**license_plate**: char[7], status: char[10], **company_name:** char[30])



```
SELECT * FROM internal_fleet
```

license_plate	status	company_name
ABC 148	delivery	AAA North America
CDE 789	delivery	AAA North America
DEF 232	maintenance	AAA North America
FGH 124	maintenance	AAA North America
GHI 616	standby	AAA North America
NULL	NULL	NULL

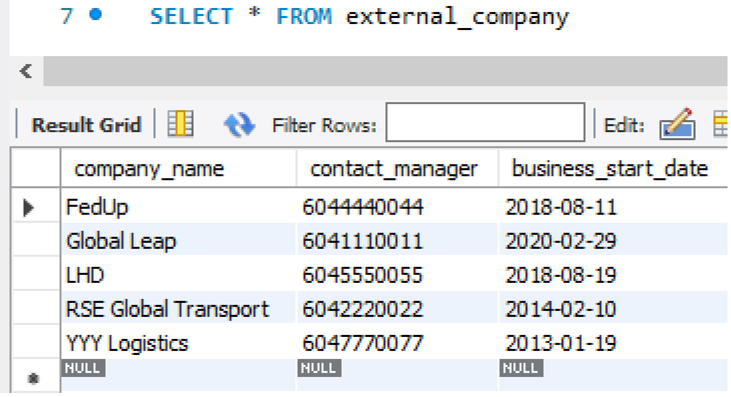
- External_Fleet(**license_plate**: char[7], contract_id: integer, **company_name**: char[30])



```
SELECT * FROM external_fleet
```

license_plate	contract_id	company_name
BCD 259	0	YYY Logistics
EFG 343	1	YYY Logistics
HIJ 120	2	YYY Logistics
IJK 987	2	RSE Global Transport
JKL 676	3	LHD
NULL	NULL	NULL

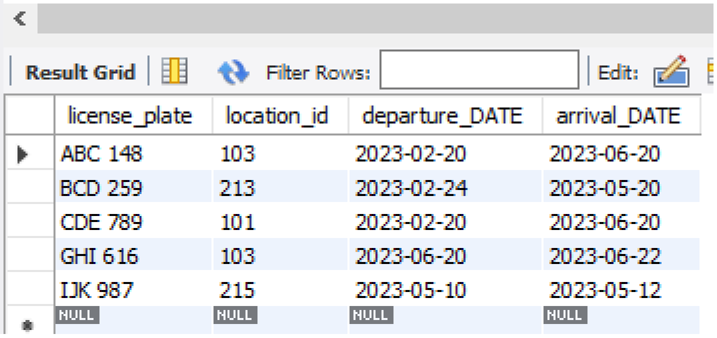
- External_Company(company_name: char[30], contact_manager: char[20], business_start_date: date, **license_plate:** char[7])



```
SELECT * FROM external_company
```

company_name	contact_manager	business_start_date
FedUp	6044440044	2018-08-11
Global Leap	6041110011	2020-02-29
LHD	6045550055	2018-08-19
RSE Global Transport	6042220022	2014-02-10
YYY Logistics	6047770077	2013-01-19
NULL	NULL	NULL

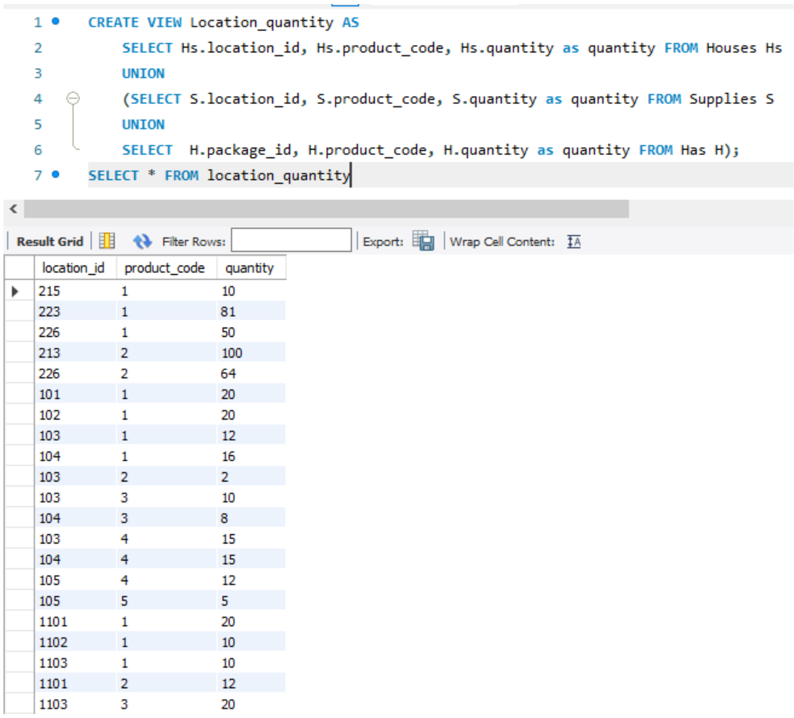
- Travels_to(license_plate: char[7], location_id: integer, departure_date: datetime, arrival_date: datetime)



license_plate	location_id	departure_DATE	arrival_DATE
ABC 148	103	2023-02-20	2023-06-20
BCD 259	213	2023-02-24	2023-05-20
CDE 789	101	2023-02-20	2023-06-20
GHI 616	103	2023-06-20	2023-06-22
IJK 987	215	2023-05-10	2023-05-12
NULL	NULL	NULL	NULL

Views:

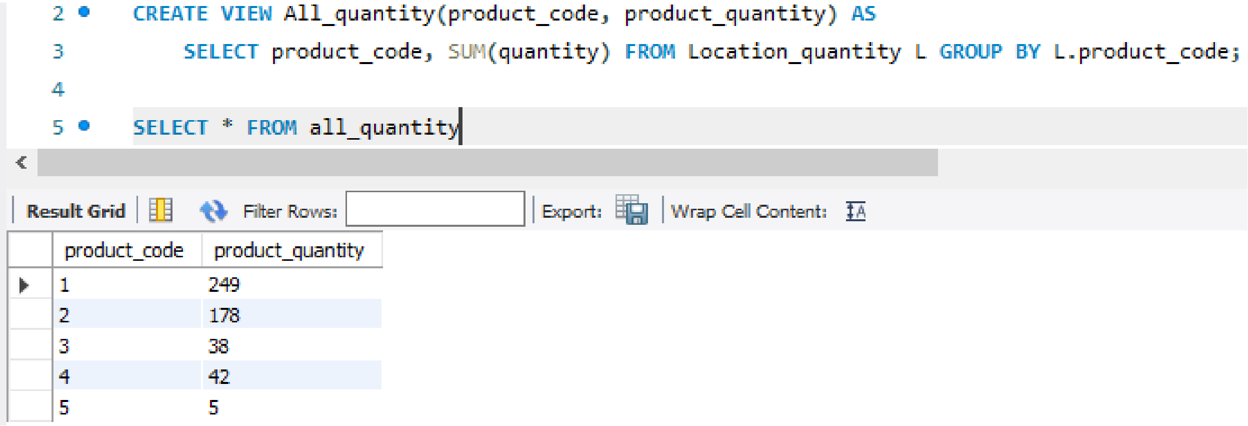
- Location_quantity(location_id: int, product_code: int, quantity: int)



```
CREATE VIEW Location_quantity AS
    SELECT Hs.location_id, Hs.product_code, Hs.quantity as quantity FROM Houses Hs
    UNION
    (SELECT S.location_id, S.product_code, S.quantity as quantity FROM Supplies S
    UNION
    SELECT  H.package_id, H.product_code, H.quantity as quantity FROM Has H);
SELECT * FROM location_quantity
```

location_id	product_code	quantity
215	1	10
223	1	81
226	1	50
213	2	100
226	2	64
101	1	20
102	1	20
103	1	12
104	1	16
103	2	2
103	3	10
104	3	8
103	4	15
104	4	15
105	4	12
105	5	5
1101	1	20
1102	1	10
1103	1	10
1101	2	12
1103	3	20

- All_quantity (product_code: int, product_quantity: int)



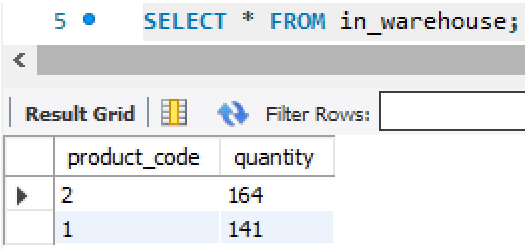
```
CREATE VIEW All_quantity(product_code, product_quantity) AS
    SELECT product_code, SUM(quantity) FROM Location_quantity L GROUP BY L.product_code;

SELECT * FROM all_quantity
```

product_code	product_quantity
1	249
2	178
3	38
4	42
5	5

- in_warehouse(product_code: int, product_quantity: int)

```
CREATE VIEW in_warehouse AS
SELECT product_code, IFNULL(SUM(quantity), 0) AS quantity
FROM Location_quantity LQ
WHERE EXISTS (SELECT * FROM Houses Hs WHERE Hs.location_id =
LQ.location_id) GROUP BY product_code;
```

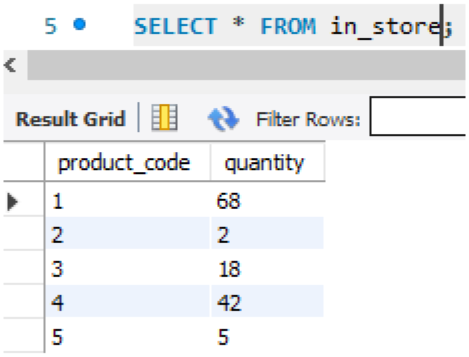


```
SELECT * FROM in_warehouse;
```

product_code	quantity
2	164
1	141

- in_store(product_code: int, product_quantity: int)

```
CREATE VIEW in_store AS
SELECT product_code, IFNULL(SUM(quantity), 0) AS quantity
FROM Location_quantity LQ
WHERE EXISTS (SELECT * FROM Supplies S WHERE S.location_id =
LQ.location_id)
GROUP BY product_code;
```

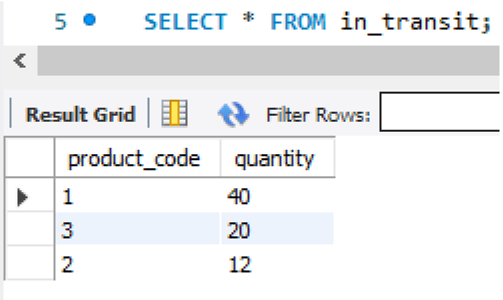


```
SELECT * FROM in_store;
```

product_code	quantity
1	68
2	2
3	18
4	42
5	5

- in_transit(product_code: int, product_quantity: int)

```
CREATE VIEW in_transit AS
SELECT product_code, IFNULL(SUM(quantity), 0) AS quantity
FROM Location_quantity LQ
WHERE EXISTS (SELECT * FROM Package P WHERE P.package_id = LQ.location_id)
GROUP BY product_code;
```



```
SELECT * FROM in_transit;
```

product_code	quantity
1	40
3	20
2	12

SQL Query List

Get a tuple for a specified product (check for existence) (site/add_new.php:101)

```
SELECT * FROM Product P WHERE P.product_code = :pid
```

Add a new product (site/add_new.php:115)

```
INSERT INTO Product VALUES ( :p_code , :p_name , :p_manu );
```

Delete a specified product (site/add_new.php:179)

```
DELETE FROM Product P WHERE P.product_code = :pid
```

Update product Information (site/update_product.php:91)

```
UPDATE product SET product_name = :new_pname , company_name = :new_pmanu
WHERE product_code = :pid
```

Insert new Inventory information for Stores/Warehouses (site/update.php:136/145)

```
INSERT INTO Supplies VALUES ( :pid , :lid , :quantity );
INSERT INTO Houses VALUES ( :pid , :lid , :quantity );
```

Update existing Inventory information Stores/Warehouses (site/update.php:139/148)

```
UPDATE Supplies SET quantity = :quantity WHERE product_code = :pid AND
location_id = :lid
UPDATE Houses SET quantity = :quantity WHERE product_code = :pid AND
location_id = :lid
```

View Store Information (site/projection.php:149)

```
SELECT $filters
                FROM store S, location_r1 LR1, location_r3 LR3, location_r4
LR4
                WHERE LR4.location_id = S.location_id
                AND LR4.address = LR1.address
                AND LR4.delivery_hours_start = LR3.delivery_hours_start
                AND LR4.delivery_hours_end = LR3.delivery_hours_end
```

Selection-Stock information(site/stock.php:62)

```
SELECT S.location_id, S.product_code, S.quantity, Pd.product_name
FROM supplies S, product Pd, location_r4 LR4
DELETE FROM Product P WHERE P.product_code = :pid
```

Selection-Product Search(site/product_search.php:55)

```
SELECT * FROM product P WHERE P.product_name LIKE '%$search_term%' LIMIT
$limit
```

Selection-Company Search(site/company_search.php:55)

```
SELECT * FROM product P WHERE P.company_name LIKE '%$search_term%' LIMIT
$limit
```

Get the the location a package is delivered to

```
SELECT P.package_id, LR4.location_id, LR4.address
FROM location_R4 LR4, Package P
WHERE LR4. location_id = P.location_id
```

Give a package_id, if the package has arrived in a location, return the location and contents of the package.

```
SELECT Pk.package_id, Pd.product_name, H.quantity, Pk.destination,
LR4.address, concat(LR1.area_code, LR4.phone_number) as phone
FROM Package Pk, Location_R4 LR4, Location_R1 LR1, Has H, Product Pd
WHERE Pk.location_id = LR4.location_id and LR4.address = LR1.address
      and Pk.package_id = $package_id and H.package_id = $package_id and
H.product_code = Pd.product_code
```

Give a package_id, if the package is in transit, return current trip details(site/package_track.php:60)

```
SELECT Pk.package_id, Pd.product_name, H.quantity, Pk.destination,
Tr.license_plate, Tr.departure_date, Tr.arrival_DATE
FROM Package Pk, Has H, Product Pd, Transportation T, Travels_to Tr
WHERE Pk.license_plate = T.license_plate
      and T.license_plate = Tr.license_plate
      and Pk.package_id = $package_id
      and $package_id = H.package_id
      and H.product_code = Pd.product_code
```

Aggregation: Nested: Find the store that has the most of the item that is in the shortest supply (site/short_supply.php:17)

```
-- get all each location's product quantity
CREATE VIEW Location_quantity AS
   SELECT Hs.location_id, Hs.product_code, Hs.quantity as quantity FROM Houses Hs
   UNION
   (SELECT S.location_id, S.product_code, S.quantity as quantity FROM Supplies S
   UNION
   SELECT  H.package_id, H.product_code, H.quantity as quantity FROM Has H);

-- get each product quantity in the system
CREATE VIEW All_quantity(product_code, product_quantity) AS
SELECT product_code, SUM(quantity) FROM Location_quantity L GROUP BY
L.product_code;

SELECT S.location_id, S.product_code, S.quantity
FROM Supplies S
WHERE S.product_code IN
(SELECT AQ.product_code FROM All_quantity as AQ
 WHERE AQ.product_quantity <= all (SELECT AQ1.product_quantity
                                   FROM All_quantity as AQ1))
       AND S.quantity <= all (SELECT S2.quantity FROM Supplies S2
                             WHERE S.product_code = S2.product_code)
```

For each product, get the quantities throughout the distribution network (at each store, warehouse, in transit, in total)(site/product_quantity.php:49)

```
CREATE VIEW in_warehouse AS
SELECT product_code, IFNULL(SUM(quantity), 0) AS quantity FROM Location_quantity LQ
WHERE EXISTS (SELECT * FROM Houses Hs WHERE Hs.location_id = LQ.location_id)
GROUP BY product_code;

CREATE VIEW in_store AS
SELECT product_code, IFNULL(SUM(quantity), 0) AS quantity FROM Location_quantity LQ
WHERE EXISTS (SELECT * FROM Supplies S WHERE S.location_id = LQ.location_id)
GROUP BY product_code;

CREATE VIEW in_transit AS
SELECT product_code, IFNULL(SUM(quantity), 0) AS quantity FROM Location_quantity LQ
WHERE EXISTS (SELECT * FROM Package P WHERE P.package_id = LQ.location_id)
GROUP BY product_code;

SELECT al.product_code, IFNULL(w.quantity, 0) as w_quantity, IFNULL(s.quantity, 0)
as s_quantity, IFNULL(t.quantity, 0) as t_quantity, IFNULL(al.product_quantity, 0)
as total_quantity
FROM (all_quantity al
LEFT JOIN in_warehouse w
on al.product_code = w.product_code
LEFT JOIN in_store s
ON al.product_code = s.product_code
LEFT JOIN in_transit t
ON al.product_code = t.product_code)
WHERE al.product_code = $product_code
```

For a product, find the the quantities of it in each location (site/add_new.php:165)

```
SELECT * From location_quantity LQ WHERE LQ.product_code = :pid
```

For a given vehicle and store, see if the vehicle can access that store(site/store_access.php:54)

```
SELECT a.type_name,l.location_name,l.location_id
FROM Accesses a, Location_R4 l
WHERE  a.location_id=l.location_id AND a.type_name = \"$vehicle_type\" AND
a.location_id =$location_id
```

Aggregation:Nested- Product Quantity (site/product_quantity.php:

```
SELECT al.product_code, IFNULL(w.quantity, 0) as w_quantity, IFNULL(s.quantity, 0) as
s_quantity, IFNULL(t.quantity, 0) as t_quantity, IFNULL(al.product_quantity, 0) as
total_quantity
               FROM (all_quantity al
               LEFT JOIN in_warehouse w
               on al.product_code = w.product_code
               LEFT JOIN in_store s
               ON al.product_code = s.product_code
               LEFT JOIN in_transit t
               ON al.product_code = t.product_code)
               WHERE al.product_code = $product_code
```

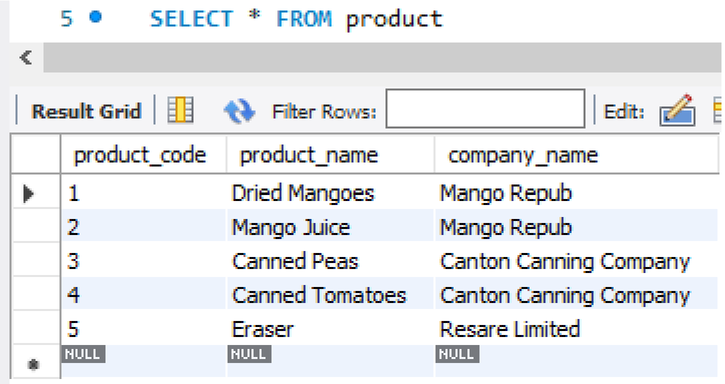
Division-Check Stores Supplying a Set of Products (site/Supplies_count.php:66)

```
SELECT s.location_id
               FROM Supplies s
               WHERE s.product_code IN ( $product_code )
               GROUP BY s.location_id
               HAVING COUNT(DISTINCT s.product_code) = $product_code_count;
```

Query Functionality Screenshots

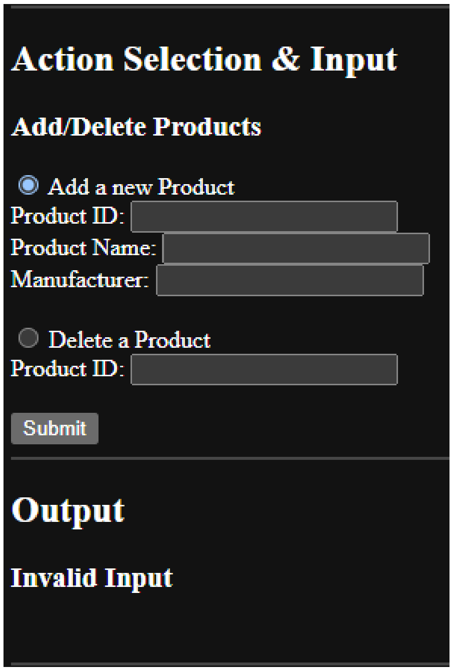
INSERT: Add Products

- Before



```
SELECT * FROM product
```

product_code	product_name	company_name
1	Dried Mangoes	Mango Repub
2	Mango Juice	Mango Repub
3	Canned Peas	Canton Canning Company
4	Canned Tomatoes	Canton Canning Company
5	Eraser	Resare Limited
NULL	NULL	NULL



Action Selection & Input

Add/Delete Products

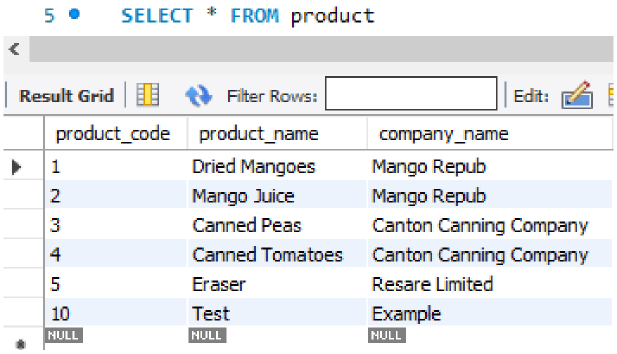
● Add a new Product
Product ID:
Product Name:
Manufacturer:

○ Delete a Product
Product ID:

Output

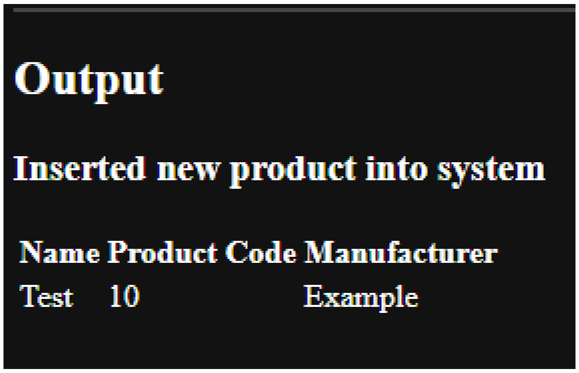
Invalid Input

- After



```
SELECT * FROM product
```

product_code	product_name	company_name
1	Dried Mangoes	Mango Repub
2	Mango Juice	Mango Repub
3	Canned Peas	Canton Canning Company
4	Canned Tomatoes	Canton Canning Company
5	Eraser	Resare Limited
10	Test	Example
NULL	NULL	NULL



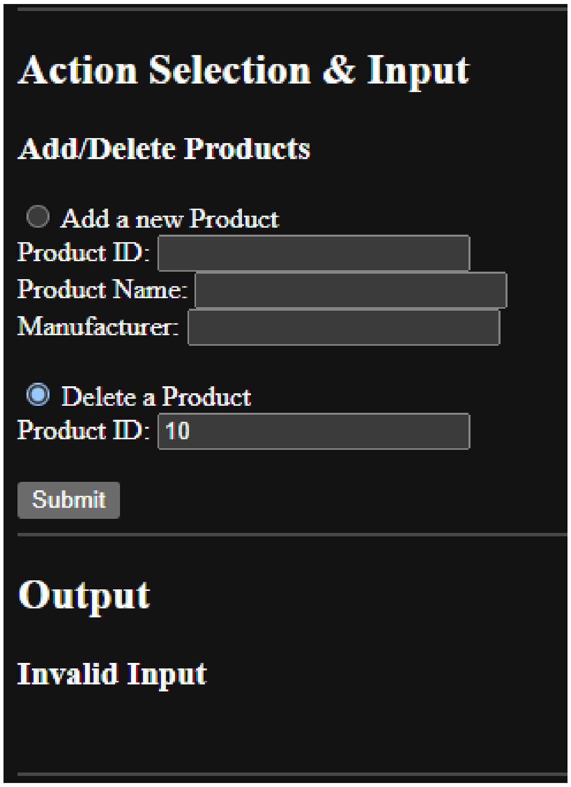
Output

Inserted new product into system

Name	Product Code	Manufacturer
Test	10	Example

DELETE: Remove Products

- Before



Action Selection & Input

Add/Delete Products

○ Add a new Product

Product ID:

Product Name:

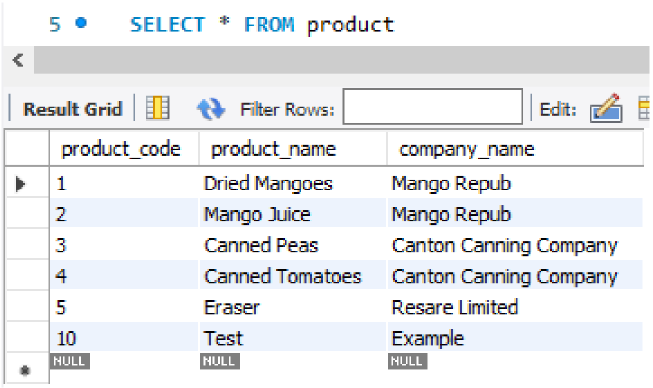
Manufacturer:

● Delete a Product

Product ID: 10

Output

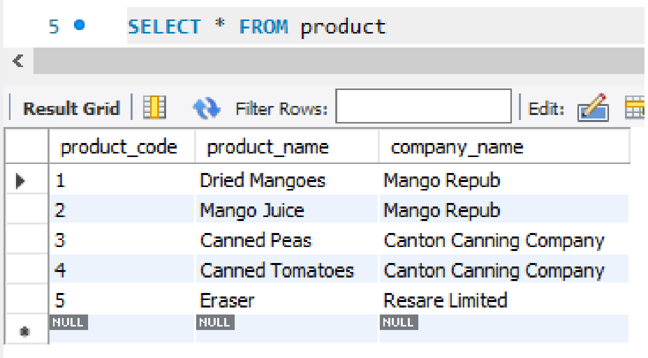
Invalid Input



```
SELECT * FROM product
```

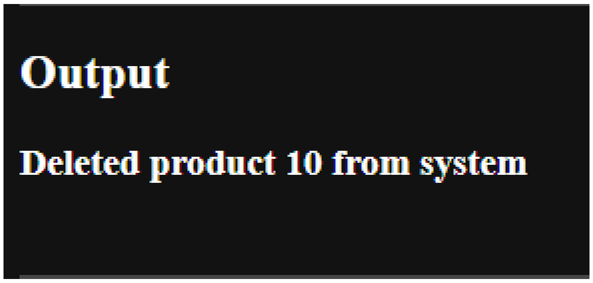
product_code	product_name	company_name
1	Dried Mangoes	Mango Repub
2	Mango Juice	Mango Repub
3	Canned Peas	Canton Canning Company
4	Canned Tomatoes	Canton Canning Company
5	Eraser	Resare Limited
10	Test	Example
NULL	NULL	NULL

- After



```
SELECT * FROM product
```

product_code	product_name	company_name
1	Dried Mangoes	Mango Repub
2	Mango Juice	Mango Repub
3	Canned Peas	Canton Canning Company
4	Canned Tomatoes	Canton Canning Company
5	Eraser	Resare Limited
NULL	NULL	NULL

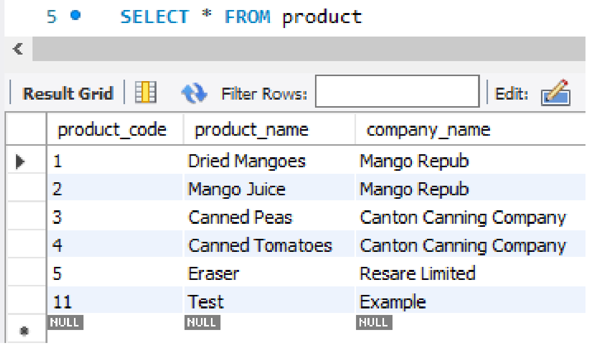


Output

Deleted product 10 from system

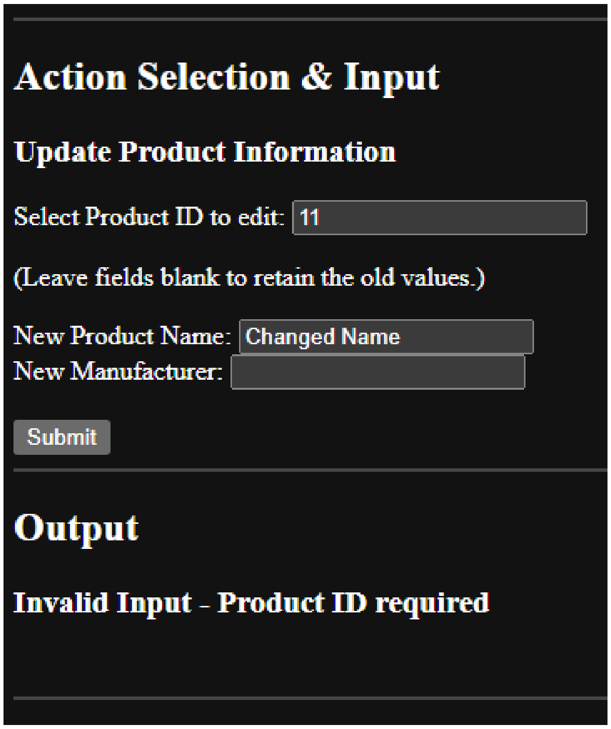
UPDATE: Update Product Information

- Before



```
SELECT * FROM product
```

product_code	product_name	company_name
1	Dried Mangoes	Mango Repub
2	Mango Juice	Mango Repub
3	Canned Peas	Canton Canning Company
4	Canned Tomatoes	Canton Canning Company
5	Eraser	Resare Limited
11	Test	Example
NULL	NULL	NULL



Action Selection & Input

Update Product Information

Select Product ID to edit: 11

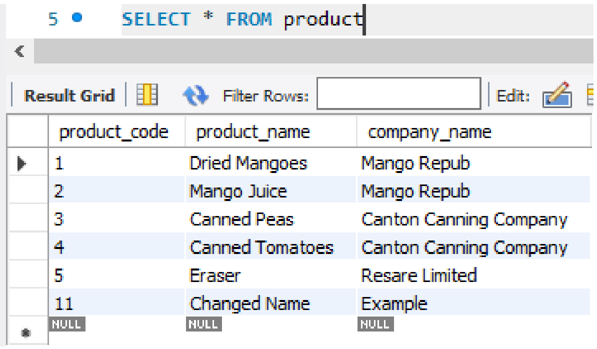
(Leave fields blank to retain the old values.)

New Product Name: Changed Name
New Manufacturer:

Output

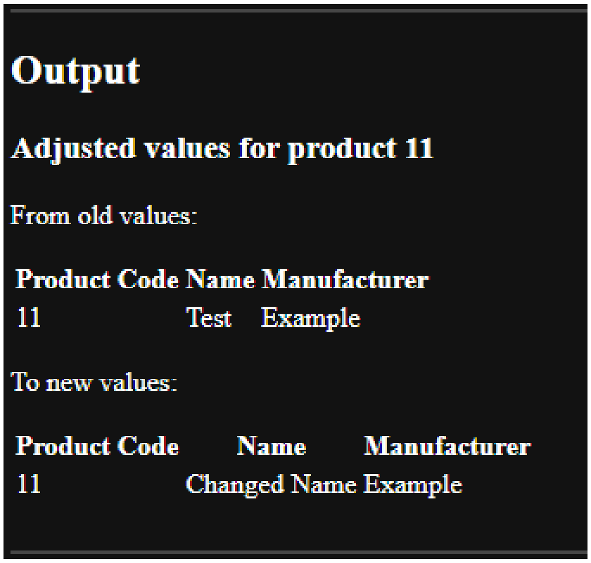
Invalid Input - Product ID required

- After



```
SELECT * FROM product
```

product_code	product_name	company_name
1	Dried Mangoes	Mango Repub
2	Mango Juice	Mango Repub
3	Canned Peas	Canton Canning Company
4	Canned Tomatoes	Canton Canning Company
5	Eraser	Resare Limited
11	Changed Name	Example
NULL	NULL	NULL



Output

Adjusted values for product 11

From old values:

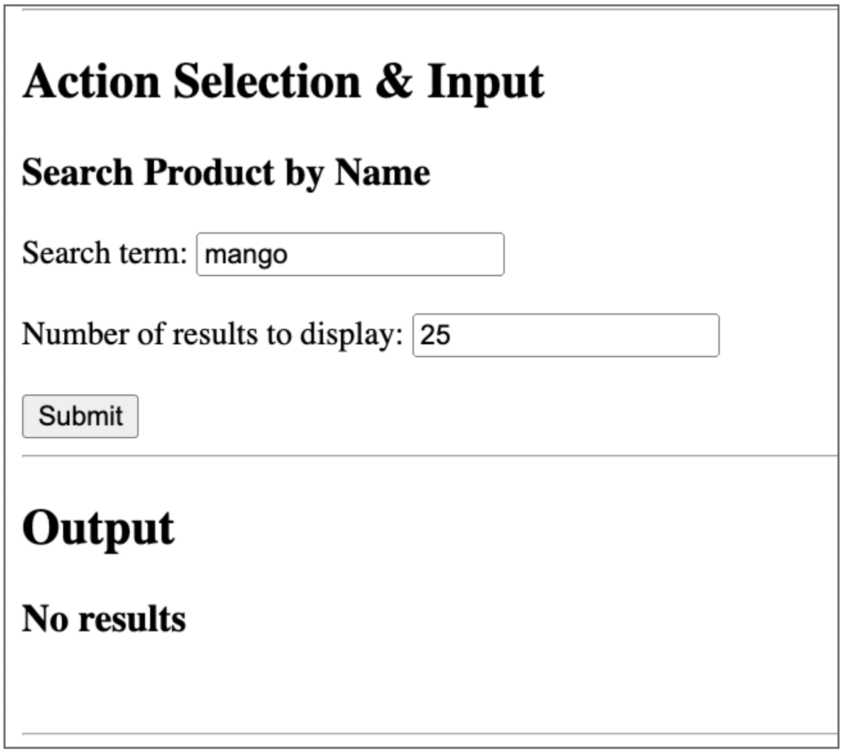
Product Code	Name	Manufacturer
11	Test	Example

To new values:

Product Code	Name	Manufacturer
11	Changed Name	Example

Selection: Product Search, etc.

- Before



Action Selection & Input

Search Product by Name

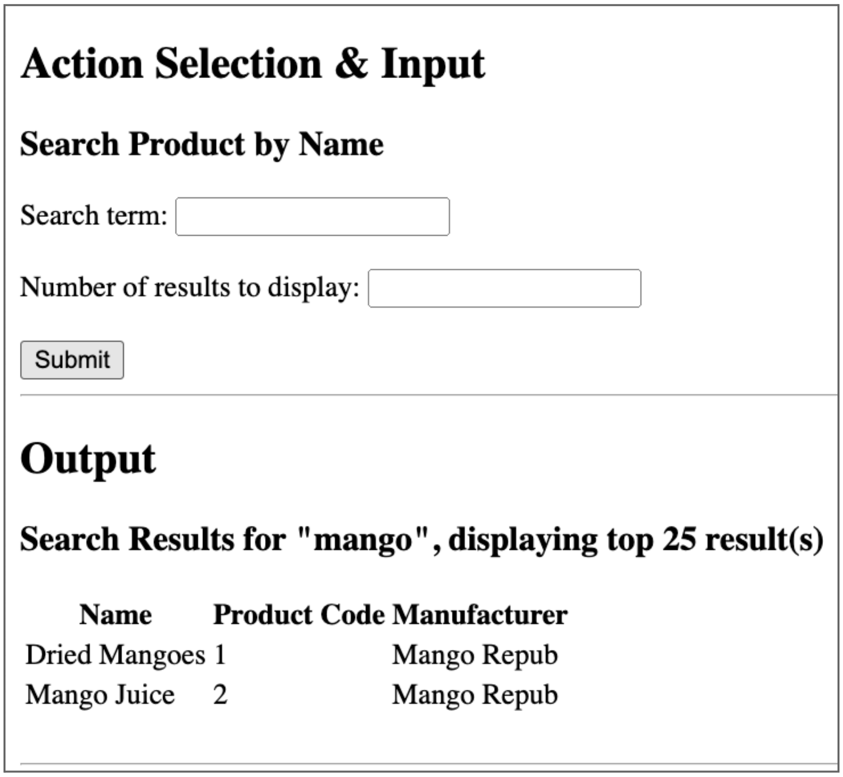
Search term: mango

Number of results to display: 25

Output

No results

- After



Action Selection & Input

Search Product by Name

Search term:

Number of results to display:

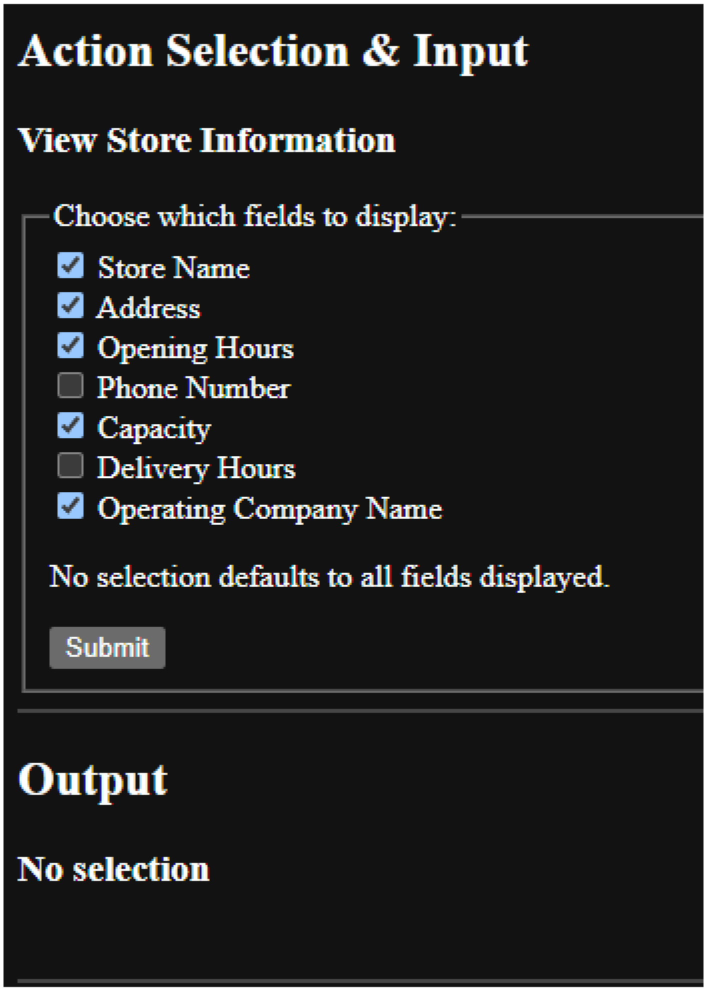
Output

Search Results for "mango", displaying top 25 result(s)

Name	Product Code	Manufacturer
Dried Mangoes	1	Mango Repub
Mango Juice	2	Mango Repub

Projection: View Store Information

- Before



Action Selection & Input

View Store Information

Choose which fields to display:

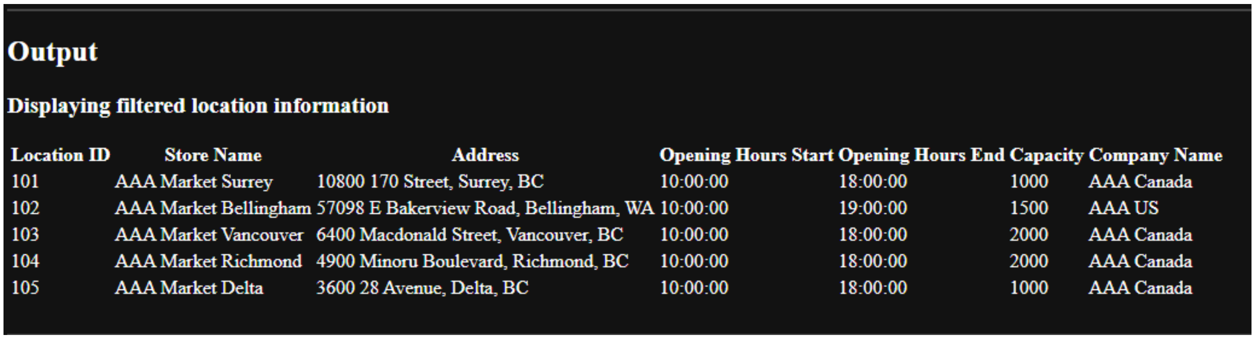
☑ Store Name
☑ Address
☑ Opening Hours
□ Phone Number
☑ Capacity
□ Delivery Hours
☑ Operating Company Name

No selection defaults to all fields displayed.

Output

No selection

- After



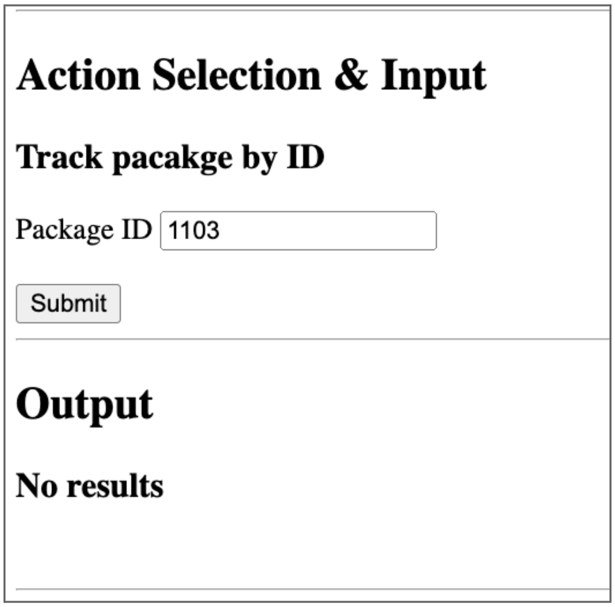
Output

Displaying filtered location information

Location ID	Store Name	Address	Opening Hours Start	Opening Hours End	Capacity	Company Name
101	AAA Market Surrey	10800 170 Street, Surrey, BC	10:00:00	18:00:00	1000	AAA Canada
102	AAA Market Bellingham	57098 E Bakerview Road, Bellingham, WA	10:00:00	19:00:00	1500	AAA US
103	AAA Market Vancouver	6400 Macdonald Street, Vancouver, BC	10:00:00	18:00:00	2000	AAA Canada
104	AAA Market Richmond	4900 Minoru Boulevard, Richmond, BC	10:00:00	18:00:00	2000	AAA Canada
105	AAA Market Delta	3600 28 Avenue, Delta, BC	10:00:00	18:00:00	1000	AAA Canada

Join: Package Track

- Before



Action Selection & Input

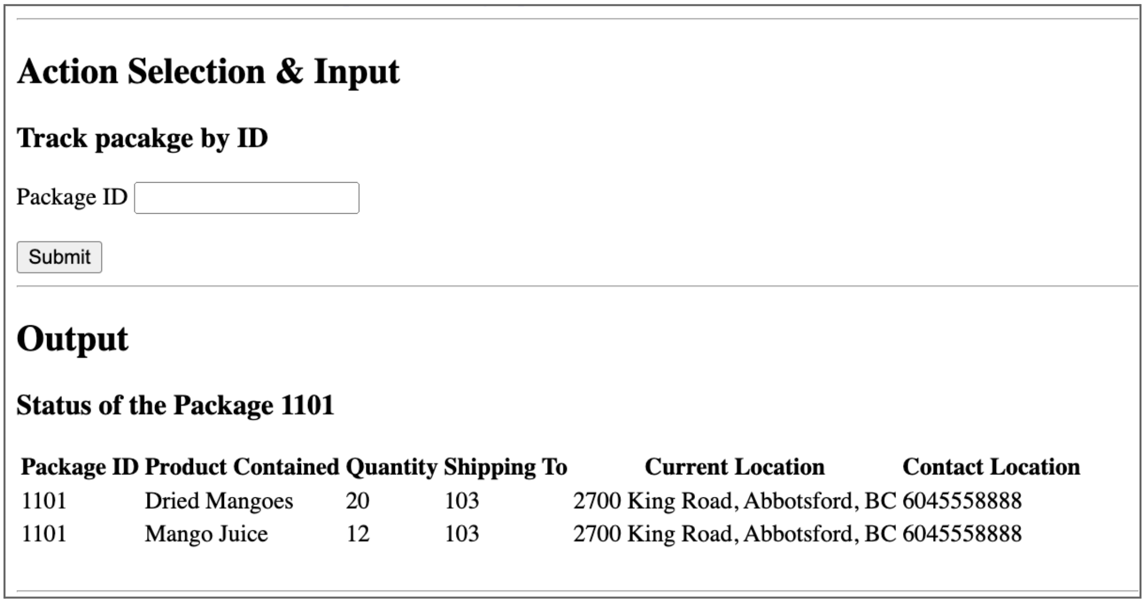
Track pacakge by ID

Package ID 1103

Output

No results

- After



Action Selection & Input

Track pacakge by ID

Package ID

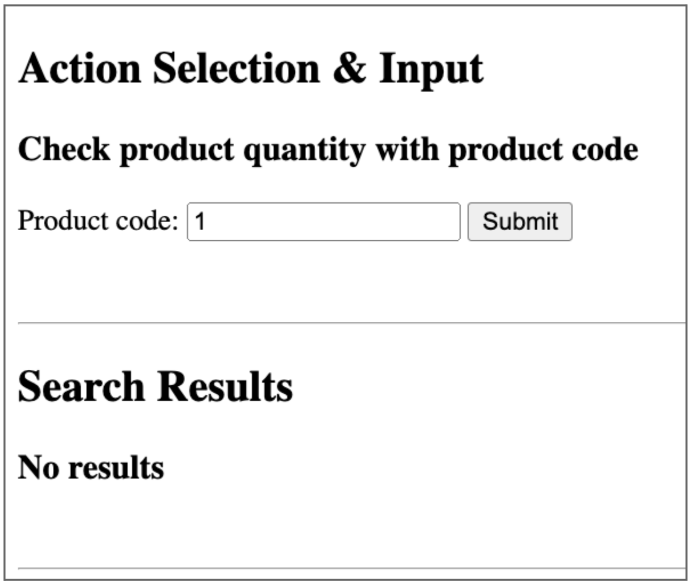
Output

Status of the Package 1101

Package ID	Product Contained	Quantity	Shipping To	Current Location	Contact Location
1101	Dried Mangoes	20	103	2700 King Road, Abbotsford, BC	6045558888
1101	Mango Juice	12	103	2700 King Road, Abbotsford, BC	6045558888

Aggregation (GROUP BY) : Product Quantity

- Before



Action Selection & Input

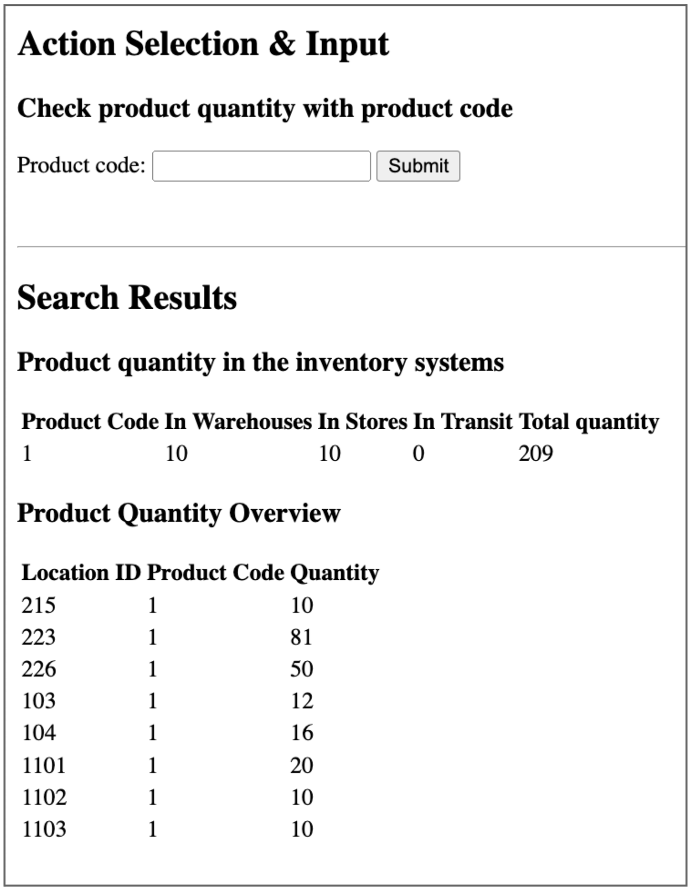
Check product quantity with product code

Product code: 1

Search Results

No results

- After



Action Selection & Input

Check product quantity with product code

Product code:

Search Results

Product quantity in the inventory systems

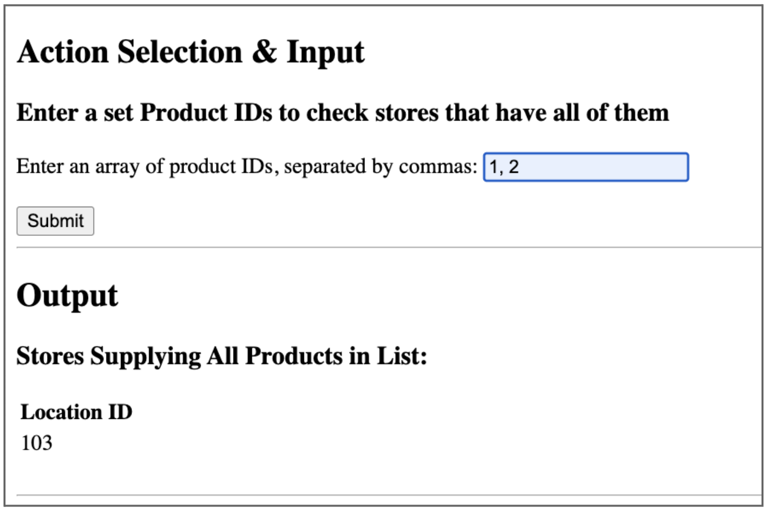
Product Code	In Warehouses	In Stores	In Transit	Total quantity
1	10	10	0	209

Product Quantity Overview

Location ID	Product Code	Quantity
215	1	10
223	1	81
226	1	50
103	1	12
104	1	16
1101	1	20
1102	1	10
1103	1	10

Aggregation (HAVING), Division: Check Stores Supplying a Set of Products

- Before



Action Selection & Input

Enter a set Product IDs to check stores that have all of them

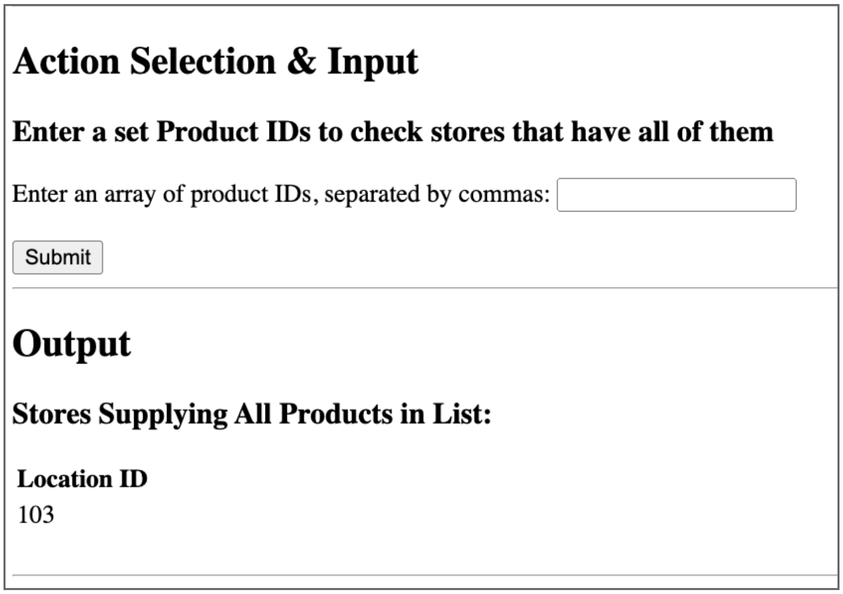
Enter an array of product IDs, separated by commas: 1, 2

Output

Stores Supplying All Products in List:

Location ID
103

- After



Action Selection & Input

Enter a set Product IDs to check stores that have all of them

Enter an array of product IDs, separated by commas:

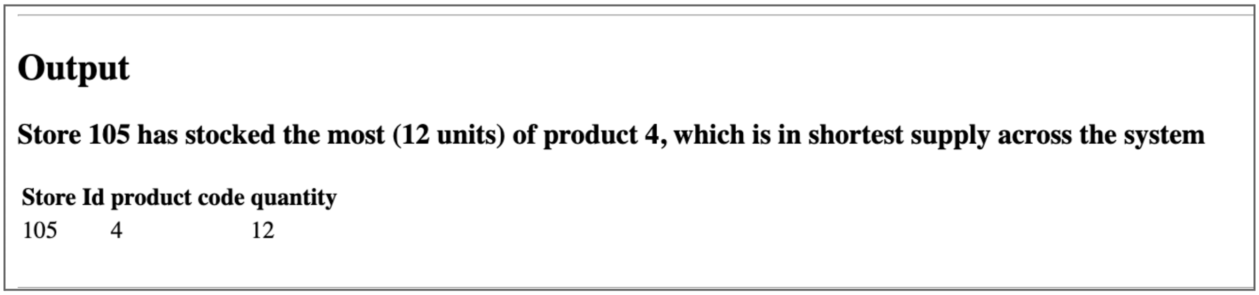
Output

Stores Supplying All Products in List:

Location ID
103

Aggregation (Nested) : Store that has the most of the item that is in the shortest supply

- Before & After



Output

Store 105 has stocked the most (12 units) of product 4, which is in shortest supply across the system

Store Id	product code	quantity
105	4	12