# BEES Data Science Challenge

Our logistics department wants to improve our delivery system by smartly allocating delivery resources. As part of this effort, we would like to predict the number of order days in the month for each user, and we want to be able to perform this prediction at every point of the month.

Let's look at an example of our data.
In the example below, we can see data of a user in January 2021.
This user had 2 orders on Jan' 4th and 3 orders on Jan' 25th. In total this user had **2 order days.**

| account_id | order_date | transaction_amount |
|---|---|---|
| BR_0000000000 | 2021-01-04 | 78.84 |
| BR_0000000000 | 2021-01-04 | 47.76 |
| BR_0000000000 | 2021-01-25 | 60.00 |
| BR_0000000000 | 2021-01-25 | 105.12 |
| BR_0000000000 | 2021-01-25 | 171.84 |

For this user, we would like our model to predict before Jan' 4th that we're expecting 2 days of orders (prediction = 2), and after Jan' 4th (and before Jan' 25th), we would like the model to predict that there is 1 day of orders left (prediction = 1).

For the sake of the exercise, we would like to predict the number of order days in August 2022
**Note that for some users,** the data in this file contains some of the days on which orders have been placed, for other users, all orders are missing (users with NaN values).

## Question 1

**Predict the number of order days** Do it for all the users that appear in the file '*august_with_missing_order_days.parquet*'.

Note : Your prediction is the remaining orders left for the month. If a user places 3 orders on the 10th, 20th, and 30th. We would want our predictions before the 10th to be 3, before the 20th to be 2, etc.

To do so, you can use the following:

- The already known orders in *'august_with_missing_order_days.parquet'*.
- The file ' *historical_orders.parquet* ' contains the users' historical transaction (you should use it as a training set).
- The file '*august_total_sales.parque*t ' contains the forecasted total transaction amount in August for each user. *You can assume that the forecasted value is accurate, and you can treat it as the true value for the total monthly sales - this information can help you make your predictions better but is not a must.

## Question 2 (Bonus question)

2. Let's explore the distribution of the order days

**2a**. Can you describe the distribution of the number of order days of a user (or a group of users) via a known probability distribution?
Propose a formula/density function and explain your solution.

**2b**. How could you estimate the parameter/s of this function?

**2c**. Given the formula/density function, propose a simple way to calculate the probability of having more than 4 days of orders.

**2d**. We want to estimate the time between days of orders. Propose a formula that can estimate this time.

## Submission guidelines

### Guidlines for question 1

You should submit 2 files:

- Submit **a csv file** with your predictions per user.
  - This prediction file should contain **all** the users in the file '*august_with_missing_order_days.parquet*'
  - This file should contain **2 columns only** - the first one is "account_id", the second one is "prediction".
    The shape of the file should be 32944 rows and 2 columns.
    The file should be called - "order_days_prediction.csv".
  - [reminder] The column "prediction" should contain the number of order days **left** in the month. The file '*august_with_missing_order_days . parquet* ' contains some days with orders. Do **not** consider those orders in the prediction - consider only the days left to order.

- Submit a **python notebook** (ipynb file) that shows all the steps that led to your prediction. We want to understand your thought process so make sure that this file contains all steps from EDA all the way to the final prediction.

### Guidelines for question 2 (bonus question)

Feel free to solve it in any way you like!
Your solution should be in the same notebook as question 1.