

# Labb datadriven youtuber

---



## Syfte

Det är inte ovanligt att man i LIA eller på jobb kastas in på en befintlig kodbas som i många fall kan vara extremt stor. I en del fall kan det dessutom vara dåligt strukturerat och svår att jobba med.

Syftet med denna labben är att bygga vidare på en fullstackdashboard baserat på en befintlig kodbas. Strukturen och skelettet ni får utgå ifrån är relativt välstrukturerad och skalbar.

## Bakgrund

En youtuber får massor av data insamlad till sin kanal, det är faktiskt mer än man tror i första hand. Det är förmodligen ännu mer data som youtube och andra sidor samlar från sina användare när de surfar in och gör olika aktiviteter på sidan. Youtube har en sida för analytics där man som innehållsutvecklare kan se sin data och olika dashboards. Dock skulle det vara intressant att undersöka datan på egen hand och välja ut egna KPIer. Så då har jag dragit ned ett utdrag av datan som ni kommer få jobba med.

En idé är att youtubers och andra influencers behöver analysera sina sociala plattformar för att hitta trender och förstå hur det ser ut med användarnas interaktioner. På så sätt kan man göra mer datadrivna beslut och eventuellt stärka sin kanal på olika sätt.

## Uppgift 0 - setup

- a) Klona kursrepot och plocka ut filerna som hör till labben dvs 10\_lab\_overview.
- b) Lägg till följande i din gitignore för att ignorera duckdb och dbeaver filer.

```
# duckdb
*.wal
*.DS_Store

# dbeaver
```

```
.dbeaver  
.project
```

c) Det finns ett Pythonskript som läser in alla mapparna i raw\_data och ändra namnen på dessa genom att endast behålla första delen. Exempelvis **Enhetsstyp 2024-08-13\_2024-09-10 AIgineer** blir **Enhetsstyp**. Försök att förstå vad skriptet gör och kör det.

d) Förstå och kör skriptet som laddar in data från CSV-filerna in till databasen

e) Använd dbeaver och koppla mot din databasfil. Skapa SQL skript för att göra EDA på din databas och försök förstå databasen och dess innehåll. Se till att dina sql scripts hamnar i en sql folder i backend.

f) Dokumentera i en markdown de olika skripten och dess funktionalitet i backenden. Skriv i en markdownfil.

## Uppgift 1 - plocka ut intressant data

a) Skapa en schema som heter marts i din databas. Det är tabellerna i marts som serveas till din dashboard, dvs din frontend ska enbart plocka data från marts schemat och inte från andra schemas.

b) Plocka ut intressant data från andra tabeller och lägg in i olika tabeller under marts. Gör också eventuella transformationer ex alias, joins, aggregations, välja kolumner mm.

### NOTE

Du kan använda samma data som jag gjort i videorna, men ta med ytterligare data som du väljer ut

## Uppgift 2 - skapa en frontend dashboard

a) Kör `__main__.py` alternativt `streamlit run dashboard.py` för att köra streamlit dashboarden.

b) Visualisera fler KPlar än de jag tagit med. Kan visualiseras med graf eller med text.

c) Skapa åtminstone en filtrering av datan där användaren kan välja på en dropdown vad som ska filtreras ut. Här kan du filtrera på grafer eller metrics.

d) Välj själv ytterligare saker du vill implementera.

## Uppgift 3 - styling (bonus)

Go wild och styla dashboarden med CSS.

## Bedömning

I den här labben kan man bara få godkänt eller icke godkänt. För kursbetyget G behövs godkänt i både labben och projektet som kommer i nästa del. För VG krävs godkänt på labben och VG på projektet.

## Godkänt

- gjort uppgifterna korrekt
- flera relevanta git commits

- använda sig av OOP för att bygga en fullstack dashboard