Department of Computer Science and Engineering

Faculty of Engineering

University of North Texas

Assignment 1 CSCE5300 Spring 2024

Due on or before January 31, 2024

**Objective:** Design and implement an ETL process to extract data from a source, transform it according to specific business rules, and load it into a target data store.

Scenario:

The World Bank provides various data to the public in different formats. The dataset used for this assignment comes from the following source: You can download these data files from canvas.

1. [World Bank Indicators](https://data.worldbank.org/indicator)

This dataset encompasses socio-economic indicators for countries worldwide. Examples of indicators include population, arable land, and central government debt.

Tasks:

Please refer to the document named **Python-codes-that-may-be-needed.docx** that contains major operations that we may need to perform these tasks in Python.

1. **Extract (10 points):**
   * Download the provided CSV files, JSON files, database files (SQLite) containing population data. Each file represents population data for each country for a several years and GDP data electricity access data. The data may have a correlation. Each file may or may not have additional information. Some may have missing values.
   * In addition, use this API endpoint to obtain population data. <http://api.worldbank.org/v2/countries/br;cn;us;de/indicators/SP.POP.TOTL/?format=json&per_page=1000>
   * Explore possibilities of obtaining specific details with this API endpoint.
   * Identify the key information in these files.
2. **Transform (30 points):**
   * Implement a transformation process to clean and enrich the data. This may include:
     + Handling missing or invalid values.
     + Calculating additional metrics (average, sum, etc.)
     + Normalizing or standardizing data (e.g., date formats).
     + Creating new features that might be useful for analysis.
3. **Load (20 points):**
   * Design and create a relational database schema suitable for storing the cleaned and transformed data. Choose appropriate data types for each column.
   * Implement a data loading process to insert the cleaned data into the database tables.
4. **Analysis (30 points):**
   * Write SQL queries to perform the following analyses:
     + Find the country with the highest population for the year 2010 with electricity access and GDP data.
     + Find countries where the access to electricity is less than 20%.
     + Find countries and the years alongside the GDP.
5. **Documentation and Presentation (10 points):**
   * Prepare a document explaining your ETL process, including the steps you took for extraction, transformation, and loading.

Submission Guidelines:

* Submit the Code (Python or any other language), SQL scripts, database schema, and documentation.
* Clearly indicate each task and provide comments in your code.
* Demonstrate the results of your analysis in the presentation.

**Useful Code segments in Python**

Importing Pandas and NumPy

Pandas ss a data analysis and manipulation library

<https://pandas.pydata.org/docs/>

NumPy is a library to perform mathematical operations on arrays.

https://numpy.org/

# Importing necessary libraries for the project

import numpy as np

import pandas as pd

Iterate through file system

import os

for dirname, \_, filenames in os.walk('folder path'):

for filename in filenames:

print(os.path.join(dirname, filename))

Load CS file data to a DataFrame using pandas

df\_pop **=** pd.read\_csv('./world-bank-datasets/population\_data.csv')

You can use the dtype option to specify the data type of each column.

df\_pop **=** pd.read\_csv('./world-bank-datasets/population\_data.csv', dtype=’str’)

To check if the data is loaded properly

df\_pop.head()

If we cannot load the data set to a Pandas DataFrame, investigate the data set. Check if we have additional meta data at the top of the file which causes Pandas to fail. You need to skip those additional details.

df\_population = pd.read\_csv('./world-bank-datasets/population\_data.csv', skiprows=4)

If you have an empty column, drop it

df\_population = df\_population.drop('Unnamed: 62', axis=1)

Use Pandas to read JSON files

df\_json = pd.read\_json('./world-bank-datasets/population\_data.json',orient='records')

df\_json.head()

Read data from a SQLite database

import sqlite3

#connection to the database

conn = sqlite3.connect('./world-bank-datasets/population\_data.db')

#run a query

pd.read\_sql('SELECT \* FROM population\_data', conn)

If you want to deal with MySQL like databases other than SQLite, use the sqlAlchemy library. Replace the connection string “'sqlite:////home/workspace/3\_sql\_exercise/population\_data.db”

With the MySQL connection string. Search the internet for this.

from sqlalchemy import create\_engine

import os

engine = create\_engine('sqlite:////home/workspace/3\_sql\_exercise/population\_data.db')

pd.read\_sql("SELECT \* FROM population\_data", engine)

Read from an API endpoint.

import requests

import pandas as pd

url = 'http://api.worldbank.org/v2/countries/br;cn;us;de/indicators/SP.POP.TOTL/?format=json&per\_page=1000'

r = requests.get(url)

r.json()

**Refer to Pandas documentation to find out data cleaning options that you may use.**