Data integration og ETL

I skal arbejde med at samle data fra forskellige kilder i en database vha. ETL-processer[1].

Læringsmål

Der er 3 primære læringsmål med denne opgave.

- For det første, at få erfaring med ETL processer. Dette indebære praktisk erfaring med at implementere processerne i kode, samt erfaring med at tænke systematisk omkring hvordan man designer ETL-processer. Herunder er et vigtigt element, hvilke beslutninger og trade-offs man er nødt til at tage for at sikre konsistente dataformater og håndtere manglende værdier mm.
- For det andet, at dokumentere de beslutninger der tages samt intentionen bag de enkelte ETL-processer. Det kunne fx være ændringer af datatyper eller formatet af tekst data, samt den overordnede datastruktur før og efter ETL-processen.
- For det tredje, at øve hvordan man kommunikere relevante dele af ETL-processen til kollegaer, ledere eller andre stakeholders. Det kunne fx være en kollega der skal arbejde med at lave analyser ud fra den endelige datastrukur, eller ledere der skal forstå hvilket datagrundlag de har mulighed for at bruge til beslutningstagning.

Vi betragter de 3 læringsmål som lige vigtige, så tænk over alle 3 elementer løbende, når I arbejder med opgaven. Vi anbefaler i høj grad, at I laver løbende dokumentation.

Case beskrivelse

Forestil dig, at du arbejder som dataanalytiker hos en fiktiv virksomhed, "BikeCorp Inc.", der ønsker at få bedre mulig for at tage data-drevne beslutninger. For at facilitere dette, har du fået til opgave at konsolidere data, der ligger spredt i forskellige kilder med forskellige formater. Data skal hentes fra de nuværende kilder, transformeres og samles i en database. Der er er 3 forskellige kilder til data:

- Der ligger data lokalt på din arbejdscomputer i form af CSV-filer.
- Der findes data hos en central butik, som du kan tilgå via en API.
- Sidst har de noget af deres data liggende i en eksisterende database.

Hvis I bliver færdige og har tid, kan I bruge det I har lært i tidligere uger, til at lave en måde at tilgå data i databasen. Det kunne fx være igennem Python eller PowerBI. Selve ETL-processen og dokumentation er dog det vigtigste, så det skal være jeres fokus.

Præsentation

Opgaven skal ende ud i en præsentation på 10-15 min. I skal forestille jer, at I overleverer resultatet af jeres ETL-process til en leder eller en gruppe kollegaer, der skal bruge den i deres arbejde. Præsentationen skal indeholde:

- Et overblik over den datastruktur I er endt med. Dvs. jeres database-schema, samt eventuelt andre ting I finder relevante, afhængigt af hvilke valg I har taget undervejs,
- En introduktion til jeres dokumentation, så brugere af databasen kan tjekke forskellige ting der kunne være relevante for deres arbejde.[2]
- I kan også komme ind på hvordan I har implementeret processen i kode. Er der ting I var i tvivl om, synes var udfordrende eller problemer I fandt en elegant løsning på? Det kan både være detaljer eller en overordnet struktur I synes fungerede godt/mindre godt. Men I skal ikke gå hele jeres kode igennem i detaljer, find et par nedslag.
- Hvis I har lavet et interface til databasen i Python, PowerBI eller noget andet, kan I også komme ind på det.

Igen skal selve ETL processen og dokumentationen være i fokus.

Opgavevejledning

Her er et forslag til en måde at komme igang med opgaven.

- Start med at danne jer et overblik over datakilderne. Prøv at udtænke en plan for hvordan I vil hente og behandle data fra de 3 kilder.
- Prøv at udtænke en strategi for hvordan I vil lave dokumentation løbende. Man kan nemt blive distraheret af selve programmeringsdelen, så det kan være en god ide at have en explicit strategi og struktur for dokumentation fra starten.
- Lav et design til jeres endelige database. Tænk på hvilke tabeller I vil have, datatyper og relationer imellem tabeller. Det behøver ikke være det design I ender med, da man ofte vil blive klogere på problemet når man arbejder med det, men det kan være godt at have en overordnet struktur at sigte efter.
- Overvej at lave en skeletstruktur for jeres kode. Hvilke moduler, klasser og funktioner tænker I vil være smart at have. Det kan nogen gange være nemmere at definere nogle tomme moduler/klasser/funktioner med deskriptive navne og så fylde dem ud senere.

I må meget gerne være kreative med jeres dokumentation. Overvej om der er diagrammer, flowcharts og lign. der kan gøre det nemmere for andre at forstå jeres ETL-process.

---

1. Extract Transform Load https://en.wikipedia.org/wiki/Extract%2C_transform%2C_load ↩
2. Forestil jer fx, at en kollega har arbejdet med den tidligere datastruktur og nu skal arbejde med jeres. Hvis de analyser de har lavet tidligere, nu skal laves anderledes, hvordan finder de så relevant information til at tilpasse dem? ↩