- What is a Data Lake? Explain its benefits, how it differs from a data warehouse, and how it might benefit a client.

  Data lake and Data warehouse are widely used terms but they cannot be used interchangeably. They essentially are used for the same purpose of storing data. The key difference is the structure and the end user leveraging that data.

  A Data Lake is where raw data is stored whereas a Data Warehouse is where we keep structured clean data for a specific purpose.

  Data Lake has unprocessed data and usually will be required by a data scientist and specialised tools to convert it into something useful

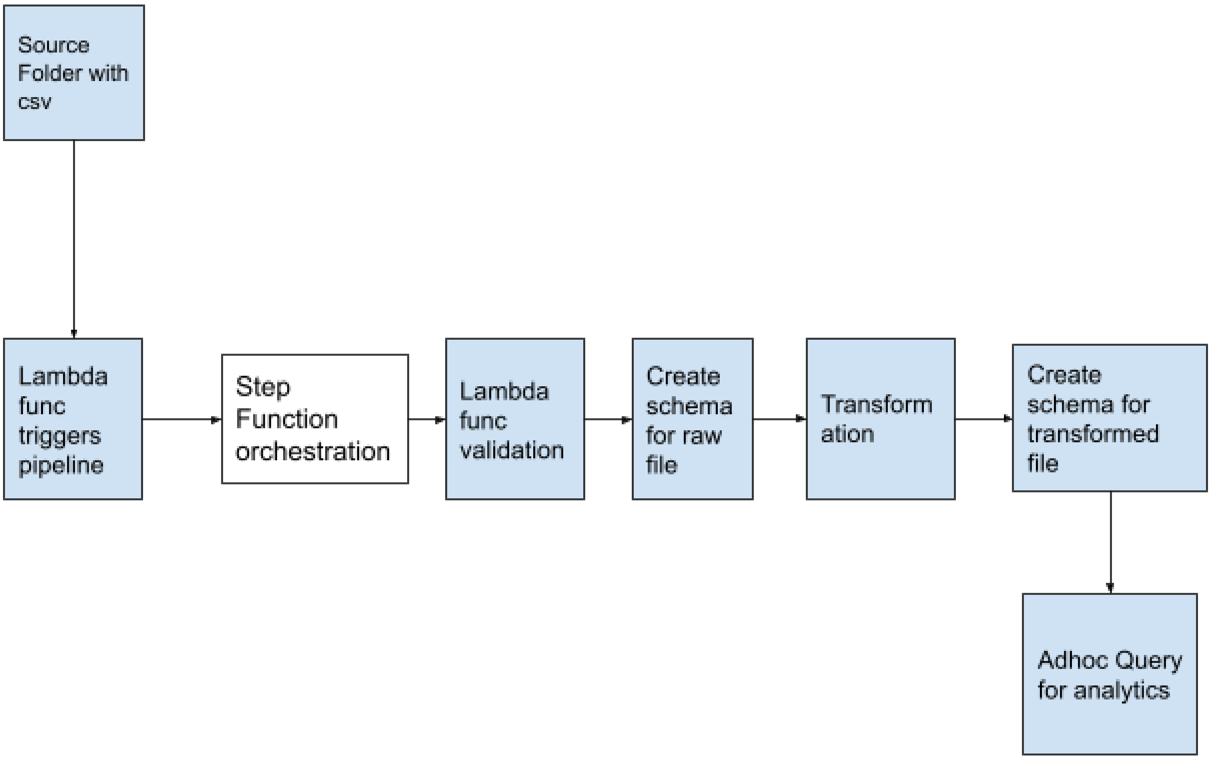
  Data Warehouse stores processed data that is used for analytics by business users. Organisations need both depending on their business needs.

  Cloud Data warehouses are a cost effective solution when companies have to manage their costs owing to their growing business needs and dont have the capacity to build on their premise. It is cost-effective and scalable. A key benefit of having an on-premises data warehouse is that if there are any problems with internet connectivity, there are likely no interruptions in your access to the data.

- Explain serverless architecture. What are its pros and cons?
  Serverless architecture makes use of a third party to provision and manage and scale servers without having to worry about where and how the code runs. They can be thought of as running in stateless compute containers that are event driven. AWS Lambda is a serverless service architecture.
  The benefit of using this approach is that
  1. it is highly available and scalable given scenarios of heavy traffic the application is not overwhelmed by the surge as it scales automatically according to growing demand,
  2. costs less as one pays only per use and the overhead cost associated with running servers 24/7 or idle time is cut off
  3. and as one is solely focused on writing code the time taken to put the application out there in the market is reduced.

  Although there maybe cases when serverless architecture maybe not that useful
  1. As they are a pay per time usage model for longer running applications it might not be that cost effective. There is when using a traditional server might be beneficial.
  2. There is some security risk involved
  3. Changing cloud vendors in later stages is a hassle and a time intensive task. The application architecture and design may need to be changed to accommodate new user

- Please provide a diagram for an ETL pipeline (ex: Section 2) using serverless AWS services. Describe each component and its function within the pipeline.


Source Folder with csv
Lambda func triggers pipeline
Step Function orchestration
Lambda func validation
Create schema for raw file
Transform ation
Create schema for transformed file
Adhoc Query for analytics

- Describe modern MLOps and how organisations should be approaching management from a tool and system perspective.

  I think of Machine learning systems as an approach that solves business problems and MLOPs as a return of investment of those ML Systems. It is an engineering discipline that aims to unify ML systems development(dev) and ML systems deployment(ops) to standardise and streamline the continuous delivery of high-performing models in production. MLOps can be thought of as an approach bridging the gap between data scientists and operations teams to ensure that models are reliable and can be easily deployed. The approach followed by MLOps is borrowed from the guidelines developed many years ago for software DevOps. Practising MLOps involves automation and monitoring at all steps of ML system construction, including integration, testing, releasing, deployment and infrastructure management. Applying MLOps to machine learning ensures faster experimentation and development of models, faster deployment of models into production and quality assurance and end-to-end lifecycle management. MLOPs ensure dealing with challenges ranging from internal teams that are struggling to bring ML models into production, extended timelines, lower than expected ROI, etc.

  For an organisation to get started with MLOPs The key aspect to realise is the breadth of systems and tools (i.e., applications, infrastructure, automation tools) process (i.e., development, testing, deployment, monitoring, automation, deployment and management) and people (i.e., data science, application development, data engineering, infrastructure and deployment).

For MLOps to be successful, all of the above facets must work in harmony, but one of the biggest challenges is being able to get all the teams together to establish and agree to standardised tooling and processes.

Tools available:

1. Kubeflow: Provides a suite of tools for running machine learning workflows on Kubernetes clusters. Provides the best open source solutions to run kubernetes clusters in a scalable, flexible and portable way. Kubeflow started as an open source implementation of TFX(TensorFlow Extended).

2. MLFlow: An opensource tool great to get things started. Industrialises end-to-end development process of Machine learning projects.Facilitates models' monitoring,reproduction,management and deployment.

3. Data Version Control: It is a python package that makes managing data science projects easier. It is an extension of Git.

4. Pachyderm: Similar to Data Version Control Pachyderm is also a version control tool. Docker and kubernetes thus can deploy machine learning projects on any cloud platform. Pachyderm ensures all data ingested into a ML model is versioned and traceable.