ETL project, startup

Project description

start:	Apr 26, 2023
examination:	May 3, 2023

Group Size

2-3 people, but you can be 1 if you very much want to.

Hand in

Hand in ONE (1), zip-file with the project on LearnPoint, as well as a GitHub link in the comments of the hand-in. The hand-in and GitHub should include a README-file with an explanation of the project, who the group members where, how to run/test the project, and what would be done if the group had more time. Also a brief presentation of your pipeline for Andreas, either online or in person. This presentation will be a simple 1-2 minute casual showing of your code, what source you used, and how it looks like when it runs.

What you are expected to achieve

Build one ETL pipeline that produces a workflow that reads **forecast data** from a weather web site and prepares it for weather statistics in a format suitable for Pandas matplotlib graphics. (See Pandassamples.zip on LearnPoint and specifically at the linechart.py example!) Pipeline chain:

[FOR HIGHER GRADES] You should orchestrate this weather pipeline by Apache*Airflow, intending to schedule it once per day, but not actually scheduling it.

- API:s under your considerations are (among others):
  - Openweatherdata API, for example the One Call API 3.0
  - SMHI Open Data API
  - Danish Meteorological Institute - Open Data
  - MET Norway Locationforecast
- Your pipeline stations should be
  - **raw:** the raw downloads saved for reference, and since they are JSON, they should be text files in JSON format
  - **harmonized:** the forecast data in JSON suitable for Pandas mathplotlib, such as

	"date":	["2022-07-12T04:00:00Z", "2022-07-12T05:00:00Z" ... *N elements*]
	"temperature":	[12.2, 15, ... *N elements*]
	"air pressure":	[1019.5, 1019.4, ... *N elements*]
	"precipitation":	[0, 0, ... *N elements*]

  - in order to test the data files suitability for presentation, you can here tweak the code in the linechart.py from the Pandassamples.
  - **cleansed:** we actually don't need to cleanse our weather data, therefore contains another copy of the data in harmonized
  - **staged:** the Pandas data tables transformed into SQL tables, such as

	date	temperature	air pressure	precipitation
	2022-07-12T04:00:00Z	12.2	1019.5	0
	2022-07-12T05:00:00Z	15	1019.4	0

  - **modelled:** this, we won't implement in this project: modelled data is for specific business intelligence purposes, and we haven't defined any such yet.