Data engineering with python

# Python data pipeline Assignment

This assignment aims to give a practical understanding of creating a data pipeline using python. It is designed to be as realistic as possible, and build practical skills building data pipelines in python.

## Case

This data comes from an US-based online lending service, where people can apply for loans to fund private matters. The company now has a slow process for granting a loan, but now they want to build a machine learning system to be able to make instant decisions of loan granting.

When the customers apply for loans, they need to state some information about themselves and the loan they want to take. They need to state:

- Loan amount ,
- Term of the loan (36/60 months)
- Employment title,
- Employment length
- Home ownership (mortgage/rent/own etc)
- Annual income
- Title, Description and purpose of the loan they want to take
- Address state

Additionally, the source data contains the following information

- Loan status
- Number of payment remarks on user
- Number of loans taken by that user

Now, based on these features, ML models are going to be trained. But before that,  the data needs to be prepared in a nice format.

## Task

The task now is to create a data pipeline preparing the data for machine learning. In this example, the data resides on Amazon S3, which is a cloud based file storage solution. Some information regarding the data:

- In the s3 bucket (details in the python file attached) a .zip file is located. In this .zip file, the data is in a json structure, one file per loan.
- The naming convention is `{loan_ID}.json`, there are 10000 unique loans included in the .zip file.

**The outline** of the case (more details in the python file attached)

**01.** Data ingestion - Ingest the data into your python environment
   a. Read from AWS S3 and extract the data
   b. Json parsing

**02.** Data cleaning - Clean the data and replace potential null values in the data
   a. Null handling

**03.** Create additional features, and a label column

**04.** Feature engineering
   a. Numerical scaling
   b. One hot encoding
   c. Bonus: textencoder

**05.** Write back to DW - AWS RDS

## How to go from here

**01.** Install python on your local machine, if you haven't done so. Download at https://www.python.org/downloads/

**02.** You are also recommended to use something else than notepad for writing code (although not needed). Either a dedicated editor (e.g. https://www.sublimetext.com/ ) or a full-fledged IDE (Integrated Development Environment, e.g. https://www.jetbrains.com/pycharm/, https://www.anaconda.com/ ).

Or, you can use jupyter notebooks for having interactive code, https://jupyter.org/install
Choose whatever you feel most comfortable with. If you have no preference we would suggest anaconda https://www.anaconda.com/products/individual .
https://docs.anaconda.com/anaconda/user-guide/getting-started/#open-nav-win

**03.** You will need to install some packages in your python environment, e.g. through `pip install` in the terminal. To follow the full assignment, run:

```
pip install sqlalchemy psycopg2 pandas sklearn nltk numpy boto3
```

**04.** Follow the outline in the python file, and fill in the blanks. If you get stuck, there are plenty of resources online. Also, you can of course contact the course representatives for more guidance. If you ask nicely, you may get some sample code…

**05.** Answers to this assignment will be presented on the last session.