CPTS 415 – Assignment 1
Mark Shinozaki

1. Give one example of Big Data application you know. Use the example to clearly explain each of the five Big V's.

- Social Media Analytics
- Volume :
    - Social media platforms like Facebook, twitter, and Instagram generate an enormous volume of data every second. Users post text, images, videos and interact with each other, leading to petabytes of data daily, these examples show how massive amounts of data are generated and an example of volume.
- Velocity:
    - Data on social media is created and updated at an incredibly high velocity. New posts, comments, likes, and shares do happen in real-time, and trending topics change rapidly. From concept or thought to public viewing, posts are published at high velocity and reach viewership quickly.
- Variety:
    - Social media data comes in various forms, including text, images, videos, and links. It's also semi-structured with metadata with timestamps, geolocation, and user demographics. Images, videos and text are all forms of media that show case variety in the terms of content that end up online.
- Veracity:
    - Verifying the accuracy of social media data can be challenging, as it may include fake news, spam, or misinformation. Data quality is a significant concern, and machine learning algorithms are used to detect and filter out unreliable information. Current systems utilized by social media platforms require a level of comparable veracity so that there is safety online.
- Value:
    - Organizations use social media analytics to gain insights into customers' sentiment, brand perception, and market trends. By analyzing this data, they can make informed decisions about product development, marketing strategies, and customer engagement, leading to improved business outcomes. Value being extracted from the data in this case is meaningful and used to analyze more data and create a better experience for the user.

2. [Relational Data Model]

    - OpenFlights Airports Database, Contains over 10,000 airports, train stations and ferry terminals spanning the globe
        - Airport ID – OpenFlight identifier
        - Name – Name of airport, may or may not contain city name
        - City – Main city served by airport, may be spelled differently from name
        - Country – Country or territory where airport is located
        - IATA – 4 letter IATA code, Null if not assigned/unknown
        - ICAO – 4-letter ICAO code
        - Latitude – Decimal degrees, usually to six significant digits. Neg is south, positive is north
        - Longitude – Decimal degrees, neg is west, positive is east
        - Altitude – in feet

- Timezone - hours offset from UTC. Fractional hours are expressed as decimals
- DST – Daylight savings time. One of E (Europe, A (US/CA), S (south America), O (Australia), Z (New zealand), N (None), U (Unknown)
- TZ – Timezone
- Type – Type of airport
- Source – Source of this data

a. Consider the following terms: relation schema, relational database schema, domain, attribute, attribute domain, relation instance. Give what these terms are with the Above Airport database

   - Relation Schema: In OpenFlights Airport database, a relation schema would correspond to the structure or blueprint of each of the three tables (Airport, Airline, and route). For example, the "Airport" relation schema would include the attributes you mentioned: Airport ID, Name, City, Country, IATA, ICAO, Latitude, Longitude, Altitude, Timezone, DST, Tz database time zone, type and source.

   - Relational Database Schema: The relational database schema for the OpenFlights Airport Database would encompass all three tables (Airport, Airline, and route), including their attributes and relationships. It describes how these tables are related to each other within the database.

   - Domain: In the context of this database, a domain could refer to the set of valid values for specific attributes. For example, the "country" attribute has a domain of countries and territories, while the "IATA" attributes has a domain of three-letter airport codes.

   - Attribute: Attributes are the individual column in each table, such as "Name," "City," "Country," etc. Each attribute represents a specific piece of information about airports, airlines, or routes.

   - Attribute Domain: Attribute domains specify the valid values for each attribute. For instance, the "Altitude" attribute may have a domain of positive numeric values representing altitude in meters.

   - Relation Instance: A relation instance in this context would be a specific set of rows and data within each table. For example, the "Airport" relation instance would contain rows for individual airports, each with its unique data.

b. There are three databases in the OpenFlight dataset: Airport, Airline, and Route. Give the schema of these three databases and mark the primary keys, foreign keys and provide examples of functional dependencies you identified over the three tables. [you may draw a diagram to illustrate the schema, PKs, FKs and FDs]

Airport Database Schema:

- Airport (Relation Schema):
    - Airport ID (PK)
    - Name
    - City
    - Country
    - IATA
    - ICAO
    - Latitude
    - Longitude
    - Altitude
    - Timezone
    - DST
    - Tz database time zone
    - Type
    - Source

- Primary Key (PK): Airport ID
- Functional Dependencies (Examples):
    - {IATA} -> {Name, City, Country}
    - {ICAO} -> {Name, City, Country}

- Airline Database Schema:
    - Airline (Relation Schema):
        - Airline ID (PK)
        - Name
        - IATA
        - ICAO
        - Country
        - ICAO code
        - Active
    - Primary Key (PK): Airline ID
    - Functional Dependencies (Examples):
        - {IATA} -> {Name, City, Country}
        - {ICAO} -> {Name, City, Country}
    - Route Database Schema:
        - Route (Relation Schema):
            - Airline (FK, references Airline, Airline ID)
            - Airline ID (FK, references Airline, Airline ID)
            - Source Airport (FK, references Airport, Airport ID)

- Source Airport ID (FK, references Airport, Airport ID)
- Destination Airport (FK, references Airport, Airport ID)
- Destination Airport ID (FK, references Airport, Airport ID)
- Codeshare
- Stops
- Equipment

- Primary Key (PK): None (a combination of Airline, Source Airport, and Destination Airport may be used as a composite PK)
- Foreign Keys (FKs): Airline, Airline ID, Source Airport, Source Airport ID, Destination Airport, Destination Airport ID
- Functional Dependencies (Examples):
  - {Airline} -> {Airline ID}
  - {Source Airport} -> {Source Airport ID, City}
  - {Destination Airport} -> {Destination Airport ID, City}

3. [Functional Dependencies] Recall Armstrong's axioms.
   - Reflexivity rule: $if\ Y\ \subseteq X\ then\ X \rightarrow Y$
   - Augmentation rule: $if\ X \rightarrow Y\ then\ XZ\ \rightarrow YZ$
   - Transitivity rule: $if\ X \rightarrow Y\ and\ Y\ \rightarrow Z\ then\ X \rightarrow Z$

   a. Give two examples for using Armstrongs inference rules to induce new FDs from the set of FDs you designed in question 2(b)

First Example(Augmentation rule):

- {IATA} -> {Name, City, Country}
- Augmentation Rule: $if\ X \rightarrow Y\ then\ XZ\ \rightarrow YZ\ for\ any\ x$
- *We can augment the FD as follows:
  - {IATA, Latitude} -> {Name, City, Country, Latitude}
  - {IATA, Longitude} -> {Name, City, Country, Longitude}
  - {IATA, Altitude} -> {Name, City, Country, Altitude}

Second Example(Transitive rule):

- {Source Airport} -> {Source Airport ID, City}
- Transitive Rule: $if\ X \rightarrow Y\ and\ Y \rightarrow Z,\ then\ X\ \rightarrow\ Z$
- Using the rule,
  - {Source Airport} -> {Source Airport ID, City}
  - {Source Airport ID, City} -> {Source Airport ID, Country}

b. Prove the following inference rules also hold, using FD definition and Armstrong's Axioms.
   i. Decomposition rule $if\ X \rightarrow YZ\ then: X \rightarrow Y\ and\ X\ \rightarrow\ Z$

   - $X \rightarrow YZ\ (Given)$
   - $X \rightarrow Y\ (Augmentation: X \rightarrow YZ\ imples\ X \rightarrow Y)$
   - $X \rightarrow Z\ (Augmentation: X \rightarrow YZ\ imples\ X \rightarrow Z)$

   ii. Psuedo transitivity: $if\ X \rightarrow Y\ and\ YW \rightarrow Z,\ then: XW\ \rightarrow\ Z$

   - $X \rightarrow Y\ (Given)$
   - $YW \rightarrow Y\ (Given)$
   - $XY \rightarrow YZ\ (Augmentation: YW \rightarrow Z\ imples\ XY \rightarrow YZ)$

   - $XY \rightarrow YZ\ (Given)$
   - $XW \rightarrow YZ\ (Transitive: XY \rightarrow YZ\ imples\ XW \rightarrow YZ)$

4. Relational Algebra (Consider the following database schema:
   - Which theaters feature "Zootopia"
     - Result <- Theater (σ Title='Zootopia' (Schedule X Movies))
   - List the names and addresses of theaters featuring a file directed by Steven Spielberg
     - SpielbergMovies <- σ Director='Steven Spielberg' (Movies)
     - Result <- ∏ Theater, Address (Location (thetaJoin) (σ Title=Movies. Title (Schedule (thetaJoin) SpielbergMovies)))
   - What are the address and phone number of the Le Champo Theater?
     - Result <- ∏ Address, Phone number (σ Theater='Le Champo' (Location))
   - List pairs of actors that acted in the same movie
     - MoviePair <- Movies X Movies
     - Result <- ∏ Actor1, Actor2 (σ MoviesPair.Title = Movies.Title ^ Actor1 < Actor2 (MoviesPair))