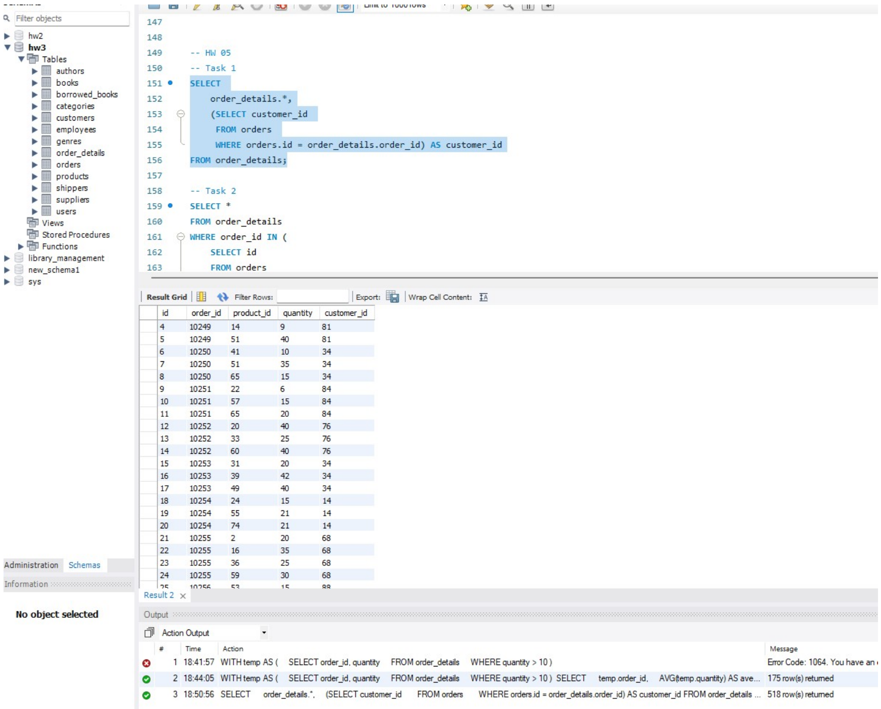
Homework 5.

Task 1.

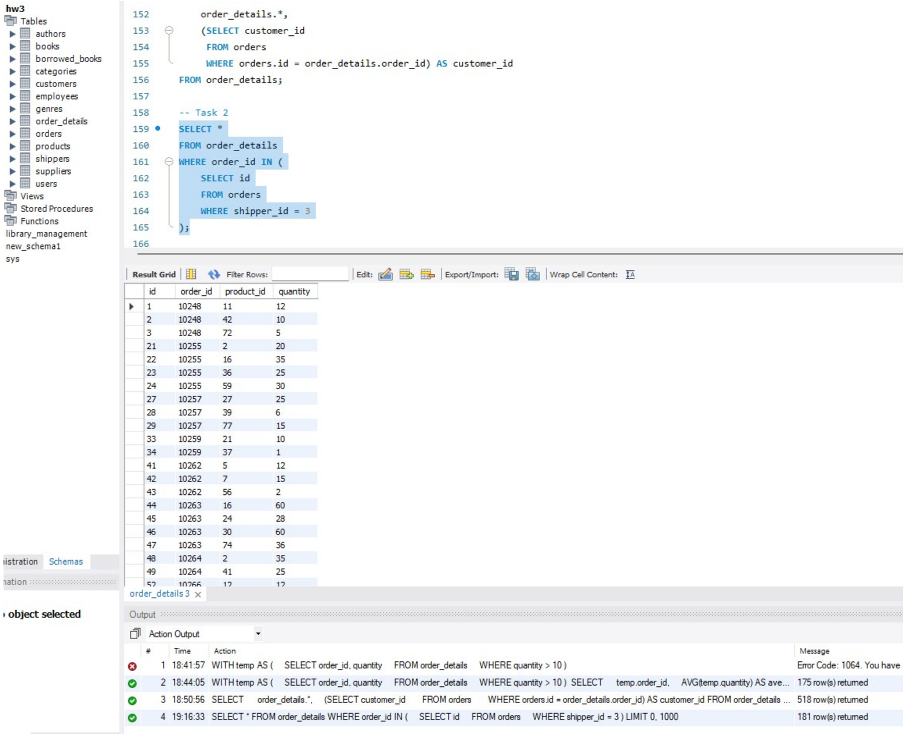


```sql


-- HW 05
-- Task 1
SELECT
    order_details.*,
    (SELECT customer_id
     FROM orders
     WHERE orders.id = order_details.order_id) AS customer_id
FROM order_details;

-- Task 2
SELECT *
FROM order_details
WHERE order_id IN (
    SELECT id
    FROM orders
```

id	order_id	product_id	quantity	customer_id
4	10249	14	9	81
5	10249	51	40	81
6	10250	41	10	34
7	10250	51	35	34
8	10250	65	15	34
9	10251	22	6	84
10	10251	57	15	84
11	10251	65	20	84
12	10252	20	40	76
13	10252	33	25	76
14	10252	60	40	76
15	10253	31	20	34
16	10253	39	42	34
17	10253	49	40	34
18	10254	24	15	14
19	10254	55	21	14
20	10254	74	21	14
21	10255	2	20	68
22	10255	16	35	68
23	10255	36	25	68
24	10255	59	30	68
25	10256	53	15	88

```
#  Time      Action                                                                                                                  Message
1  18:41:57  WITH temp AS (   SELECT order_id, quantity   FROM order_details   WHERE quantity > 10 )                                  Error Code: 1064. You have an
2  18:44:05  WITH temp AS (   SELECT order_id, quantity   FROM order_details   WHERE quantity > 10 )  SELECT   temp.order_id,   AVG(temp.quantity) AS ave...   175 row(s) returned
3  18:50:56  SELECT   order_details.*,   (SELECT customer_id   FROM orders   WHERE orders.id = order_details.order_id) AS customer_id FROM order_details ...   518 row(s) returned
```

Task 2.

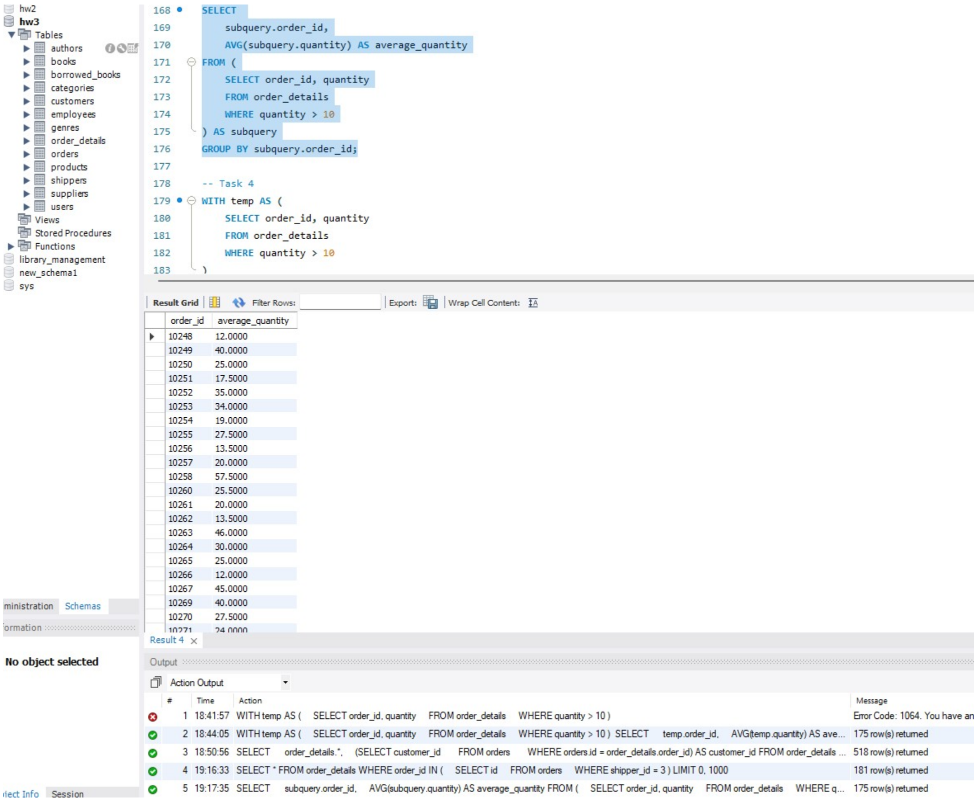


```sql
    order_details.*,
    (SELECT customer_id
     FROM orders
     WHERE orders.id = order_details.order_id) AS customer_id
FROM order_details;

-- Task 2
SELECT *
FROM order_details
WHERE order_id IN (
    SELECT id
    FROM orders
    WHERE shipper_id = 3
);

```

id	order_id	product_id	quantity
1	10248	11	12
2	10248	42	10
3	10248	72	5
21	10255	2	20
22	10255	16	35
23	10255	36	25
24	10255	59	30
27	10257	27	25
28	10257	39	6
29	10257	77	15
33	10259	21	10
34	10259	37	1
41	10262	5	12
42	10262	7	15
43	10262	56	2
44	10263	16	60
45	10263	24	28
46	10263	30	60
47	10263	74	36
48	10264	2	35
49	10264	41	25
52	10266	12	12

```
#  Time      Action                                                                                                                  Message
1  18:41:57  WITH temp AS (   SELECT order_id, quantity   FROM order_details   WHERE quantity > 10 )                                  Error Code: 1064. You have
2  18:44:05  WITH temp AS (   SELECT order_id, quantity   FROM order_details   WHERE quantity > 10 )  SELECT   temp.order_id,   AVG(temp.quantity) AS ave...   175 row(s) returned
3  18:50:56  SELECT   order_details.*,   (SELECT customer_id   FROM orders   WHERE orders.id = order_details.order_id) AS customer_id FROM order_details ...   518 row(s) returned
4  19:16:33  SELECT * FROM order_details WHERE order_id IN (   SELECT id   FROM orders   WHERE shipper_id = 3 ) LIMIT 0, 1000          181 row(s) returned
```

Task 3.

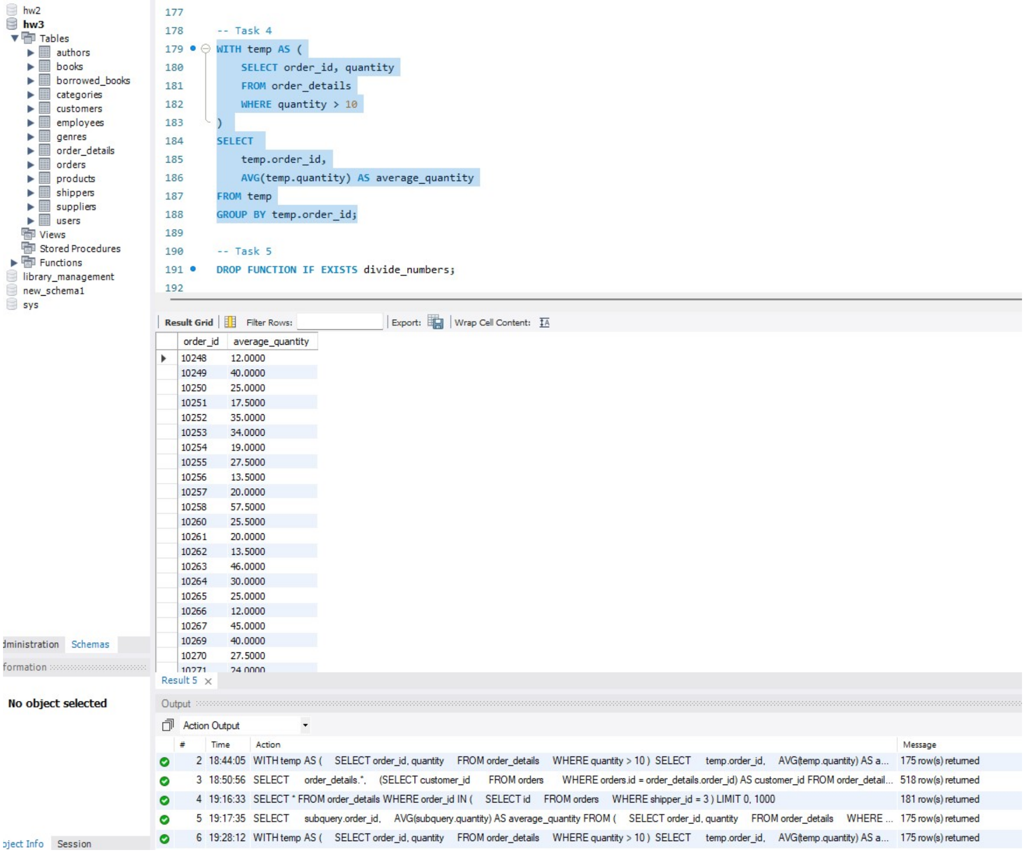


```sql
SELECT
    subquery.order_id,
    AVG(subquery.quantity) AS average_quantity
FROM (
    SELECT order_id, quantity
    FROM order_details
    WHERE quantity > 10
) AS subquery
GROUP BY subquery.order_id;

-- Task 4
WITH temp AS (
    SELECT order_id, quantity
    FROM order_details
    WHERE quantity > 10
)
```

order_id	average_quantity
10248	12.0000
10249	40.0000
10250	25.0000
10251	17.5000
10252	35.0000
10253	34.0000
10254	19.0000
10255	27.5000
10256	13.5000
10257	20.0000
10258	57.5000
10260	25.5000
10261	20.0000
10262	13.5000
10263	46.0000
10264	30.0000
10265	25.0000
10266	12.0000
10267	45.0000
10269	40.0000
10270	27.5000
10271	24.0000

```
#  Time      Action                                                                                                                   Message
1  18:41:57  WITH temp AS (   SELECT order_id, quantity   FROM order_details   WHERE quantity > 10 )                                  Error Code: 1064. You have an
2  18:44:05  WITH temp AS (   SELECT order_id, quantity   FROM order_details   WHERE quantity > 10 )  SELECT   temp.order_id,   AVG(temp.quantity) AS ave...   175 row(s) returned
3  18:50:56  SELECT   order_details.*,   (SELECT customer_id   FROM orders   WHERE orders.id = order_details.order_id) AS customer_id FROM order_details ...   518 row(s) returned
4  19:16:33  SELECT * FROM order_details WHERE order_id IN (   SELECT id   FROM orders   WHERE shipper_id = 3 ) LIMIT 0, 1000         181 row(s) returned
5  19:17:35  SELECT   subquery.order_id,   AVG(subquery.quantity) AS average_quantity FROM (   SELECT order_id, quantity   FROM order_details   WHERE q...   175 row(s) returned
```

Task 4.

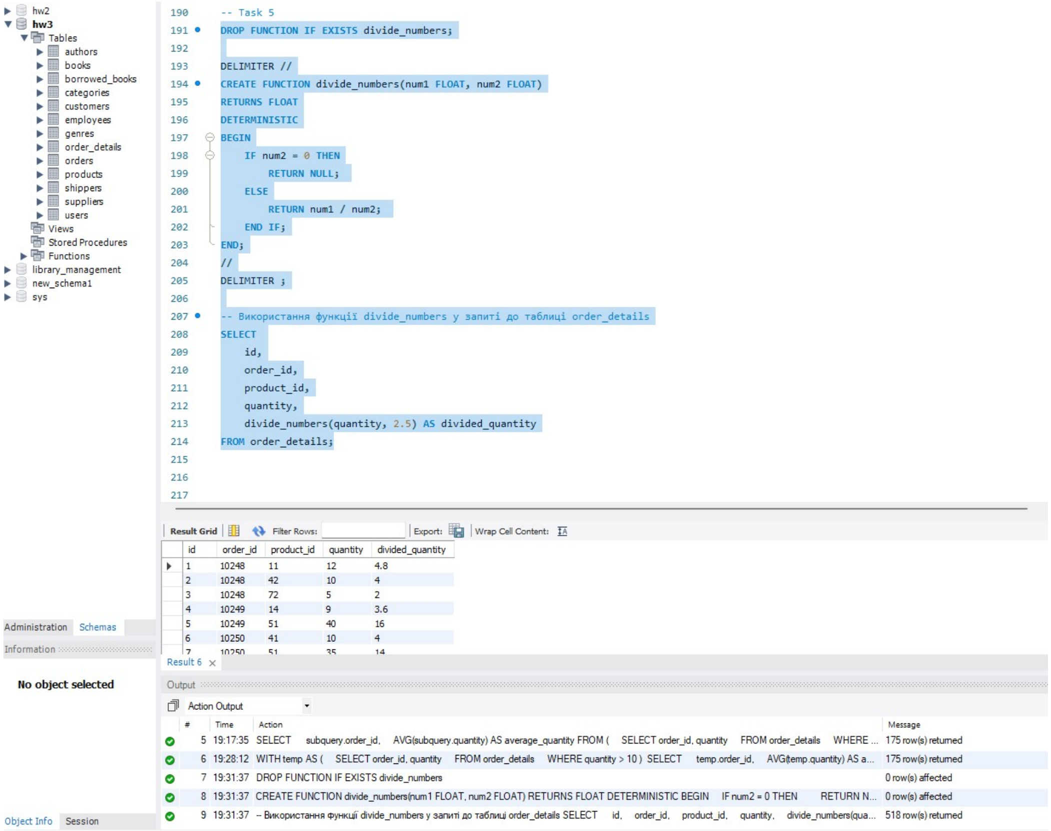


```sql
-- Task 4
WITH temp AS (
    SELECT order_id, quantity
    FROM order_details
    WHERE quantity > 10
)
SELECT
    temp.order_id,
    AVG(temp.quantity) AS average_quantity
FROM temp
GROUP BY temp.order_id;

-- Task 5
DROP FUNCTION IF EXISTS divide_numbers;
```

order_id	average_quantity
10248	12.0000
10249	40.0000
10250	25.0000
10251	17.5000
10252	35.0000
10253	34.0000
10254	19.0000
10255	27.5000
10256	13.5000
10257	20.0000
10258	57.5000
10260	25.5000
10261	20.0000
10262	13.5000
10263	46.0000
10264	30.0000
10265	25.0000
10266	12.0000
10267	45.0000
10269	40.0000
10270	27.5000
10271	24.0000

```
#  Time      Action                                                                                                                   Message
2  18:44:05  WITH temp AS (   SELECT order_id, quantity   FROM order_details   WHERE quantity > 10 )  SELECT   temp.order_id,   AVG(temp.quantity) AS a...   175 row(s) returned
3  18:50:56  SELECT   order_details.*,   (SELECT customer_id   FROM orders   WHERE orders.id = order_details.order_id) AS customer_id FROM order_detail...   518 row(s) returned
4  19:16:33  SELECT * FROM order_details WHERE order_id IN (   SELECT id   FROM orders   WHERE shipper_id = 3 ) LIMIT 0, 1000         181 row(s) returned
5  19:17:35  SELECT   subquery.order_id,   AVG(subquery.quantity) AS average_quantity FROM (   SELECT order_id, quantity   FROM order_details   WHERE ...   175 row(s) returned
6  19:28:12  WITH temp AS (   SELECT order_id, quantity   FROM order_details   WHERE quantity > 10 )  SELECT   temp.order_id,   AVG(temp.quantity) AS a...   175 row(s) returned
```

Task 5.



```sql
-- Task 5
DROP FUNCTION IF EXISTS divide_numbers;

DELIMITER //
CREATE FUNCTION divide_numbers(num1 FLOAT, num2 FLOAT)
RETURNS FLOAT
DETERMINISTIC
BEGIN
    IF num2 = 0 THEN
        RETURN NULL;
    ELSE
        RETURN num1 / num2;
    END IF;
END;
//
DELIMITER ;

-- Використання функції divide_numbers у запиті до таблиці order_details
SELECT
    id,
    order_id,
    product_id,
    quantity,
    divide_numbers(quantity, 2.5) AS divided_quantity
FROM order_details;
```

id	order_id	product_id	quantity	divided_quantity
1	10248	11	12	4.8
2	10248	42	10	4
3	10248	72	5	2
4	10249	14	9	3.6
5	10249	51	40	16
6	10250	41	10	4
7	10250	51	35	14

#	Time	Action	Message
5	19:17:35	SELECT subquery.order_id, AVG(subquery.quantity) AS average_quantity FROM (SELECT order_id, quantity FROM order_details WHERE ...	175 row(s) returned
6	19:28:12	WITH temp AS (SELECT order_id, quantity FROM order_details WHERE quantity > 10) SELECT temp.order_id, AVG(temp.quantity) AS a...	175 row(s) returned
7	19:31:37	DROP FUNCTION IF EXISTS divide_numbers	0 row(s) affected
8	19:31:37	CREATE FUNCTION divide_numbers(num1 FLOAT, num2 FLOAT) RETURNS FLOAT DETERMINISTIC BEGIN IF num2 = 0 THEN RETURN N...	0 row(s) affected
9	19:31:37	-- Використання функції divide_numbers у запиті до таблиці order_details SELECT id, order_id, product_id, quantity, divide_numbers(qua...	518 row(s) returned