Queries on Glowbox A/B Test
.CSV files separate.

Extract the analysis:

```
SELECT
   u.id AS user_id,
   u.country AS user_country,
   u.gender AS user_gender,
   g.device AS user_device_type,
   g.group AS user_test_group,
   CASE WHEN a.spent > 0 THEN 'Converted' ELSE 'Not Converted' END AS conversion,
   SUM (COALESCE(a.spent, 0)) AS total_spending
FROM users u
JOIN groups g ON u.id = g.uid
LEFT JOIN activity a ON u.id = a.uid
GROUP BY u.id, u.country, u.gender, g.device, g.group, a.spent;
```

Initial 10 Questions:

1.)

```
SELECT COUNT (uid) AS purchase_count, COUNT(DISTINCT uid) AS distinct_user
FROM activity;
```

Yes, a user may make more than one purchase at different times thereby showing up more than once. Also see 4.

2.) Left Join the two tables users.id = activity.uid

3.) Coalesce

4.)

```
Select COUNT(uid) AS all_user_activity,
        Count(DISTINCT uid)AS distinct_users,
        (2233-2094) AS Difference,
        MIN(activity.dt) AS start_date,
        MaX(activity.dt) AS end_date

From activity
        LEFT JOIN users ON activity.uid = users.id
;
```

The experiment ran from 01/25/23 - 02/06/23

5.)
```
SELECT COUNT(id)
FROM users;
```

-48,943 total users in experiment

6.)
```
SELECT
   g.group,
   COUNT(*) AS choice_count
FROM groups g
GROUP BY g.group;
```

A- 24,343 B - 24,600

7.)
```
SELECT
   COUNT(DISTINCT a.uid) AS total_purchases,
   COUNT(DISTINCT u.id) AS total_users,
   COUNT(DISTINCT a.uid) / COUNT(DISTINCT u.id) AS conversion_rate
FROM
   users u
LEFT JOIN
   activity a ON u.id = a.uid;
```

.04278

8.)
```
SELECT
   g.group AS group,
   COUNT(DISTINCT u.id) AS total_users,
   COUNT(DISTINCT CASE WHEN a.dt >= '2023-01-25' AND a.dt <= '2023-06-23'
        THEN u.id END) AS total_conversions,
   ROUND (COUNT(DISTINCT CASE WHEN a.dt >= '2023-01-25' AND a.dt <= '2023-06-23'
        THEN u.id END) * 1.0 / COUNT(DISTINCT u.id),4) AS conversion_rate

FROM groups g
LEFT JOIN users u ON g.uid = u.id
LEFT JOIN activity a ON u.id = a.uid
WHERE g.group IN ('A', 'B')
GROUP BY g.group;
```

A - .0392 B - .0463

9.)

```
With user_level as (
  SELECT u.id as user_id, g.group as test_group,
  SUM (coalesce (a.spent,0)) as amount_spent,
  CASE When SUM(a.spent)>0 then 1 Else 0 END as converted
  From users as u
  Inner Join groups as g ON u.id = g.uid
  Left Join activity as a ON u.id = a.uid
  Group BY 1,2)

  Select test_group,
         AVG(amount_spent)as average_spent
         From user_level
  Group BY 1
;
```

A - 3.3745 B - 3.3908

GloBox X-tras

Novelty effect SQL

```
-- Create a dataset with counts for users and conversions by group (A and B) by date
SELECT
    purchase_date,
    SUM(CASE WHEN ab_group = 'A' THEN 1 ELSE 0 END) AS user_count_A,
    SUM(CASE WHEN ab_group = 'B' THEN 1 ELSE 0 END) AS user_count_B,
    SUM(CASE WHEN ab_group = 'A' AND sales > 0 THEN 1 ELSE 0 END) AS conversions_A,
    SUM(CASE WHEN ab_group = 'B' AND sales > 0 THEN 1 ELSE 0 END) AS conversions_B,
    ROUND(SUM(CASE WHEN ab_group = 'A' THEN sales ELSE 0 END),4) AS sales_A,
    ROUND(SUM(CASE WHEN ab_group = 'B' THEN sales ELSE 0 END),4) AS sales_B,
    ROUND(AVG(sales), 4) AS average_sales
FROM (-- Combine data from activity and groups worksheets and include all users
SELECT
    COALESCE(a.dt, g.join_dt) AS purchase_date,
    COALESCE(a.uid, g.uid) AS user_id,
    COALESCE(a.spent, 0) AS sales,
    g.group AS ab_group
FROM groups g
LEFT JOIN activity a ON g.uid = a.uid
UNION
SELECT
```

```
    a.dt AS purchase_date,
    a.uid AS user_id,
    a.spent AS sales,
    g.group AS ab_group
FROM activity a
LEFT JOIN groups g ON a.uid = g.uid
ORDER BY purchase_date


) AS combined_data
GROUP BY purchase_date
ORDER BY purchase_date;
```

No novelty effect seen over time in the experiment with regards to metrics.