1 BrightLearn Data Analytics - SQL Exercise: UNION and UNION ALL Solutions with Expected Outcomes

1.1 Unique Customer Names

```sql
SELECT customer_name FROM online_sales
UNION
SELECT customer_name FROM store_sales;
```

Expected Output:

```
customer_name
------------
Alice
Brian
Carol
Daniel
Emma
Fiona
George
Henry
```

1.2 All Customers (Including Duplicates)

```sql
SELECT customer_name FROM online_sales
UNION ALL
SELECT customer_name FROM store_sales;
```

Expected Output:

```
customer_name
------------
Alice
Brian
Carol
Daniel
Emma
Fiona
Brian
George
Alice
Henry
```

1.3 Unique Sale Dates

```
1 SELECT sale_date FROM online_sales
2 UNION
3 SELECT sale_date FROM store_sales
4 ORDER BY sale_date ASC;
```

Expected Output:

```
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
```

1.4   All Sale Dates (Including Duplicates)

```
1 SELECT sale_date FROM online_sales
2 UNION ALL
3 SELECT sale_date FROM store_sales;
```

Expected Output:

```
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
```

1.5   High-Value Customers

```
1 SELECT customer_name, amount FROM online_sales WHERE amount > 250
2 UNION
3 SELECT customer_name, amount FROM store_sales WHERE amount > 250;
```

Expected Output:

```
customer_name | amount
--------------|-------
Carol         | 300
George        | 310
Henry         | 270
```

1.6   Combined Sales Data

```sql
1  SELECT customer_name, amount, sale_date FROM online_sales
2  UNION ALL
3  SELECT customer_name, amount, sale_date FROM store_sales;
```

Expected Output:

```
customer_name | amount | sale_date
--------------|--------|-----------
Alice         | 150    | 2025-01-12
Brian         | 250    | 2025-02-05
Carol         | 300    | 2025-03-10
Daniel        | 220    | 2025-04-15
Emma          | 180    | 2025-05-02
Fiona         | 200    | 2025-01-20
Brian         | 250    | 2025-02-08
George        | 310    | 2025-03-25
Alice         | 150    | 2025-04-18
Henry         | 270    | 2025-05-05
```

1.7   Add Sales Source Label

```sql
1  SELECT customer_name, amount, sale_date, 'Online' AS source FROM
       online_sales
2  UNION ALL
3  SELECT customer_name, amount, sale_date, 'Store' AS source FROM
       store_sales;
```

Expected Output:

```
customer_name | amount | sale_date   | source
--------------|--------|-------------|-------
Alice         | 150    | 2025-01-12 | Online
Brian         | 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
```

1.8 Customers Appearing in Both Tables

```
1 SELECT customer_name, COUNT(*) as occurrences
2 FROM (
3     SELECT customer_name FROM online_sales
4     UNION ALL
5     SELECT customer_name FROM store_sales
6 ) AS combined
7 GROUP BY customer_name
8 HAVING COUNT(*) > 1;
```

Expected Output:

```
customer_name | occurrences
--------------|------------
Alice         | 2
Brian         | 2
```

1.9 Total Combined Sales

```
1 SELECT SUM(amount) AS total_amount
2 FROM (
3     SELECT amount FROM online_sales
4     UNION ALL
5     SELECT amount FROM store_sales
6 ) AS combined_sales;
```

Expected Output:

```
total_amount
------------
2280
```

1.10 Bonus Challenge: Customer Total Spending

```
1 SELECT customer_name, SUM(amount) AS total_spent
2 FROM (
3     SELECT customer_name, amount FROM online_sales
4     UNION ALL
5     SELECT customer_name, amount FROM store_sales
6 ) AS combined_sales
7 GROUP BY customer_name;
```

Expected Output:

```
customer_name | total_spent
--------------|------------
Alice         | 300
Brian         | 500
Carol         | 300
Daniel        | 220
Emma          | 180
```

```
Fiona        | 200
George       | 310
Henry        | 270
```