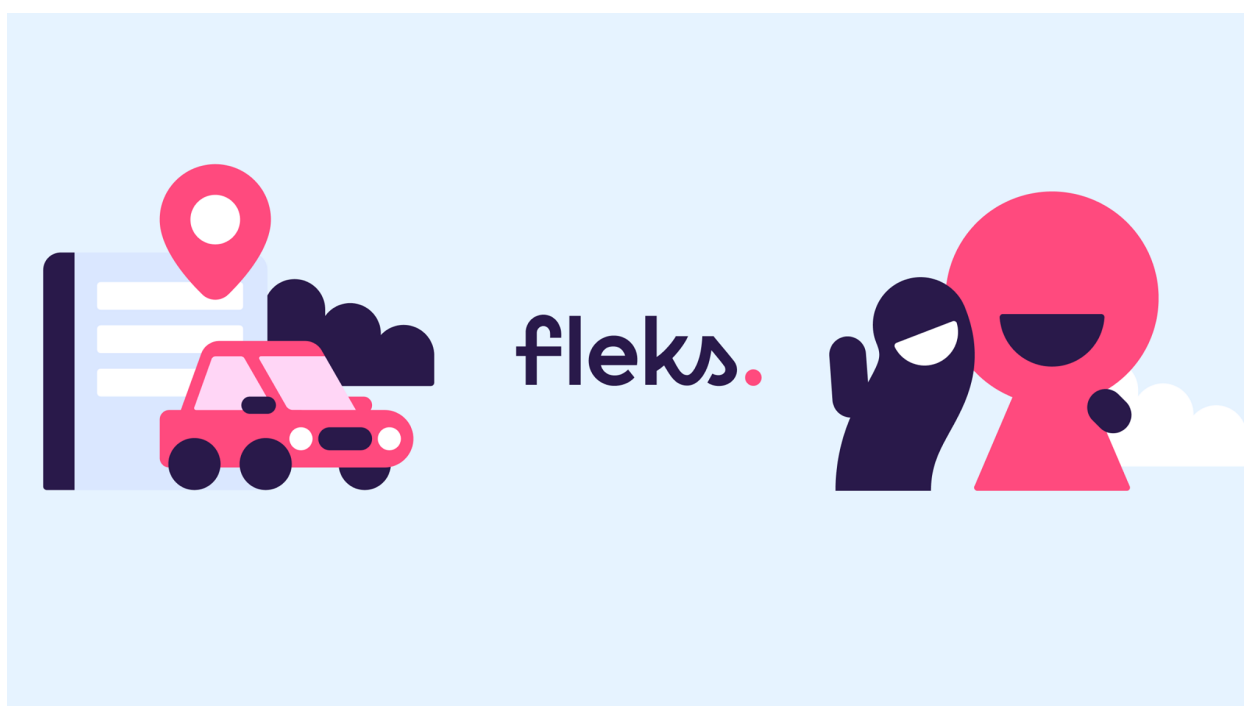




Jr Fullstack Developer

▼ Team	Tech
☰ Tags	Case



Case Oppgave

Posisjon: Junior Fullstack Developer

Kontaktperson: Jesper Karsrud (jesper@fleks.no / +47 97777264)

I anledning ditt andregangsintervju hos oss ønsker vi at du løser en case oppgave som inneholder to deler. Caset dekker en rekke reelle arbeidsoppgaver man vil møte som fullstack

developer hos oss i Fleks. Case brukes både for at vi skal kunne kartlegge dine ferdigheter på områder vi anser som relevante for å lykkes og trives hos oss, samt at du skal bli kjent med noen av arbeidsoppgavene du vil møte på.

Generelle instruksjoner:

- Det er ønskelig at du forbereder en presentasjonen i Powerpoint eller lignende format. Powerpointen skal oversendes til din kontaktperson i Fleks senest 12 timer i forkant av intervjuet.
- Estimert forberedelse tid er satt til omtrent 3-4 timer.
- Det er satt av totalt 30 min: 15 minutter til presentasjon av caset og 15 minutter til spørsmål.

Lykke til!

Task 1 — Birthdays

We would like you to create a single-page application using React. The application should sort a list of people by name and by age (not birthdate). It should be possible to select what to sort on through radio buttons, and it should be possible to select the sort order on those fields.

The initial state of the application should display an unsorted list of people and their ages.

The base of the application has been made using “create-react-app”, and is found in the zip file called “task1.zip”.

The file can be downloaded from here: https://www.icloud.com/iclouddrive/055_YStn-4lICIceAHg28GnWg#task1

Task 2 — Subscription summary in SQL

We would like you to create a SQL query that calculates how much we income we had for each active subscription between 2022-06-01 and 2022-07-01.

Each Subscription has a list of SubscriptionPeriods, which describes the timeframes that the customer is expected to pay for. At the start of every period, we will invoice whatever Plans are active at the time. A Plan is considered active as long as it is connected to the Subscription

through a SubscriptionPlan, the startedAt is equal to or before the Subscription Period start and the endedAt is either null or greater than the period start.

The results should be grouped by subscription period id and include a sum of all the subscription plans that were active at the start of each subscription period.

The database files are contained in the zip file named "task2.zip", and the files are "create-tables.sql" and "inserts.sql". The scripts are based on PostgreSQL. If you want to run these in a different database engine, there might be some issues getting them to run.

The files can be downloaded from here:

<https://www.icloud.com/iclouddrive/081CkhXofOPo6mc9i4Dh9iJwQ#task2>