Zadania:

1. Napisz skrypt (bash lub powershell), który **automatycznie** wykona następujące kroki:
   1. pobierze plik z internetu: [*https://home.agh.edu.pl/~wsarlej/Customers\_Nov2021.zip*](https://home.agh.edu.pl/~wsarlej/Customers_Nov2021.zip)
   2. rozpakuje pobrany plik, hasło do pliku to: **agh**
   3. sprawdzi poprawność pliku odrzucając błędne wiersze do pliku *Customers\_Nov2021.****bad****\_${TIMESTAMP}.* Warunki walidacji:
      * zignoruj puste linie,
      * porównaj plik wejściowy z plikiem *Customers\_old.csv*, pozostaw te wiersze, które nie występują w pliku *Customers\_old.csv*: <https://home.agh.edu.pl/~wsarlej/Customers_old.csv> (pobrać manualnie),
   4. w bazie danych PostgreSQL utworzy tabelę *CUSTOMERS\_${NUMERINDEKSU}* tylko jeśli taka tabela już nie istnieje (warunek *if not exists*),
   5. załaduje dane ze zweryfikowanego pliku do tabeli *CUSTOMERS\_${NUMERINDEKSU},*
   6. przeniesie przetworzony plik do podkatalogu *PROCESSED* dodając prefix *${TIMESTAMP}\_* do nazwy pliku,
   7. wyśle email zawierający nst. raport: *temat*: *CUSTOMERS LOAD - ${TIMESTAMP},* *treść*:
      * liczba wierszy w pliku pobranym z internetu,
      * liczba poprawnych wierszy (po czyszczeniu),
      * liczba duplikatów w pliku wejściowym,
      * ilość danych załadowanych do tabeli *CUSTOMERS\_${NUMERINDEKSU}.*
   8. uruchomi kwerendę SQL, która znajdzie imiona i nazwiska klientów, którzy mieszkają w promieniu 50 kilometrów od punktu: *41.39988501005976, -75.67329768604034* (funkcja [ST\_DistanceSpheroid](https://postgis.net/docs/ST_Distance_Spheroid.html)) i zapisze je do tabeli *BEST\_CUSTOMERS\_${NUMERINDEKSU},*
   9. wyeksportuje zawartość tabeli *BEST\_CUSTOMERS\_${NUMERINDEKSU}* do pliku csv o takiej samej nazwie jak tabela źródłowa,
   10. skompresuje wyeksportowany plik csv,
   11. wyśle skompresowany plik do adresata poczty razem z raportem o treści: data ostatniej modyfikacji, ilość wierszy w pliku csv,
   12. po każdym kroku (a-k) sprawdzi, czy komenda wykonała się poprawnie oraz zaloguje, że taki krok miał miejsce i wykonał się prawidłowo (wraz z krótkim opisem oraz czasem w którym nastąpiło wykonanie komendy (np. *20200101121212 – Archiving Step - Successful*).

Zalogowane zdarzenia będą zapisywane w pliku *PROCESSED/nazwa\_skryptu\_${TIMESTAMP}.log*

1. Skrypt, plik .log z uruchomienia oraz kwerendę SQL proszę udostępnić w swoim repozytorium na github.

Dodatkowe uwagi odnośnie skryptu:

* na początku skryptu proszę dodać krótki changelog wraz z opisem skryptu oraz datą utworzenia,
* proszę w skrypcie sparametryzować wartości typu: numer indexu, automatycznie wyznaczana dzisiejsza data ${TIMESTAMP} w formacie MMDDYYYY, nazwa logu, adres pobranego pliku, hasło do archiwum, sql hostname, sql userid, sql hasło (hasła nie mogą być zapisywane do pliku .log!),
* wszelkie walidacje/zliczenia ilości wierszy muszą być automatyczne i działać również w przypadku otrzymania pliku z innymi danymi,
* adresat automatycznych maili to Państwa adres email (proszę uważać szczególnie w przypadku, kiedy wysyłanie realizowane jest w pętli, proszę nie używać innych adresów email niż swój, proszę nie przesadzać z ilością wysyłanych maili),
* proszę dodać odpowiednie nagłówki do tworzonych plików i pomijać je przy zliczaniu ilości wierszy lub ładowaniu danych do bazy,
* proszę rozsądnie korzystać z plików tymczasowych (i usuwać je kiedy nie są potrzebne),
* proszę dopisać krótki komentarz przed kluczowymi krokami/komendami (np. *#* *Archiwizowanie*),
* proszę nie używać wyrażeń regularnych.

Komponenty/opcje, które mogą okazać się przydatne: Bash; Powershell; linux: wget, cat, sed, awk, cut, psql, sort, uniq, diff, sdiff, mv, cp, wc, zip, unzip, mailx, echo, $?, >, <; sql: CREATE TABLE IF EXISTS, ST\_DistanceSpheroid