→ Airflow is an orchestration tool that helps us in managing & scheduling jobs.

Job/Task — Combination of task is a Job.

Architecture of Airflow

DAG is the directory where the python files having the job code will be stored.

Types of Executor :-

① Sequential Executor

→ By default executor is airflow.

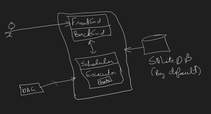
→ Seq executor runs within the scheduler itself (executor & scheduler will share the same machine).

→ It can run only one task at a time.
(no worker is present)

→ SQLite is the by default metadata database that is used. (In-memory DB that utilizes SQL Engine to store data).

→ Pros :-
- No setup is required. (By default exe)
- Light weight
- Cheap
- Less Complexity

→ Cons :-
- Not production ready
- Not scalable
- Slow (no workers)
- Single point of failure



Local Executor :-

→ The executor runs within the scheduler itself.

→ We can include multiple tasks within the executor.

Pros :-
→ Multiple task within the executor (Faster than Local one)

→ Easy to setup

→ Cheap (business cost maintenance cost)

→ Light weighted (not utilizing/using much of resource)

→ Cons :-
- Not available for production
- Not scalable
- Single point of failure (not fault tolerant)
- Cannot handle high load.

Celery Executor :-

→ This executor is most widely used in the industry

→ This executor runs on a dedicated machine. (different machine than scheduler).

→ It can handle multiple jobs/tasks at the same time.

→ We will utilize the queuing source for the task executor.

→ Pros :-
- Production Ready
- Easily Scale (horizontal scaling easy)
- Fault tolerant (multiple connections with the DB & metadata DB)
- Can work well with high load.
- Concurrent execution is possible (fast).

→ Cons :-
- High maintenance
- More complex than sequential and local exe
- Queuing system is the bottle neck
- High wasting of resources (dealing with both HW)

Kubernetes Executor :-

→ This executor is the latest.

→ The workers run on a dedicated pod.

→ Uses Kubernetes API to manage the pods.

Frontend
Backend
Scheduler
Kubernetes
Executor
MySQL
DAG
Worker
Worker
Worker
Worker