

**Comprehensive Apache Airflow**

**Course Number:** PYTH-224
**Duration:** 5 days

**Overview**

This Comprehensive Airflow training course teaches software engineers and data engineers the fundamental and advanced Airflow skills they need to successfully orchestrate production-ready data pipelines. Students learn how to create sophisticated DAGs (Directed Acyclic Graphs) and apply security practices to Apache Airflow. In addition, students learn how to scale Airflow within Kubernetes.

**Prerequisites**

All attendees should have basic Python knowledge or object-oriented programming experience.

**Materials**

All Airflow training students receive comprehensive courseware.

**Software Needed on Each Student PC**

* Python 3.5 or later
* Airflow 2.1 or later

**Objectives**

* Create production-ready data pipelines in Airflow
* Build pipelines in Airflow that are able to scale to hundreds of tasks
* Enforce modularization and reusability of Airflow tasks across projects
* Scale Airflow in Kubernetes
* Secure your Apache Airflow installation
* Create highly concurrent DAGs in Kubernetes
* Leverage most of the new functionality Airflow 2.x brings

**Outline**

* Introducing Apache Airflow
	+ What Airflow is and what does it solve?
	+ Airflow architecture
	+ How do we represent a Pipeline?
	+ Our first DAG
	+ Tasks, TaskFlow, and Operators
	+ First Pipeline
* Mastering scheduling
	+ execution\_date, start\_date and schedule\_interval
	+ Handling non-default schedule\_intervals
	+ Playing with time
* Abstracting functionality
	+ Using custom operators
	+ Creating TaskGroups vs subDAGs
	+ Sharing data with xCOMs
	+ Branching and Triggers
	+ Sensors and SmartSensors
* Executors and Scaling Airflow
	+ Abandoning SQLite for PostgreSQL
	+ Executors: Debug, Local, Celery
	+ Concurrency and parallelism
	+ Concurrency with Celery
	+ Airflow in Kubernetes, the old and new ways
	+ KEDA and HA scheduler
	+ Deploying a highly availability fault-tolerant Airflow
* Creating DAGs
	+ Secrets, connections, and variables
	+ Creating connections on startup
	+ Using Pools for long-running and demanding tasks
	+ Simulating long-running tasks
	+ DAG serialization
	+ DAG versioning
	+ Testing DAGs
	+ CI/CD in Airflow
* Modularizing DAGs
	+ TaskGroups vs subDAGs
	+ TaskFlowAPI and XComs
	+ Modularizing
	+ Dynamic and Functional DAGs
	+ SmartSensors and timeouts
* Airflow Security
	+ RBAC in Airflow
	+ Setting up OAuth authentication
	+ Add Google OAuth
	+ Adding SSL certs
	+ Default Roles and custom roles
	+ Creating a custom role
* Airflow in Kubernetes
	+ The Helm chart
	+ Deploying Airflow with Helm
	+ Deploying single tasks to Kubernetes: KubernetesPodOperator
	+ Adding a task in Kubernetes
	+ Scaling Airflow with Kubernetes executor
	+ Changing the Helm charts values
	+ KEDA autoscaler
	+ Preparing DAGs for Kubernetes
	+ Creating a DAG fully in Kubernetes
	+ The CeleryKubernetes executor for extreme scalability
* Upgrading from Airflow 1.10
* Conclusion