# Technical Phone Screen Questions

## Data Description

**production.publications**

| id | subdomain | created_at | country | author_id |
|---|---|---|---|---|
| 1 | barrysbuddies | @July 25, 2020 7:13 AM | US | 3 |
| 2 | sendmetomars | @July 1, 2020 1:14 AM | US | 1 |
| 3 | willyswonders | @July 2, 2020 4:17 PM | GB | 2 |

The `production.publications` table keeps track of every Substack newsletter that is created. It has a reference to a user (`publications.users`) who is the primary author of the publication via `author_id`.

**production.users**

| id | name | email | created_at | country |
|---|---|---|---|---|
| 1 | Emon Tusk | emon@example.com | @June 16, 2020 1:14 PM (EDT) | US |
| 2 | Will Gates | william@microsoft.com | @June 25, 2020 10:07 AM (EDT) | US |
| 3 | Barry Gage | bg@google.com | @June 30, 2020 12:23 PM (EDT) | US |
| 4 | Geoff Beesoz | geoff@amazon.com | @July 5, 2020 5:33 AM (EDT) | US |

The `production.users` table keeps track of all users that we know about at Substack. A `User` basically represents an email address, and a user can go on to subscribe to publications, author a publication, or both.

**production.subscriptions**

| id | publication_id | user_id | created_at | expires_at | email_disabled | type |
|---|---|---|---|---|---|---|
| 1 | 2 | 3 | @July 1, 2020 12:23 AM (EDT) | @August 1, 2020 12:23 AM (EDT) | ☐ | comp |
| 2 | 2 | 2 | @July 29, 2020 11:08 AM (EDT) | @August 29, 2020 11:08 AM (EDT) | ☐ | |
| 3 | 3 | 2 | @July 29, 2020 1:43 PM (EDT) | @July 29, 2021 1:43 PM (EDT) | ☐ | gift |
| 4 | 3 | 1 | @August 2, 2020 12:21 PM (EDT) | | ☑ | |
| 5 | 2 | 1 | @July 27, 2020 7:11 AM (EDT) | @August 3, 2022 7:11 AM (EDT) | ☐ | |
| 6 | 1 | 1 | @July 26, 2020 9:16 PM (EDT) | @August 26, 2020 9:16 PM (EDT) | ☑ | |
| ... | | | | | ☐ | |

In the `production.subscriptions` table, we log both free subscribers and paid subscriptions.

- A paid subscription remains paid until the subscription expires. There are three different type of paid subscribers:

    - A comped subscription means that the author gave away the subscription for free and the subscription has a `type` of `comp`
    - A gifted subscription means that someone else (not the person represented in this row) bought the gift for the user represented in this row and the subscription has a `type` of `gift`
    - A paying subscription is any subscription that is not a gifted or comped subscription
- A Free subscription either does not have an `expires_at` date set or was a paid subscription that has now expired. However, if a free subscription disables email from a publication, then that user is effectively unsubscribed from that publication.
- Authors are automatically subscribed to their own publication but should NOT be included in any subscription/subscriber counts.
- **Notes**:
    - Blank cells should be treated as `NULL` and not empty string or anything tricky
    - The checkboxes are represent boolean `TRUE` / `FALSE` values in the database (i.e., checked = `TRUE`)

**events.user_subscribed**

| Aa id | # publication_id | # user_id | timestamp |
|---|---|---|---|
| 1 | 2 | 3 | @July 1, 2020 2:07 AM |
| 2 | 1 | 1 | @July 26, 2020 2:10 PM |
| ... | | | |

In the `events.user_subscribed` table, we track an event (a row) that we fire after we finish making the API call to create a subscription.

**events.user_unsubscribed**

| Aa id | # publication_id | # user_id | timestamp | unsubscribe_reason |
|---|---|---|---|---|
| 1 | 1 | 1 | @August 1, 2020 4:15 AM | too_expensive |
| 2 | 1 | 2 | @August 4, 2020 12:00 AM | |
| ... | | | | |

In the `events.user_unsubscribed` table, we track an event (a row) that we fire after we finish making the API call to cancel or remove a subscription. You may assume that once a user unsubscribes, the user *never* resubscribes to that publication.

---

## Assignment

For questions 1 and 2, use the `production` tables. For questions 3 and 4, use the `events` tables.

1. How many publications are created per day?
2. For each publication (*by subdomain*):

    a. As of ***today***, how many paid and free subscriptions are there?
        i. *The output should be → 1 row per pub w/ 3 columns*
    b. Now, adjust query to split out these columns:
        - total_email_list (free + paid)
        - paying_subscription
        - comped_subscriptions
        - gifted_subscriptions
        i. *The output should be → 1 row per pub w/ 5 columns*
3. For each publication, what is the percentage change in the number of new subscriptions each week?
    - For example, pretend publication X had 10 subscriptions this week and 8 subscriptions in the week prior. Then publication X had a 25% week-over-week (`WoW`) increase in subscriptions.
4. For each publication, what is the total number of subscribers on any given day?
    - The output should be a table with 3 columns, corresponding to 1 row per day per pub and the total subscriber count on that day.