

# SQL Training Exercises

This set of exercises is designed to help you strengthen your SQL skills using real-world data from our Data Warehouse (DWH). Each task focuses on common SQL operations such as filtering, joining, grouping, and aggregating data. These exercises reflect typical analytical tasks you might encounter in your day-to-day work.

All exercises are based on tables within the Data Warehouse (DWH). The tables we use most frequently are typically located in the **intermediate** or **staging** schemas, as these follow our modelling conventions. For example, to retrieve all bookings in the intermediate schema, you would use the following query:

```
SELECT *  
FROM intermediate.int_core__bookings;
```

We recommend that you only use the answers to compare your results **after** doing them yourself.

## Basic Exercises

- 1- Retrieve the first name and last name of all users (both hosts and guests).
- 2- List all listings (accommodations) that are in "New York". Return their Id, address and post code.
- 3- Get the total number of bookings made by each guest. Display the user ID, the state of the booking and the total count of bookings. Order them in descending order of booking count.
- 4- Find all users who have been rejected on their verifications. You can find the `is_verifier_rejected` status in the `int_core__verification_requests` table.
- 5- Display the total amount paid in GBP with and without taxes for all verification requests that were created in May 2024. Consider only those that have `payment_status = "Paid"`.

## Intermediate Exercises

- 6- Retrieve the top 10 deals with refunded payments. Display the deal ID and the total refunded amount, ordered by the refunded amount in descending order.
- 7- Retrieve the total payment amount for all completed verification requests related to bookings by `id_user_host = '547bff01-34e0-4c68-80de-f3180fd9bea9'` for each year from 2022 to 2024. Display the results separately by the year of payment.
- 8- Retrieve the year, month, and day (as separate columns) from the booking creation date for the **last 20 bookings** associated with the following deal IDs: `'15463726437', '20529225110', '1604445496'`. Also include the accommodation town and the first and last name of the guest in your result.

## Solutions

```
1-
SELECT first_name,
last_name
FROM intermediate.int_core__unified_user;
```

```
2-
SELECT id_accommodation,
address_line_1,
postcode
FROM staging.stg_core__accommodation
WHERE town = 'New York'
```

```
3-
SELECT id_user_guest,
booking_state,
count(*) AS booking_count
FROM intermediate.int_core__bookings icb
GROUP BY id_user_guest, booking_state
ORDER BY booking_count desc
```

```
4-
SELECT icb.id_user_guest, icuu.first_name, icuu.last_name, icuu.email
FROM intermediate.int_core__bookings icb
INNER JOIN intermediate.int_core__verification_requests icvr
ON icb.id_verification_request = icvr.id_verification_request
INNER JOIN intermediate.int_core__unified_user icuu
ON icuu.id_user = icb.id_user_guest
WHERE icvr.is_verifier_rejected IS TRUE
```

5-

```
SELECT sum(icvpv.total_amount_in_gbp) AS total_payment_in_gbp,  
sum(icvpv.amount_without_taxes_in_gbp) AS total_payment_without_taxes_in_gbp  
FROM intermediate.int_core__verification_requests icvr  
INNER JOIN intermediate.int_core__verification_payments_v2 icvpv  
ON icvr.id_verification_request = icvpv.id_verification_request  
WHERE icvr.created_date_utc BETWEEN '2024-05-01' AND '2024-05-31'  
AND icvpv. payment_status = 'Paid'
```

6-

```
SELECT icuu.id_deal,  
icb.id_accommodation,  
sum(icvpv.total_amount_in_gbp) AS total_refunded_in_gbp  
FROM intermediate.int_core__bookings icb  
INNER JOIN intermediate.int_core__unified_user icuu  
ON icuu.id_user = icb.id_user_host  
INNER JOIN intermediate.int_core__verification_requests icvr  
ON icb.id_verification_request = icvr.id_verification_request  
INNER JOIN intermediate.int_core__verification_payments_v2 icvpv  
ON icvr.id_verification_request = icvpv.id_verification_request  
WHERE icvpv. payment_status = 'Refunded'  
GROUP BY icuu.id_deal, icb.id_accommodation  
ORDER BY total_refunded_in_gbp desc  
LIMIT 100
```

7-

SELECT

icb.id\_user\_host,

SUM(CASE WHEN EXTRACT(YEAR FROM icvpv.payment\_paid\_date\_utc) = 2022 THEN  
icvpv.total\_amount\_in\_gbp ELSE 0 END) AS total\_payment\_in\_gbp\_2022,

SUM(CASE WHEN EXTRACT(YEAR FROM icvpv.payment\_paid\_date\_utc) = 2023 THEN  
icvpv.total\_amount\_in\_gbp ELSE 0 END) AS total\_payment\_in\_gbp\_2023,

SUM(CASE WHEN EXTRACT(YEAR FROM icvpv.payment\_paid\_date\_utc) = 2024 THEN  
icvpv.total\_amount\_in\_gbp ELSE 0 END) AS total\_payment\_in\_gbp\_2024

FROM intermediate.int\_core\_\_bookings icb

INNER JOIN intermediate.int\_core\_\_verification\_requests icvr

ON icb.id\_verification\_request = icvr.id\_verification\_request

INNER JOIN intermediate.int\_core\_\_verification\_payments\_v2 icvpv

ON icvr.id\_verification\_request = icvpv.id\_verification\_request

WHERE icb.id\_user\_host = '547bff01-34e0-4c68-80de-f3180fd9bea9'

AND icvpv.payment\_paid\_date\_utc BETWEEN '2022-01-01' AND '2024-12-31'

GROUP BY icb.id\_user\_host;

8-

SELECT

icb.guest\_first\_name,

icb.guest\_last\_name,

EXTRACT(YEAR FROM icb.created\_date\_utc) AS "year",

EXTRACT(MONTH FROM icb.created\_date\_utc) AS "month",

EXTRACT(DAY FROM icb.created\_date\_utc) AS "day",

sca.town

FROM intermediate.int\_core\_\_bookings icb

INNER JOIN intermediate.int\_core\_\_user\_host icuh ON icuh.id\_user\_host = icb.id\_user\_host

INNER JOIN intermediate.int\_core\_\_accommodation sca ON icb.id\_accommodation =  
sca.id\_accommodation

WHERE icuh.id\_deal IN ('15463726437', '20529225110', '1604445496')

ORDER BY icb.created\_date\_utc DESC

LIMIT 20