

Mini påske opgaver ETL 1

Fleksibilitet OOP mindset

Alt efter hvordan du har bygget systemet op kan denne opgave allerede være løst. Opgaven går ud på at tage den eksisterende ETL process og gøre den mere fleksibel. Det vil sige at man skal kunne udskifte de forskellige processor ud.

Sørg for at de tre processorer ligger i 3 scripts. Et der står for selve ekstraktion, et der står for alle transformationerne og et andet der står for loading. Lav derefter et main script kaldet `main.py` der sætter alle processorne i gang.

Lav extractor om til en klasse der hedder `Extractor` eller lignende. Den skal have tre funktioner der henter data fra de enkelte kilder. Altså en funktion der hedder `extract_csv` der henter data fra en csv fil. Sørg for at de alle tre returnerer et dataframe.

Lav nu din transformer om til en klasse. Den skal have mindst en funktion kaldet `transform`, der bruger et dataframe som input og returnerer et dataframe. Behold eller lav gerne underfunktioner der hjælper med at transformere data.

Lav din loader om til en klasse. Den skal have mindst en funktion der hedder `load`, som tager et dataframe som input og returnerer en boolean. Den skal returnere `True` hvis load processen var korrekt, eller `False` hvis den fejlede.

I dit `main.py` script skal du nu initialisere de tre klasser og bruge `extract_`, `transform` og `load` funktionerne.

Lav derefter et script kaldet `transform_extra` eller lignende med en klasse med samme navn. Den skal mindst have en funktion der hedder `transform`, som tager et dataframe og returnerer et transformeret dataframe. Transformationen skal være anderledes fra den oprindelige transform klasse. F.eks kan du lave en transformation der capitaliserer alle string værdier.

Du kan nu gå tilbage til din `main.py` script og importere den anden transformer og initialiserer den korrekte klasse. Læg mærke til at du ikke behøver at ændre funktions kaldet da de har samme navn, input og retur type. Pointen er at du har lavet en ETL process som du nemt kan ændre, da de enkelte processor er uafhængige af hinanden.

Ekstra udfordring vil være at kigge transformer klasserne igennem og kun lave transformationerne ud fra hvad der ligger i dataframet. Altså at der ikke er behov for at have kendskab til dataet på forhånd. Derved kan du lave den samme type af transformationer på et hvilket som helst dataframe object uden at skulle kode en ny transformations script

Mini crud UI

Hvis man syntes det kunne værer fedt at arbejde videre med bruger venlighed kan man se på noget user interface (UI). Hvis man allerede har kendskab til UI kan man vælge at lave graphiske elementer til UI (det bliver derfor til en Graphic User Interface (GUI)).

Start med at lave et main program der tager bruger inputs i stil med [Input](#) functionen.

Lav derefter CRUD operationer som kalder jeres stored proceeedures. Det kan forkesempel være at brugeren skriver "READ" hvorefter brugeren bliver spurgt om de forskellige værdier proceduren skal bruge.

Ønsker man at lave grafiske elementer kan man bruge [Streamlit](#) eller andre GUI biblioteker hvis man kender nogen.