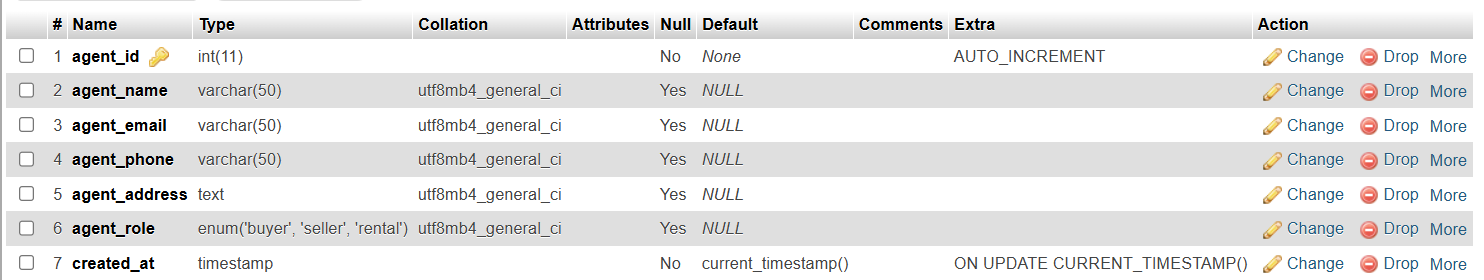
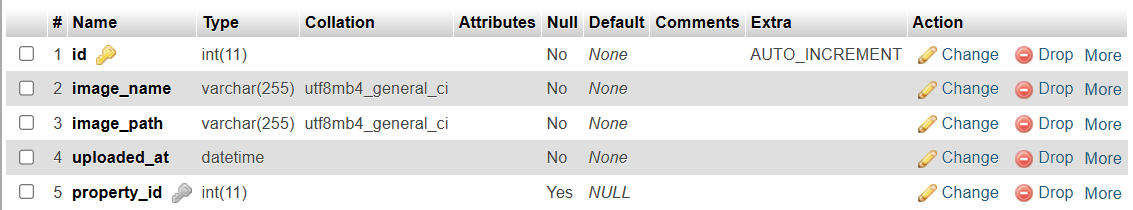
# **1. Database Schema:**

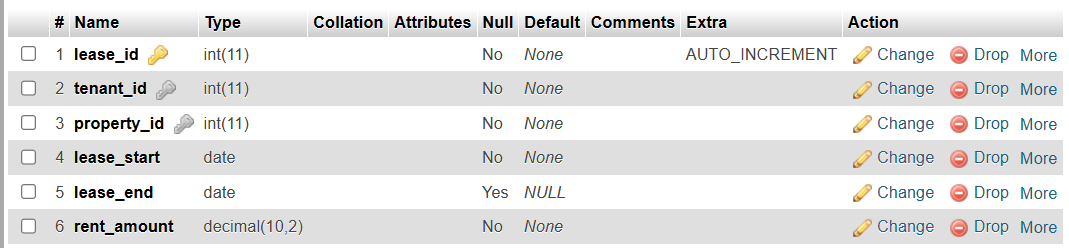
**Table Name:** agents

****

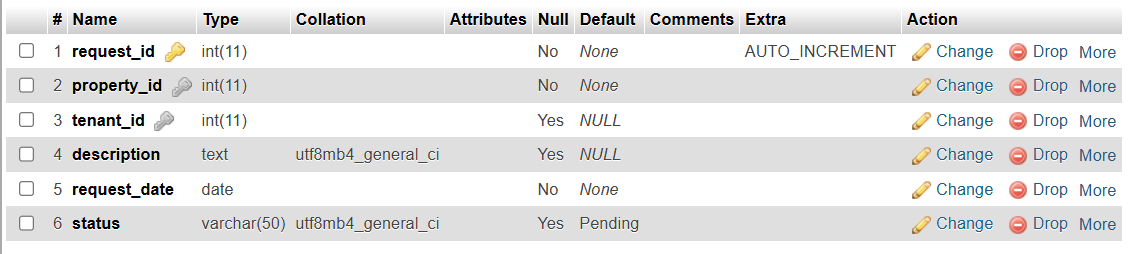
**Table Name:** images

****

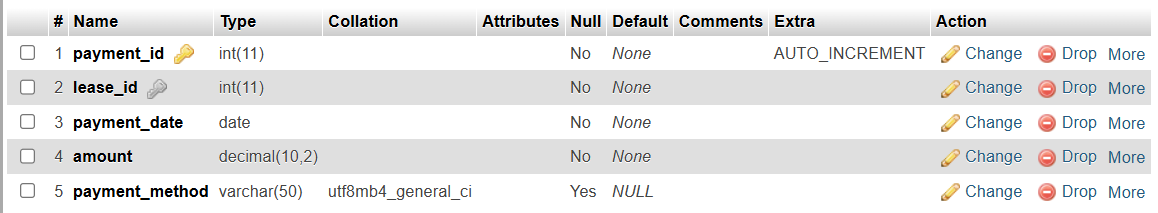
**Table Name:** leases

****

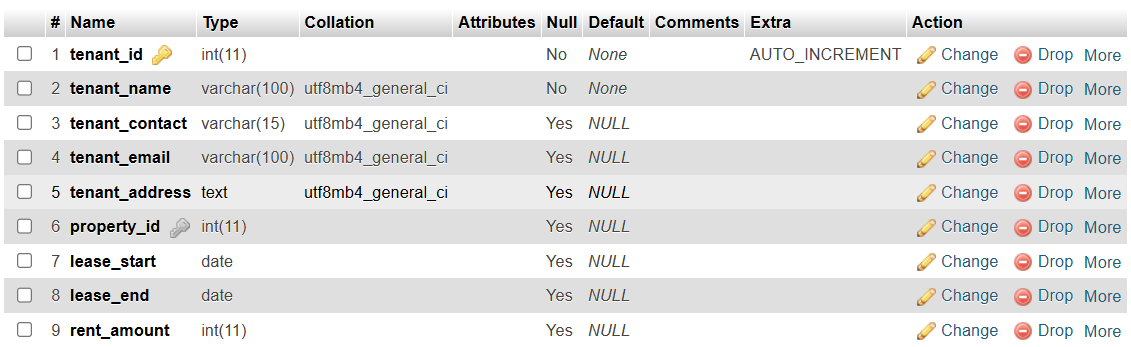
**Table Name:** maintenance requests

****

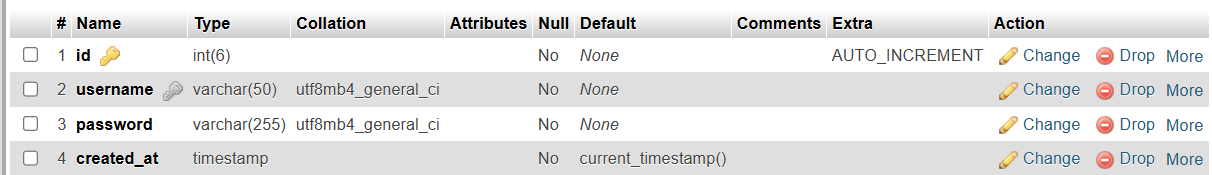
**Table Name:** payments

****

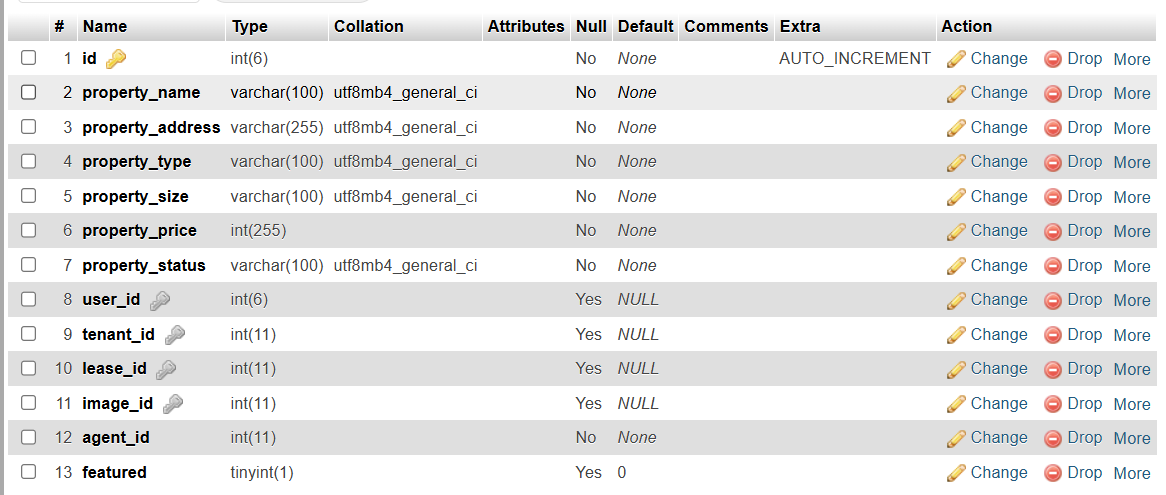
**Table Name:** tenants

****

**Table Name:** users

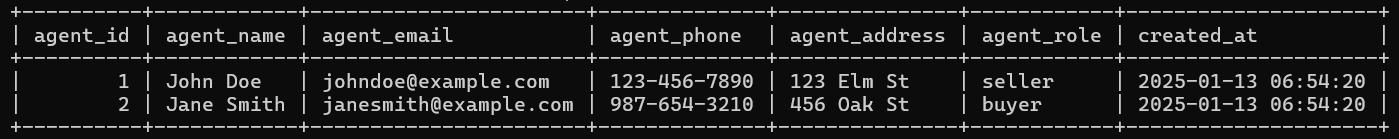
****

**Table Name:** properties

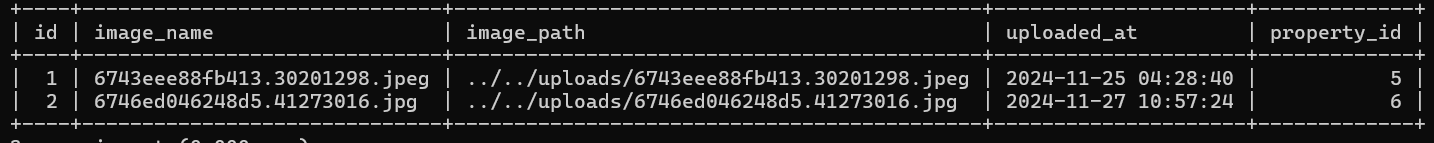
****

# **2. Data Population:**

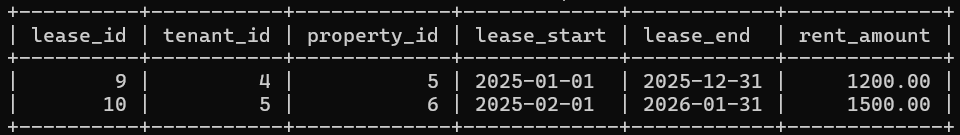
**Agents Table**

****

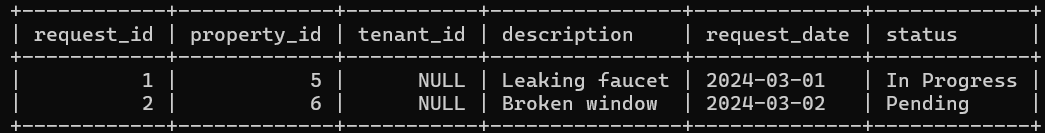
**Images Table**

****

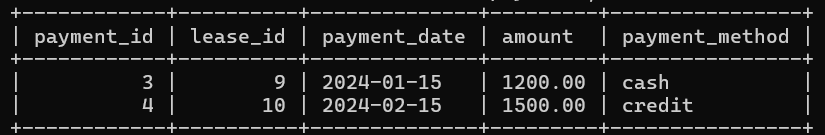
**Leases Table**

****

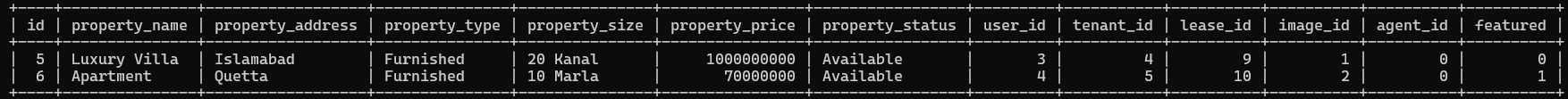
**Maintenance\_Requests Table**

****

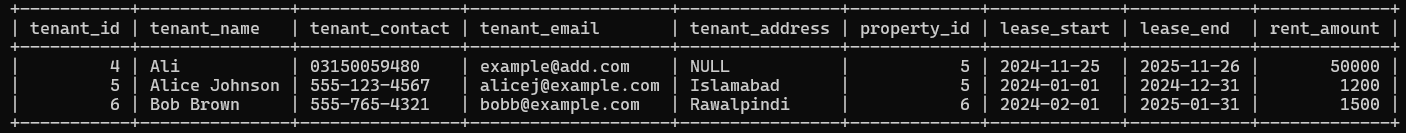
**Payments Table**

****

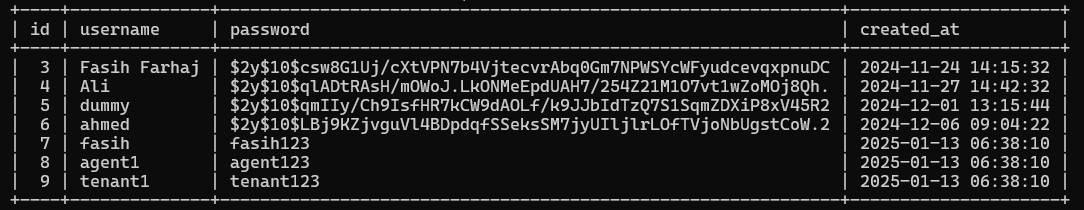
**Properties Table**

****

**Tenants Table**

****

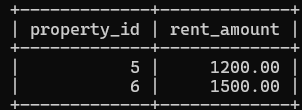
**Users Table**

****

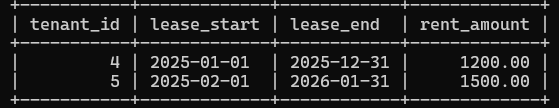
# **3. SQL Queries:**

Below are 25 SQL queries that will demonstrate the various querying capabilities of the database:

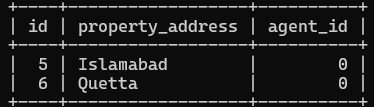
1. SELECT property\_id, rent\_amount FROM Leases;



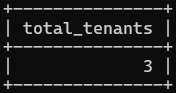
2. SELECT tenant\_id, lease\_start, lease\_end, rent\_amount FROM Leases;



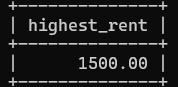
3. SELECT id, property\_address, agent\_id FROM Properties;



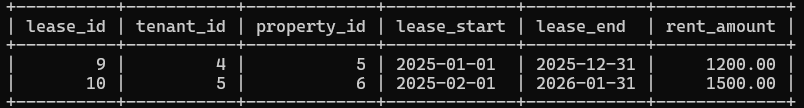
4. SELECT COUNT(\*) AS total\_tenants FROM Tenants;



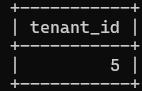
5. SELECT MAX(rent\_amount) AS highest\_rent FROM Leases;



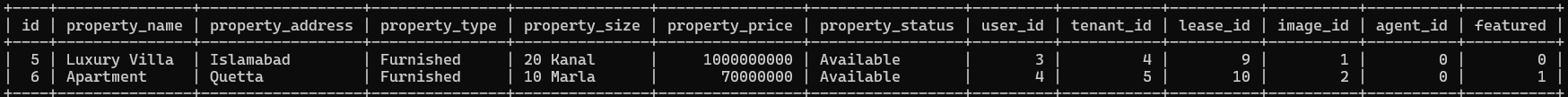
6. SELECT \* FROM Leases WHERE YEAR(lease\_start) = 2025;



7. SELECT tenant\_id FROM Leases WHERE rent\_amount > 1300;



8. SELECT \* FROM Properties WHERE property\_price > 300000;



9. SELECT \* FROM Properties

WHERE id NOT IN (SELECT property\_id FROM Leases);



10. SELECT \* FROM Leases WHERE lease\_end <= DATE\_ADD(CURDATE(), INTERVAL 6 MONTH);

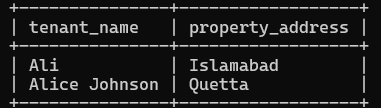


11. SELECT T.tenant\_name, P.property\_address

FROM Tenants T

JOIN Leases L ON T.tenant\_id = L.tenant\_id

JOIN Properties P ON L.property\_id = P. id;



12. SELECT P.property\_address, A.agent\_name

FROM Properties P

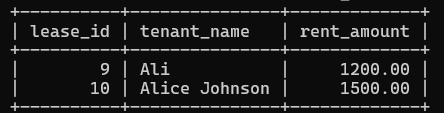
JOIN Agents A ON P.agent\_id = A.agent\_id;

****

13. SELECT L.lease\_id, T.tenant\_name, L.rent\_amount

FROM Leases L

JOIN Tenants T ON L.tenant\_id = T.tenant\_id;

****

14. SELECT DISTINCT A.agent\_name

FROM Agents A

JOIN Properties P ON A.agent\_id = P.agent\_id

JOIN Leases L ON P.id = L.property\_id

WHERE YEAR(L.lease\_start) = 2025;

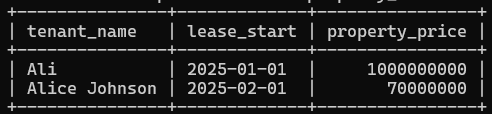
****

15. SELECT T.tenant\_name, L.lease\_start, P. property\_price

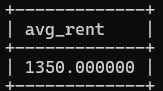
FROM Tenants T

JOIN Leases L ON T.tenant\_id = L.tenant\_id

JOIN Properties P ON L.property\_id = P. id;



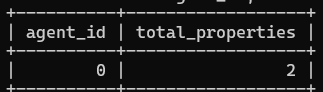
16. SELECT AVG(rent\_amount) AS avg\_rent FROM Leases;



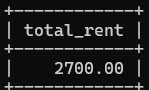
17. SELECT agent\_id, COUNT(\*) AS total\_properties

FROM Properties

GROUP BY agent\_id;



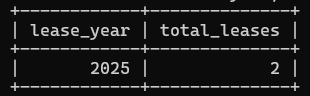
18. SELECT SUM(rent\_amount) AS total\_rent FROM Leases;



19. SELECT YEAR(lease\_start) AS lease\_year, COUNT(\*) AS total\_leases

FROM Leases

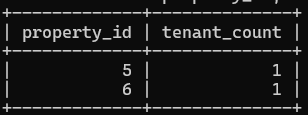
GROUP BY lease\_year;



20. SELECT property\_id, COUNT(\*) AS tenant\_count

FROM Leases

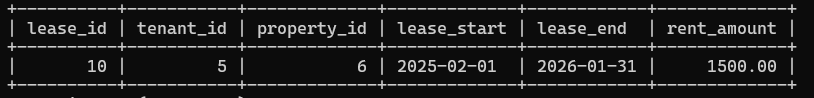
GROUP BY property\_id;



21. SELECT \*

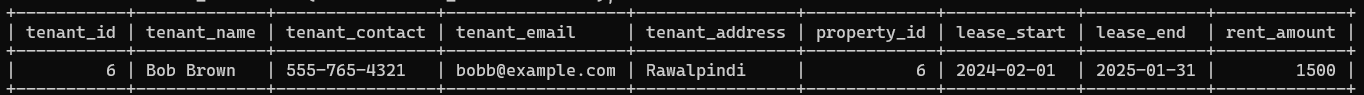
FROM Leases

WHERE rent\_amount = (SELECT MAX(rent\_amount) FROM Leases);



22. SELECT \* FROM Tenants

WHERE tenant\_id NOT IN (SELECT tenant\_id FROM Leases);



23. SELECT L.lease\_id, A.agent\_name

FROM Leases L

JOIN Properties P ON L.property\_id = P.id

JOIN Agents A ON P.agent\_id = A.agent\_id;



24. SELECT property\_id

FROM Leases

GROUP BY property\_id

HAVING COUNT(\*) > 1;

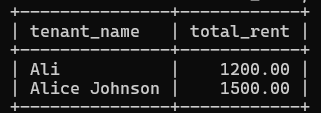


25. SELECT T.tenant\_name, SUM(L.rent\_amount) AS total\_rent

FROM Tenants T

JOIN Leases L ON T.tenant\_id = L.tenant\_id

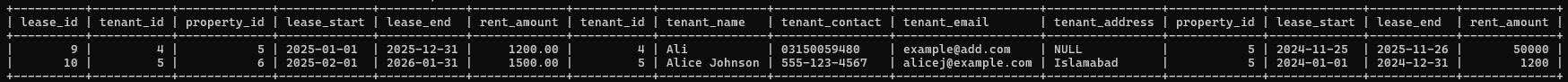
GROUP BY T.tenant\_name;



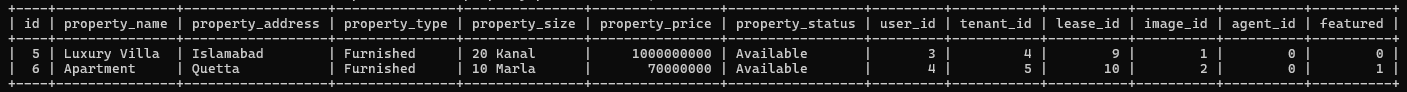
# **4. Relational Algebra Queries:**

Below are 20 Relational Algebra queries that will demonstrate the various querying capabilities of the database:

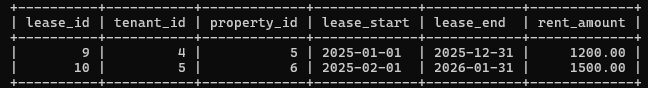
1. σ (Leases.tenant\_id = Tenants.tenant\_id) (Leases ⨝ Tenants)



2. σ price > 300000 (Properties)



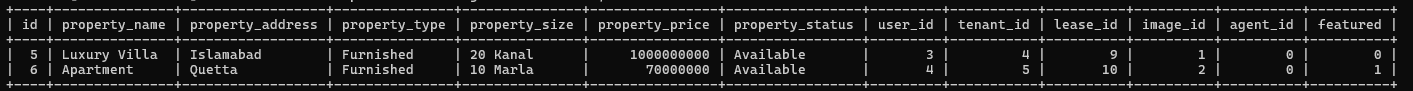
3. σ lease\_start >= '2025-01-01' ∧ lease\_start <= '2025-12-31' (Leases)



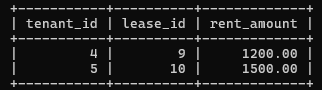
4. π tenant\_id (σ rent\_amount > 1500 (Leases))



5. σ agent\_id = 'A001' (Properties)



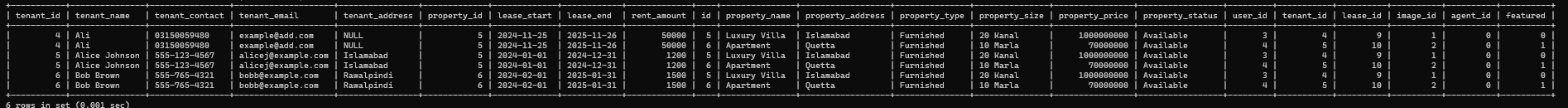
6. π tenant\_id, lease\_id, rent\_amount (Leases)



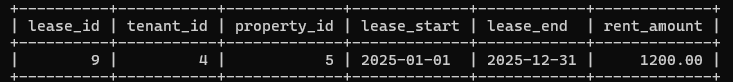
7. Properties - π property\_id (Leases)



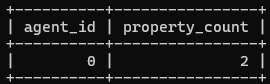
8. Tenants × Properties



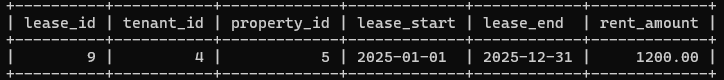
9. σ lease\_end >= '2025-01-01' ∧ lease\_end <= '2025-12-31' (Leases)



10. γ agent\_id, COUNT(property\_id) (Properties)



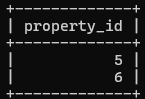
11. σ rent\_amount < AVG(rent\_amount) (Leases)



12. σ lease\_end - lease\_start > 12 months (Leases)



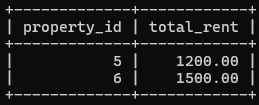
13. π property\_id (Leases ⨝ Properties)



14. σ lease\_id = 'L001' (Leases ⨝ Tenants)



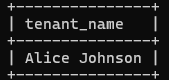
15. γ property\_id, SUM(rent\_amount) (Leases)



16. π property\_id (Properties) ÷ π property\_id (Leases)



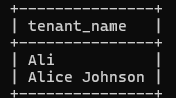
17. σ rent\_amount = MAX(rent\_amount) (Leases ⨝ Tenants)



18. γ agent\_id, COUNT(property\_id) HAVING COUNT(property\_id) > 2 (Properties)



19. σ agent\_id = '0' (Tenants ⨝ Leases ⨝ Properties ⨝ Agents)



20. γ tenant\_id, AVG(rent\_amount) (Leases)

