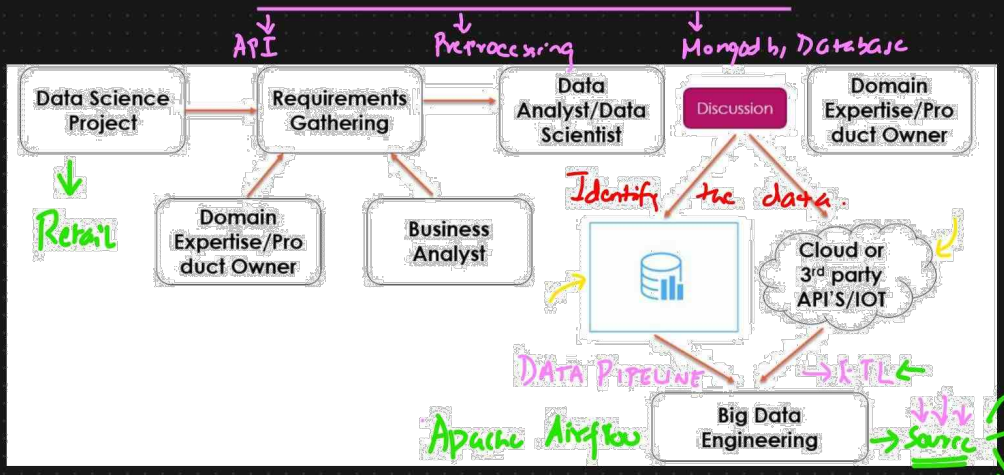
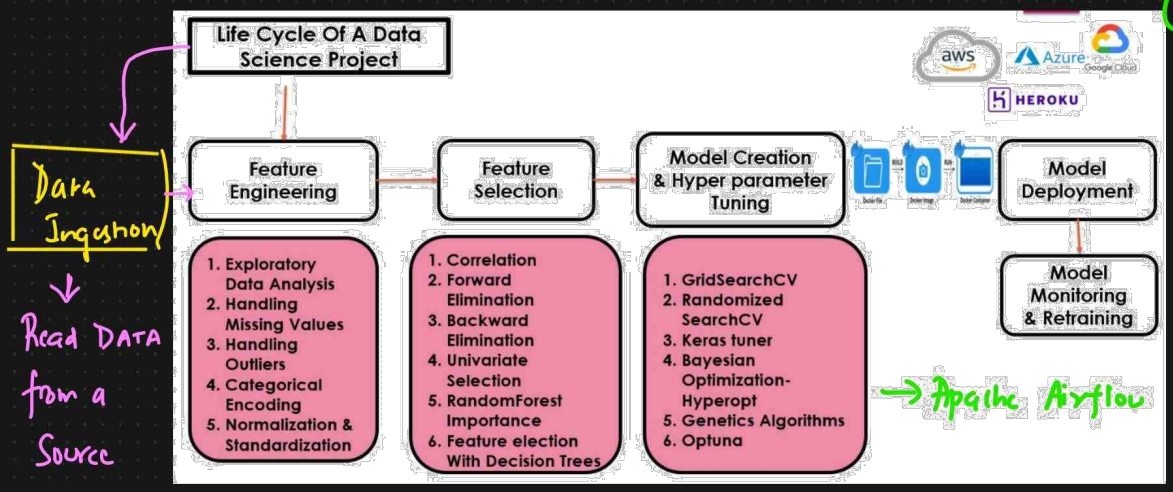
# Apache Airflow

**Apache Airflow is an open-source platform used to programmatically author, schedule, and monitor workflows. It allows you to define complex workflows as code and manage their execution. Airflow is commonly used for data pipelines, where tasks like data extraction, transformation, and loading (ETL) are orchestrated across multiple systems.**




## Key Concepts In Apache Airflow

1) DAG (Directed Acyclic GRAPH) : Collection of tasks that you want to Schedule and run.

↓

Directed Graph

① Directed → Task must have a specific seq

② Acyclic

↳ No task should be depend on itself.

A<<B<<C

A>>B>>C


2) Tasks : Task represents the individual unit of work in a DAG.

1) python Function

2) Querying a Database

3) Sending an HTTP Request.

3) Dependencies :

**Tasks in a DAG have dependencies, meaning one task might need to finish before another task can start. These dependencies allow you to control the order in which tasks are executed. Airflow provides mechanisms like set_upstream and set_downstream to define these dependencies between tasks.**

## ③ Why Airflow For MLOPS

**In MLOps (Machine Learning Operations), orchestrating ML workflows efficiently is crucial for ensuring that data pipelines, model training, and deployment tasks happen smoothly and in an automated manner. Airflow is well-suited for this purpose because it allows you to define, automate, and monitor every step in an ML pipeline.**

### 1) Orchestrating ML Pipelines And ETh Pipelines

DAG → TASKS, Dependencies


### 2) TASK Automation

### 3) Monitoring And Alerts

1) Real Time Monitoring → Airflow UI

2) Tasks logs

3) Alerts And Notifications → Emails

4) Retry Mechanism → Retry Task → Pre defined Rules.