UNIONS AND UNION ALL EXCERSES. VINOLIAH MAKAMU

Table: online_sales

Sales_Id	Customer_Name	amount	Sales_date
1	Alice	150	2025-01-12
2	Brain	250	2025-02-05
3	Carol	300	2025-03-10
4	Danniel	220	2025-04-15
5	Emma	180	2025-05-02

Table: Store_Sales

Sale_Id	Customer_name	amount	sales_date
1	Fiona	200	2025-01-20
2	Brian	250	2025-02-08
3	George	310	2025-03-25
4	Alice	150	2025-04-18
5	Henry	270	2025-05-05

* Checking tables, column or table dataset.

```
SELECT* "online_db". "public". "onlinesales
    Limit 5;
```

```
SELECT* "online_db"."Public". "Storesales"
        Limit 5;
```

Q1. List all unique customer names from both tables using UNION.

```
SELECT
    customer_name
From "online_db"."public"."onlinesales"
    UNION
SELECT
    customer_name
FROM "online_db"."public"."storesales"
    ORDER BY customer_name;
```

Customer_Name
Alice
Brain
Carol
Danniel
Emma
Fiona
George
Henry

Q2. List all customers name from both tables using UNION ALL.

```
SELECT customer_name
FROM "online_db"."public"."OnlineSales"
UNION ALL
FROM online_db.Public.StoreSales;
```

Customer_name
Alice
Brain
Carol
Danniel
Emma
Fiona
Brain
George
Mice
Henry

NB: UNION ALL
(include dupplicate)

Q3. show all unique sale date from both
tables in ascending order!

SELECT Sale_date
FROM Online.db. Public. online Sales
    UNION
SELECT sale_date
    FROM Online_db. Public. store Sales
        ORDER BY sale_date ASC;

Sale_Date
2025-01-12
2025-01-20
2025-02-05
2025-02-08
2025-03-10
2025-03-25
2025-04-15
2025-04-18
2025-05-02
2025-05-05

4. List all sale dates (with duplicates) using UNION ALL.

Sale_date
2025-01-12
2025-02-05
2025-03-10
2025-04-15
2025-05-02
2025-01-20
2025-02-08
2025-03-25
2025-04-18
2025-05-05

~~Q5. Combine all records from both~~

Q5. Combine both table and list unique customer who made purchase greater than 250.

```
SELECT
      customer_name,
      amount
FROM online_db. public. onlineSales
   WHERE
         amount > 250

   UNION

   SELECT
         customer_name,
         amount
   FROM online_db. public. StoreSales
      WHERE   amount > 250;
```

Customer_name	Amount
Carol	300
George	310
Henry	270

Q6. Combine all records From both tables using UNION ALL.
```sql
SELECT
    customer_name,
    amount,
    Sale_date
FROM online_db. Publiz. onlinesales
    UNION ALL
SELECT
    customer_name,
    amount,
    Sales_date
FROM online_db. Publiz, storesales;
```

Customer_name	Amount	Sales_date
Alice	150	2025-01-12
Brain	250	2025-02-05
Carol	300	2025-03-10
Daniel	220	2025-03-15
Emma	180	2025-04-02
Fiona	200	2025-04-20
Brain	250	2025-02-08
George	310	2025-03-25
Alize	150	2025-04-18
Henry	270	2025-05-05

Q7. Combine both tables, but Include a new
column "source" that indicates whether the sales
was made online or store.

```sql
SELECT
    customer_name,
    amount,
    sales_date, 'online' AS source
FROM online_db. publiz. onlinesales
    UNION ALL
SELECT
    customer_name,
    amount,
    Sale_date, 'store' AS source
FROM online_db. publiz. storesales;
```

customer_name	amount	sale_date	source
Alice	150	2025-01-12	online
Bram	250	2025-02-05	online
carol	300	2025-03-10	online
Daniel	220	2025-04-15	online
Emma	180	2025-05-02	online
Fiona	200	2025-01-20	Store.
Brian	250	2025-02-08	Store
George	310	2025-03-25	store
Alice	150	2025-04-18	Store
Henry	270	2025-05-05	Store

Q8. Find all customers who appear in both
online_sales and store_sales (use UNION ALL,
GROUP BY, and Having)

```sql
SELECT
       customer_name,
COUNT (*) AS occurrences
FROM (
       SELECT
              customer_name
    FROM online_db. Public. Online sales
       UNION ALL
SELECT customer_name
FROM online_db. Public. Store sales ) AS Combined
    GROUP BY customer_name
       Having COUNT (*) > 1;
```

Customer_name	Occurrences
Alice	2
Brain	2

Q9. Combine both tables using UNION ALL and
calculate the total sales amount accross
both.

```sql
SELECT Sum (amount) AS Total_Amount.
From
        ( SELECT amount
        FROM online_db.public.onlinesales
        UNION ALL
        SELECT amount
        FROM online_db.public.storesales) AS
                                    CombineSales;
```

Total Amount
2280

Create a single query that is lists each
customer's total combined amount from both
tables (sum of all their purchases) ~~As total Spent~~

~~SELECT customer_name,~~
~~amount~~
~~FROM online_db. public. onlinesales~~
~~UNION ALL~~
~~SELECT~~
~~customer_name,~~
~~amount~~
~~FROM online_db. Public. Storesales~~

SELECT customer_name, Sum (amount) As total_spend
FROM (SELECT customer_name, amount
        FROM online_db. Public. onlinesales
    UNION ALL
    SELECT customer_name, amount
    FROM online_db. public. Storesales) As Combine
                                              Sales
        Group By customer_name
        ORDER By total_spend DESC;

Total_amount

2280