Programme Data Engineering

|  |  |  |  |
| --- | --- | --- | --- |
| Semaine | Technologie | Objectif | Ajouté / Modifié |
| 1-2 | Docker, Kubernetes, CI/CD | Conteneurisation, déploiement automatisé | ✅ GitHub Actions, Terraform |
| 3-4 | Apache Airflow, Airbyte, dbt | Orchestration et ETL/ELT moderne | ✅ Pipeline complet Airbyte → Airflow → dbt |
| 7-8 | Pandas + SQL, Data Quality, BigQuery | Manipulation et validation des données | ✅ Great Expectations, GCP Dataflow |
| 9-10 | Kafka, Pub/Sub | Streaming en temps réel | ✅ Pub/Sub pour cloud-native |
| 11-12 | Spark & PySpark | Traitement Big Data sur Cloud | ✅ Spark sur Dataproc |
| 13-14 | Spark avancé, Cloud Storage | Optimisation & stockage Cloud | ✅ Remplacement Hadoop par S3/GCS |
| 15-16 | Approfondissement dbt/Spark & Tests | Transformation avancée ou Big Data | ✅ Choix flexible |
| 17-18 | Projet final | Déploiement complet | ✅ GCP, CI/CD, Terraform |

🔥 Ce programme est centré sur des outils (tels que GCP, Airbyte, Kafka, dbt, Spark, etc.).   
 Il est conçu pour être à la fois pratique, évolutif, et directement applicable à des projets en entreprise.

Voici le \*\*programme final optimisé\*\* sur \*\*18 semaines\*\*, intégrant tes choix (\*\*GCP comme cloud provider\*\*) et les améliorations proposées (tests, CI/CD, modernisation).

---

# \*\*🚀 Programme Complet Data Engineering (18 Semaines) – Version Optimisée\*\*

\*\*✅ Approche :\*\* 100% pratique, orienté \*\*GCP\*\*, avec intégration des bonnes pratiques modernes (CI/CD, tests, Data Quality).

---

## \*\*🗓️ Semaines 1-2 : Docker & Kubernetes (+ CI/CD Basics)\*\*

📌 \*\*Objectifs :\*\* Maîtriser la conteneurisation et les déploiements automatisés.

✅ \*\*Contenu :\*\*

- Docker : Images, volumes, réseaux, Docker Compose.

- Kubernetes : Pods, Deployments, Services (avec \*\*GKE\*\* pour un cluster GCP).

- CI/CD : \*\*GitHub Actions\*\* pour builder et déployer une image Docker sur GCP.

🎯 \*\*Projet :\*\*

- Déployer une API FastAPI conteneurisée sur \*\*Google Kubernetes Engine (GKE)\*\*.

---

## \*\*🗓️ Semaines 3-4 : Apache Airflow (+ Terraform)\*\*

📌 \*\*Objectifs :\*\* Orchestrer des workflows et automatiser l’infra.

✅ \*\*Contenu :\*\*

- Airflow avec Docker (Composer GCP si possible).

- Création de DAGs (PythonOperator, BashOperator).

- \*\*Terraform\*\* pour déployer Airflow sur GCP.

🎯 \*\*Projet :\*\*

- Pipeline Airflow qui extrait des données de \*\*Google Cloud Storage (GCS)\*\* et les charge dans \*\*BigQuery\*\*.

---

## \*\*🗓️ Semaines 5-6 : Airbyte & dbt (+ Tests Data)\*\*

📌 \*\*Objectifs :\*\* ELT moderne avec tests de qualité.

✅ \*\*Contenu :\*\*

- Airbyte : Sources (GCS, APIs) → Destination (BigQuery).

- dbt : Modèles SQL, tests, docs.

- \*\*Great Expectations\*\* pour valider les données.

🎯 \*\*Projet :\*\*

- Pipeline \*\*Airbyte → BigQuery → dbt\*\* avec tests automatiques.

---

## \*\*🗓️ Semaines 7-8 : Pandas + SQL (+ Tests Python)\*\*

📌 \*\*Objectifs :\*\* Automatiser les transformations avec qualité.

✅ \*\*Contenu :\*\*

- Pandas : Nettoyage, jointures, aggregations.

- SQL avec \*\*BigQuery\*\* (Python Client).

- \*\*PyTest\*\* pour tester les scripts ETL.

🎯 \*\*Projet :\*\*

- Script Python qui lit un CSV depuis GCS, le transforme, et le charge dans BigQuery.

---

## \*\*🗓️ Semaines 9-10 : Kafka (+ Pub/Sub GCP)\*\*

📌 \*\*Objectifs :\*\* Streaming et temps réel.

✅ \*\*Contenu :\*\*

- Kafka avec Docker (ou \*\*Pub/Sub GCP\*\* pour une alternative cloud).

- Producers/Consumers en Python.

🎯 \*\*Projet :\*\*

- Pipeline Kafka/Pub/Sub qui ingère des logs en temps réel.

---

## \*\*🗓️ Semaines 11-12 : PySpark (+ Dataproc GCP)\*\*

📌 \*\*Objectifs :\*\* Traitement Big Data serverless.

✅ \*\*Contenu :\*\*

- PySpark : DataFrames, Spark SQL.

- \*\*Dataproc GCP\*\* pour exécuter des jobs Spark.

🎯 \*\*Projet :\*\*

- Traiter un dataset de 10+ Go sur Dataproc.

---

## \*\*🗓️ Semaines 13-14 : Dataflow (Apache Beam)\*\*

📌 \*\*Objectifs :\*\* Traitement serverless batch/streaming.

✅ \*\*Contenu :\*\*

- Apache Beam (Python SDK).

- Déploiement sur \*\*Dataflow GCP\*\*.

🎯 \*\*Projet :\*\*

- Pipeline Dataflow qui agrège des données en temps réel.

---

## \*\*🗓️ Semaines 15-16 : Data Observability & Monitoring\*\*

📌 \*\*Objectifs :\*\* Surveiller les pipelines et détecter les anomalies.

✅ \*\*Contenu :\*\*

- \*\*Data Quality\*\* (Great Expectations, Monte Carlo).

- \*\*Monitoring\*\* (Grafana + Cloud Logging).

🎯 \*\*Projet :\*\*

- Configurer des alertes sur un pipeline Airflow.

---

## \*\*🗓️ Semaines 17-18 : Projet Final – Pipeline GCP Complet\*\*

📌 \*\*Objectifs :\*\* Intégrer toutes les compétences dans un cas réel.

✅ \*\*Architecture :\*\*

1. \*\*Extraction\*\* : Airbyte (API → BigQuery).

2. \*\*Streaming\*\* : Pub/Sub + Dataflow.

3. \*\*Orchestration\*\* : Airflow.

4. \*\*Transformation\*\* : dbt + PySpark.

5. \*\*Monitoring\*\* : Grafana + Tests.

🎯 \*\*Déploiement :\*\*

- Tout déployer sur GCP avec \*\*Terraform\*\*.

---

# \*\*📊 Résumé du Programme Optimisé\*\*

| Semaines | Technologies | Focus GCP |

|-----------|------------------------------------------------|-----------------------------------|

| 1-2 | Docker, Kubernetes, CI/CD | GKE, GitHub Actions |

| 3-4 | Airflow, Terraform | Composer GCP |

| 5-6 | Airbyte, dbt, Great Expectations | BigQuery |

| 7-8 | Pandas, SQL, PyTest | BigQuery Client |

| 9-10 | Kafka / Pub/Sub | Pub/Sub GCP |

| 11-12 | PySpark, Dataproc | Dataproc GCP |

| 13-14 | Dataflow (Apache Beam) | Dataflow Serverless |

| 15-16 | Data Observability (Monitoring, Tests) | Grafana, Cloud Logging |

| 17-18 | \*\*Projet Final – Pipeline GCP End-to-End\*\* | Terraform, Multi-Services |

---