Assignment 3

```
Install the necessary libraries
```

In [1]:
```
! pip install pyarrow
! pip install fastavro
! pip install pygeohash
! pip install snappy
! pip install jsonschema
! pip install google
! pip install protobuf
```

```
Requirement already satisfied: pyarrow in c:\users\basiab1\anaconda3\lib\si
te-packages (9.0.0)

WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)


Requirement already satisfied: numpy>=1.16.6 in c:\users\basiab1\anaconda3
\lib\site-packages (from pyarrow) (1.20.3)

WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)

Requirement already satisfied: fastavro in c:\users\basiab1\anaconda3\lib\s
ite-packages (1.6.1)
Requirement already satisfied: pygeohash in c:\users\basiab1\anaconda3\lib
\site-packages (1.2.0)

WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
```

```
Requirement already satisfied: snappy in c:\users\basiab1\anaconda3\lib\sit
e-packages (3.0.3)
Requirement already satisfied: plink>=2.4.1 in c:\users\basiab1\anaconda3\l
ib\site-packages (from snappy) (2.4.1)
Requirement already satisfied: cypari>=2.3 in c:\users\basiab1\anaconda3\li
b\site-packages (from snappy) (2.4.1)
Requirement already satisfied: decorator in c:\users\basiab1\anaconda3\lib
\site-packages (from snappy) (5.1.0)
Requirement already satisfied: FXrays>=1.3 in c:\users\basiab1\anaconda3\li
b\site-packages (from snappy) (1.3.5)
Requirement already satisfied: snappy-manifolds>=1.1.2 in c:\users\basiab1
\anaconda3\lib\site-packages (from snappy) (1.1.2)
Requirement already satisfied: spherogram>=2.1 in c:\users\basiab1\anaconda
3\lib\site-packages (from snappy) (2.1)
Requirement already satisfied: pypng in c:\users\basiab1\anaconda3\lib\site
-packages (from snappy) (0.20220715.0)
Requirement already satisfied: ipython>=5.0 in c:\users\basiab1\anaconda3\l
ib\site-packages (from snappy) (7.29.0)
Requirement already satisfied: six in c:\users\basiab1\anaconda3\lib\site-p
ackages (from cypari>=2.3->snappy) (1.16.0)

WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)

Requirement already satisfied: future in c:\users\basiab1\anaconda3\lib\s
ite-packages (from cypari>=2.3->snappy) (0.18.2)
Requirement already satisfied: matplotlib-inline in c:\users\basiab1\anac
onda3\lib\site-packages (from ipython>=5.0->snappy) (0.1.2)
Requirement already satisfied: prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.
0.0 in c:\users\basiab1\anaconda3\lib\site-packages (from ipython>=5.0->s
nappy) (3.0.20)
Requirement already satisfied: traitlets>=4.2 in c:\users\basiab1\anaconda
a3\lib\site-packages (from ipython>=5.0->snappy) (5.1.0)
Requirement already satisfied: pygments in c:\users\basiab1\anaconda3\lib
\site-packages (from ipython>=5.0->snappy) (2.10.0)
Requirement already satisfied: jedi>=0.16 in c:\users\basiab1\anaconda3\l
ib\site-packages (from ipython>=5.0->snappy) (0.18.0)
Requirement already satisfied: pickleshare in c:\users\basiab1\anaconda3
\lib\site-packages (from ipython>=5.0->snappy) (0.7.5)
Requirement already satisfied: backcall in c:\users\basiab1\anaconda3\lib
\site-packages (from ipython>=5.0->snappy) (0.2.0)
Requirement already satisfied: colorama in c:\users\basiab1\anaconda3\lib
\site-packages (from ipython>=5.0->snappy) (0.4.4)
Requirement already satisfied: setuptools>=18.5 in c:\users\basiab1\anaco
nda3\lib\site-packages (from ipython>=5.0->snappy) (58.0.4)
Requirement already satisfied: parso<0.9.0,>=0.8.0 in c:\users\basiab1\an
```

```
aconda3\lib\site-packages (from jedi>=0.16->ipython>=5.0->snappy) (0.8.2)
Requirement already satisfied: wcwidth in c:\users\basiab1\anaconda3\lib
\site-packages (from prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0->ipytho
n>=5.0->snappy) (0.2.5)
Requirement already satisfied: networkx in c:\users\basiab1\anaconda3\lib
\site-packages (from spherogram>=2.1->snappy) (2.6.3)
Requirement already satisfied: knot-floer-homology>=1.1 in c:\users\basia
b1\anaconda3\lib\site-packages (from spherogram>=2.1->snappy) (1.2)
Requirement already satisfied: jsonschema in c:\users\basiab1\anaconda3\l
ib\site-packages (3.2.0)
Requirement already satisfied: pyrsistent>=0.14.0 in c:\users\basiab1\ana
conda3\lib\site-packages (from jsonschema) (0.18.0)
Requirement already satisfied: attrs>=17.4.0 in c:\users\basiab1\anaconda
3\lib\site-packages (from jsonschema) (21.2.0)
Requirement already satisfied: setuptools in c:\users\basiab1\anaconda3\l
ib\site-packages (from jsonschema) (58.0.4)
Requirement already satisfied: six>=1.11.0 in c:\users\basiab1\anaconda3
\lib\site-packages (from jsonschema) (1.16.0)

WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)

Requirement already satisfied: google in c:\users\basiab1\anaconda3\lib\sit
e-packages (3.0.0)

WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)


Requirement already satisfied: beautifulsoup4 in c:\users\basiab1\anaconda3
\lib\site-packages (from google) (4.10.0)
Requirement already satisfied: soupsieve>1.2 in c:\users\basiab1\anaconda3
\lib\site-packages (from beautifulsoup4->google) (2.2.1)
Requirement already satisfied: protobuf in c:\users\basiab1\anaconda3\lib\s
ite-packages (3.20.1)

WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anacond
a3\lib\site-packages)
```

```
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anacond
a3\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anacond
a3\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anacond
a3\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anacond
a3\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anacond
a3\lib\site-packages)
```

Import libraries and define common helper functions

In [2]:
```
import os
import sys
import gzip
import json
from pathlib import Path
import csv

import pandas as pd
import pyarrow as pa
from pyarrow.json import read_json
import pyarrow.parquet as pq
import fastavro
import pygeohash
import snappy
import jsonschema
from jsonschema.exceptions import ValidationError
```

In [3]:
```
current_dir = Path(os.getcwd()).absolute()
schema_dir = current_dir.joinpath('schemas')
results_dir = current_dir.joinpath('results')
results_dir.mkdir(parents=True, exist_ok=True)
```

In [4]:
```
def read_jsonl_data_lm():
    src_data_path = r'D:\VZW Twinsburg\Tuition assistance\Bellevue University
    with open(src_data_path, 'rb') as f_gz:
        with gzip.open(src_data_path, 'rb') as f:
            records = [json.loads(line) for line in f.readlines()]


    return records
```

Load the records from routes.jsonl.gz file located in the local folder

In [5]:
```
records = read_jsonl_data_lm()
```

In [6]:
```
# Let's see how the data looks like
records[0:2]
```

Out[6]:

```
[{'airline': {'airline_id': 410,
   'name': 'Aerocondor',
   'alias': 'ANA All Nippon Airways',
   'iata': '2B',
   'icao': 'ARD',
   'callsign': 'AEROCONDOR',
   'country': 'Portugal',
   'active': True},
  'src_airport': {'airport_id': 2965,
   'name': 'Sochi International Airport',
   'city': 'Sochi',
   'country': 'Russia',
   'iata': 'AER',
   'icao': 'URSS',
   'latitude': 43.449902,
   'longitude': 39.9566,
   'altitude': 89,
   'timezone': 3.0,
   'dst': 'N',
   'tz_id': 'Europe/Moscow',
   'type': 'airport',
   'source': 'OurAirports'},
  'dst_airport': {'airport_id': 2990,
   'name': 'Kazan International Airport',
   'city': 'Kazan',
   'country': 'Russia',
   'iata': 'KZN',
   'icao': 'UWKD',
   'latitude': 55.606201171875,
   'longitude': 49.278701782227,
   'altitude': 411,
   'timezone': 3.0,
   'dst': 'N',
   'tz_id': 'Europe/Moscow',
   'type': 'airport',
   'source': 'OurAirports'},
  'codeshare': False,
  'equipment': ['CR2']},
 {'airline': {'airline_id': 410,
   'name': 'Aerocondor',
   'alias': 'ANA All Nippon Airways',
   'iata': '2B',
   'icao': 'ARD',
   'callsign': 'AEROCONDOR',
   'country': 'Portugal',
   'active': True},
  'src_airport': {'airport_id': 2966,
   'name': 'Astrakhan Airport',
   'city': 'Astrakhan',
   'country': 'Russia',
   'iata': 'ASF',
   'icao': 'URWA',
   'latitude': 46.2832984924,
   'longitude': 48.0063018799,
```

```
 'altitude': -65,
 'timezone': 4.0,
 'dst': 'N',
 'tz_id': 'Europe/Samara',
 'type': 'airport',
 'source': 'OurAirports'},
'dst_airport': {'airport_id': 2990,
 'name': 'Kazan International Airport',
 'city': 'Kazan',
 'country': 'Russia',
 'iata': 'KZN',
 'icao': 'UWKD',
 'latitude': 55.606201171875,
 'longitude': 49.278701782227,
 'altitude': 411,
 'timezone': 3.0,
 'dst': 'N',
 'tz_id': 'Europe/Moscow',
 'type': 'airport',
 'source': 'OurAirports'},
'codeshare': False,
'equipment': ['CR2']}]
```

3.1

3.1.a JSON Schema

In [7]:

```
def validate_jsonl_data(records):
    schema_path = schema_dir.joinpath('routes-schema.json')
    with open(schema_path) as f:
        schema = json.load(f)

    with open('validation_csv_path', 'w', encoding='utf-8') as f:
        for i, record in enumerate(records):
            try:
                ## TODO: Validate record
                jsonschema.validate(record, schema)
                pass
            except ValidationError as e:
                ## Print message if invalid record
                f.write(f"Error: {e.message}; failed validating {e.validator}
                print(e)
                pass

validate_jsonl_data(records)
```

3.1.b Avro

In [9]:
```
def create_avro_dataset(records):
    schema_path = schema_dir.joinpath('routes.avsc')
    data_path = results_dir.joinpath('routes.avro')
    ## TODO: Use fastavro to create Avro dataset
    with open(schema_path, 'r') as f1:
        schema = json.loads(f1.read())
    parsed_schema = fastavro.parse_schema(schema)
    ## create dataset
    with open(data_path, 'wb') as out:
        fastavro.writer(out, parsed_schema, records)

create_avro_dataset(records)
```

In [10]:
```
# Check if file was created successfully
# view contents
data_path = results_dir.joinpath('routes.avro')
with open(data_path, mode = 'rb') as f:
    reader = fastavro.reader(f)
    records = [r for r in reader]
    df = pd.DataFrame.from_records(records)
    print(df.head())
```

```
                                             airline  \
0  {'airline_id': 410, 'name': 'Aerocondor', 'ali...
1  {'airline_id': 410, 'name': 'Aerocondor', 'ali...
2  {'airline_id': 410, 'name': 'Aerocondor', 'ali...
3  {'airline_id': 410, 'name': 'Aerocondor', 'ali...
4  {'airline_id': 410, 'name': 'Aerocondor', 'ali...

                                         src_airport  \
0  {'airport_id': 2965, 'name': 'Sochi Internatio...
1  {'airport_id': 2966, 'name': 'Astrakhan Airpor...
2  {'airport_id': 2966, 'name': 'Astrakhan Airpor...
3  {'airport_id': 2968, 'name': 'Chelyabinsk Bala...
4  {'airport_id': 2968, 'name': 'Chelyabinsk Bala...

                                         dst_airport  codeshare  stops  \
0  {'airport_id': 2990, 'name': 'Kazan Internatio...      False      0
1  {'airport_id': 2990, 'name': 'Kazan Internatio...      False      0
2  {'airport_id': 2962, 'name': 'Mineralnyye Vody...      False      0
3  {'airport_id': 2990, 'name': 'Kazan Internatio...      False      0
4  {'airport_id': 4078, 'name': 'Tolmachevo Airpo...      False      0

  equipment
0     [CR2]
1     [CR2]
2     [CR2]
3     [CR2]
4     [CR2]
```

3.1.c Parquet

In [10]:
```
def create_parquet_dataset():
    src_data_path = 'D:/VZW Twinsburg/Tuition assistance/Bellevue University_
    parquet_output_path = results_dir.joinpath('routes.parquet')
    with gzip.open(src_data_path, 'rb') as f:
        table = read_json(f)
    pq.write_table(table, parquet_output_path)

create_parquet_dataset()
```

In [11]:
```
# Check if file was created successfully
# view contents
parquet_output_path = results_dir.joinpath('routes.parquet')
pqFile = pq.ParquetFile(parquet_output_path)
pqFile.metadata
```

Out[11]:
```
<pyarrow._parquet.FileMetaData object at 0x0000020E57E41450>
  created_by: parquet-cpp-arrow version 9.0.0
  num_columns: 38
  num_rows: 67663
  num_row_groups: 1
  format_version: 2.6
  serialized_size: 7567
```

3.1.d Protocol Buffers

In [12]:
```
pwd
```

Out[12]:
```
'D:\\VZW Twinsburg\\Tuition assistance\\Bellevue University_MSDS\\DSC 650
\\DSC650_Big Data\\dsc650\\dsc650\\assignments\\assignment03'
```

In [13]:
```
ls
```

```
 Volume in drive D is Data
 Volume Serial Number is B077-C018

 Directory of D:\VZW Twinsburg\Tuition assistance\Bellevue University_MSDS
\DSC 650\DSC650_Big Data\dsc650\dsc650\assignments\assignment03

09/18/2022  10:05 PM    <DIR>          .
09/18/2022  10:05 PM    <DIR>          ..
09/17/2022  05:15 PM    <DIR>          .ipynb_checkpoints
09/02/2022  08:37 PM                 0 __init__.py
09/17/2022  10:27 PM    <DIR>          __pycache__
09/17/2022  05:15 PM             8,518 Assignment 3.ipynb
09/18/2022  10:05 PM            73,775 assignment03_BasitAbdul.ipynb
09/18/2022  11:31 AM    <DIR>          results
09/02/2022  08:37 PM            15,981 routes_pb2.py
09/02/2022  08:37 PM    <DIR>          schemas
09/18/2022  10:04 PM                 0 validation_csv_path
               5 File(s)         98,274 bytes
               6 Dir(s)  791,718,875,136 bytes free
```

In [14]:
```
%env
```

```
 'COMPUTERNAME': 'ILRMEDNW53',
 'COMSPEC': 'C:\\windows\\system32\\cmd.exe',
 'DRIVERDATA': 'C:\\Windows\\System32\\Drivers\\DriverData',
 'HOMEDRIVE': 'P:',
 'HOMEPATH': '\\',
 'HOMESHARE': '\\\\win.eng.vzwnet.com\\greatlakes1\\homes\\basiab1',
 'LOCALAPPDATA': 'C:\\Users\\basiab1\\AppData\\Local',
 'LOGONSERVER': '\\\\TXSLADDCP3',
 'NUMBER_OF_PROCESSORS': '16',
 'OS': 'Windows_NT',
 'PATH': 'C:\\Users\\basiab1\\Anaconda3;C:\\Users\\basiab1\\Anaconda3\\Li
brary\\mingw-w64\\bin;C:\\Users\\basiab1\\Anaconda3\\Library\\usr\\bin;
C:\\Users\\basiab1\\Anaconda3\\Library\\bin;C:\\Users\\basiab1\\Anaconda3
\\Scripts;c:\\Program Files (x86)\\RSA SecurID Token Common;c:\\Program F
iles\\RSA SecurID Token Common;C:\\windows\\system32;C:\\windows;C:\\wind
ows\\System32\\Wbem;C:\\windows\\System32\\WindowsPowerShell\\v1.0\\;
C:\\windows\\System32\\OpenSSH\\;C:\\Program Files (x86)\\Intel\\Intel(R)
Management Engine Components\\DAL;C:\\Program Files\\Intel\\Intel(R) Mana
gement Engine Components\\DAL;c:\\Program Files (x86)\\Pulse Secure\\VC14
2.CRT\\X64\\;c:\\Program Files (x86)\\Pulse Secure\\VC142.CRT\\X86\\;
```

In [16]:
```
! pip install protobuf
```

```
Requirement already satisfied: protobuf in c:\users\basiab1\anaconda3\lib\s
ite-packages (3.20.1)

WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
WARNING: Ignoring invalid distribution -rotobuf (c:\users\basiab1\anaconda3
\lib\site-packages)
```

In [17]:

```
sys.path.insert(0, os.path.abspath('routes_pb2'))

import routes_pb2

def _airport_to_proto_obj(airport):
    obj = routes_pb2.Airport()
    if airport is None:
        return None
    if airport.get('airport_id') is None:
        return None

    obj.airport_id = airport.get('airport_id')
    if airport.get('name'):
        obj.name = airport.get('name')
    if airport.get('city'):
        obj.city = airport.get('city')
    if airport.get('iata'):
        obj.iata = airport.get('iata')
    if airport.get('icao'):
        obj.icao = airport.get('icao')
    if airport.get('altitude'):
        obj.altitude = airport.get('altitude')
    if airport.get('timezone'):
        obj.timezone = airport.get('timezone')
    if airport.get('dst'):
        obj.dst = airport.get('dst')
    if airport.get('tz_id'):
        obj.tz_id = airport.get('tz_id')
    if airport.get('type'):
        obj.type = airport.get('type')
    if airport.get('source'):
        obj.source = airport.get('source')

    obj.latitude = airport.get('latitude')
    obj.longitude = airport.get('longitude')

    return obj


def _airline_to_proto_obj(airline):
    obj = routes_pb2.Airline()
    if not airline.get('name'):
        return None
    if not airline.get('airline_id'):
        return None

    obj.airline_id = airline.get('airline_id')
    obj.name = airline.get('name')

    if airline.get('alias'):
        obj.alias = airline.get('alias')
    ## TODO

    return obj
```

```
def create_protobuf_dataset(records):
    routes = routes_pb2.Routes()
    for record in records:
        route = routes_pb2.Route()
        ## TODO: Implement the code to create the Protocol Buffers Dataset
        airline = _airport_to_proto_obj(record.get('airline', {}))
        if airline:
            route.airline.CopyFrom(airline)
        src_airport = _airport_to_proto_obj(record.get('src_airport', {}))
        if src_airport:
            route.src_airport.CopyFrom(src_airport)
        dst_airport = _airport_to_proto_obj(record.get('dst_airport', {}))
        if dst_airport:
            route.dst_airport.CopyFrom(dst_airport)

        if record.get('codeshare'):
            route.codeshare = record.get('codeshare')
        else:
            route.codeshare = False

        if record.get('stops'):
            route.stops = record.get('stops')

        equipment = record.get('equipment')

        if len(equipment) > 1:
            for i, v in enumerate(equipment):
                route.equipment.append(v)
        else:
            equpiment = record.get('equipment')

        routes.route.append(route)

    data_path = results_dir.joinpath('routes.pb')

    with open(data_path, 'wb') as f:
        f.write(routes.SerializeToString())

    compressed_path = results_dir.joinpath('routes.pb.snappy')

    with open(compressed_path, 'wb') as f:
        f.write(snappy.compress(routes.SerializeToString()))

create_protobuf_dataset(records)
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_28332/3996100642.py in <module>
     98         f.write(snappy.compress(routes.SerializeToString()))
     99
--> 100 create_protobuf_dataset(records)

~\AppData\Local\Temp/ipykernel_28332/3996100642.py in create_protobuf_datas
et(records)
     96
     97     with open(compressed_path, 'wb') as f:
---> 98         f.write(snappy.compress(routes.SerializeToString()))
```

```
     99
    100 create_protobuf_dataset(records)

AttributeError: module 'snappy' has no attribute 'compress'
```

3.2

3.2.a Simple Geohash Index

In [18]:

```
def create_hash_dirs(records):
    geoindex_dir = results_dir.joinpath('geoindex')
    geoindex_dir.mkdir(exist_ok=True, parents=True)
    hashes = []
    ## TODO: Create hash index
    for record in records:
        src_airport = record.get('src_airport', {})
        if src_airport:
            latitude = src_airport.get('latitude')
            longitude = src_airport.get('longitude')
            if latitude and longitude:
                hashes.append(pygeohash.encode(latitude, longitude))
    hashes.sort()

    three_letter = sorted(list(set([entry[:3] for entry in hashes])))

    hash_index = {value: [] for value in three_letter}

    for record in records:
        geohash = record.get('geohash')
        if geohash:
            hash_index[geohash[:3]].append(record)

    for key, values in hash_index.items():
        output_dir = geoindex_dir.joinpath(str(key[:1])).joinpath(str(key[:2]
        output_dir.mkdir(exist_ok=True, parents=True)
        output_path = output_dir.joinpath('{}.jsonl.gz'.format(key))
        with gzip.open(output_path, 'w') as f:
            json_output = '\n'.join([json.dumps(value) for value in values])
            f.write(json_output.encode('utf-8'))

create_hash_dirs(records)
```

3.2.b Simple Search Feature

In [19]:
```
def airport_search(latitude, longitude):
    ## TODO: Create simple search to return nearest airport
    a = pygeohash.encode(latitude, longitude)
    dist = 0
    name = ''

    for i, record in enumerate(records):
        src_airport = record.get('src_airport', {})
        if src_airport:
            lat = src_airport.get('latitude')
            long = src_airport.get('longitude')
            airport_name = src_airport.get('name')
            if lat and long:
                a1 = pygeohash.encode(lat, long)

                dist_n = pygeohash.geohash_approximate_distance(a, a1)
                if i==0:
                    dist = dist_n
                    name = airport_name
                else:
                    if dist > dist_n:
                        dist = dist_n
                        name = airport_name
    print(name)
```

In [20]:
```
airport_search(41.1499988, -95.91779)
```

```
Eppley Airfield
```

In [21]:
```
# Airport search for O'Hare International Airport
# Coordinates searched in google: 41.9803° N, 87.9090° W
airport_search(41.9803, -87.9090)
```

```
Chicago O'Hare International Airport
```