

# Data Engineering Resources

[Kubernetes Hands-On - Deploy Microservices to the AWS Cloud](#)

[Learn Apache Kafka® & Flink®](#)

[Building Apache Flink Applications in Java by Confluent](#)

[Youtube Playlist](#)

[Building Apache Flink Applications in Java - Cheatsheet](#)

[Streaming at Scale | Streaming Data Governance by Confluent \(Youtube Playlist\)](#)

[Airscholar Github - Data Engineering repos](#)

[Saving 95% Cloud Cost with Data Lakehouse - End to End Data Engineering Project - PART 01](#)

[Reddit Data Pipeline Engineering | AWS End to End Data Engineering](#)

[Example of Real Time Using Flink](#)

[Example of Batch Using Flink](#)

[Datalake Rock Paper Scissors: Iceberg + Flink or Iceberg + Spark? | Current 2023 \(by Confluent\)](#)

[In Case If Someone Needs Justifications Why Tight Coupling is Bad](#)

[Learn By Example : Apache Flink](#)

[Flink - Iceberg Unit Tests](#)

[Example of Batch ETL Using Flink](#)

[Example of Stream Processing Using Flink](#)

[Basic Flink Operations Examples](#)

[Apache Flink with Scala 3](#)

[Apache Flink 101 Developer Course Formula- PyFlink Hands On](#)

[Apache Iceberg](#)

## Kubernetes Hands-On - Deploy Microservices to the AWS Cloud

<https://www.udemy.com/course/kubernetes-microservices/>

## Learn Apache Kafka® & Flink®

<https://developer.confluent.io/courses/>

# Building Apache Flink Applications in Java by Confluent

## Youtube Playlist

<https://www.youtube.com/playlist?list=PLa7VYi0yPIH0QEIcyvZE5p4zMR0In4aAe>

## Building Apache Flink Applications in Java - Cheatsheet

<https://github.com/confluentinc/learn-building-flink-applications-in-java-exercises/blob/main/cheatsheet.md>

## Streaming at Scale | Streaming Data Governance by Confluent (Youtube Playlist)

[https://www.youtube.com/watch?v=yLkKDY3sGec&list=PLa7VYi0yPIH1sKNegs6Y8m92PRoYrcQzm&index=2&ab\\_channel=Confluent](https://www.youtube.com/watch?v=yLkKDY3sGec&list=PLa7VYi0yPIH1sKNegs6Y8m92PRoYrcQzm&index=2&ab_channel=Confluent)

## Airscholar Github - Data Engineering repos

<https://github.com/airscholar?tab=repositories>

## Saving 95% Cloud Cost with Data Lakehouse - End to End Data Engineering Project - PART 01

[https://www.youtube.com/watch?v=p36YixNqGLg&ab\\_channel=CodeWithYu](https://www.youtube.com/watch?v=p36YixNqGLg&ab_channel=CodeWithYu)

## Reddit Data Pipeline Engineering | AWS End to End Data Engineering

[https://www.youtube.com/watch?v=LSlt6iVI\\_9Y&ab\\_channel=CodeWithYu](https://www.youtube.com/watch?v=LSlt6iVI_9Y&ab_channel=CodeWithYu)

## Example of Real Time Using Flink

<https://github.com/airscholar/FlinkCommerce>

## Example of Batch Using Flink

<https://github.com/airscholar/ApacheFlink-SalesAnalytics>

## Datalake Rock Paper Scissors: Iceberg + Flink or Iceberg + Spark? | Current 2023 (by Confluent)

[https://www.youtube.com/watch?v=lfeNEQ3MtvM&ab\\_channel=Confluent](https://www.youtube.com/watch?v=lfeNEQ3MtvM&ab_channel=Confluent)

Bloomberg uses Apache Kafka® and Apache Iceberg® as core elements in their real-time data pipelines and storage sinks. In this talk, Sitarama Chekuri and Ben de Vera share their lessons learned testing both Apache Flink® and Apache Spark® to ingest data from Kafka into their Iceberg data lake at near-real-time speeds. They compare and contrast the two technologies with regard to functionality, performance, fault-tolerance, scaling, and resource utilization.

## In Case If Someone Needs Justifications Why Tight Coupling is Bad

[https://www.youtube.com/watch?v=7fkS-18KBlw&ab\\_channel=Confluent](https://www.youtube.com/watch?v=7fkS-18KBlw&ab_channel=Confluent)

[https://www.youtube.com/watch?v=J-kKR3omk-g&ab\\_channel=Confluent](https://www.youtube.com/watch?v=J-kKR3omk-g&ab_channel=Confluent)

Also contains basics of event sourcing.

On resiliency: [https://www.youtube.com/watch?v=YEsP9zW1h10&ab\\_channel=Confluent](https://www.youtube.com/watch?v=YEsP9zW1h10&ab_channel=Confluent)

## Learn By Example : Apache Flink

<https://www.udemy.com/course/learn-by-example-apache-flink/>

Flink is a stream processing technology with added capability to do lots of other things like batch processing, graph algorithms, machine learning etc. Using Flink you can build applications which need you to be highly responsive to the latest data such as monitoring spikes in payment gateway failures or triggering trades based on live stock price movements.

This course has 30 Solved Examples on building Flink Applications for both Streaming and Batch Processing.

*RZ: pretty basic course bordering on useless. Just an overview of streaming implemented in Flink. If you never worked with streaming it may provide some value.*

## Flink - Iceberg Unit Tests

<https://github.com/apache/iceberg/blob/main/flink/v1.19/flink/src/test/java/org/apache/iceberg/flink/sink/TestFlinkIcebergSink.java>

This is the latest v1.19 version as of June 3, 2024. Check for the latest version if the url above doesn't work. Looks like they tend to remove or reorganize older versions.

## Example of Batch ETL Using Flink

<https://github.com/airscholar/ApacheFlink-SalesAnalytics/blob/main/src/main/java/salesAnalysis/DataBatchJob.java>

## Example of Stream Processing Using Flink

[https://github.com/developer-advocacy-dremio/flink-iceberg-nessie-environment/blob/main/flink-job/iceberg-flink-example/src/main/java/com/dremio\\_developer\\_advocacy/HadoopIceberg.java](https://github.com/developer-advocacy-dremio/flink-iceberg-nessie-environment/blob/main/flink-job/iceberg-flink-example/src/main/java/com/dremio_developer_advocacy/HadoopIceberg.java)

## Basic Flink Operations Examples

[https://github.com/apache/flink/blob/master/flink-python/pyflink/examples/table/basic\\_operations.py](https://github.com/apache/flink/blob/master/flink-python/pyflink/examples/table/basic_operations.py)

## Apache Flink with Scala 3

<https://www.udemy.com/course/apache-flink-with-scala-3/>

Succinct Flink course using Scala. Has a section about Flink app deployment.

## Apache Flink 101 Developer Course Formula-PyFlink Hands On

<https://www.udemy.com/course/flink-mastery/>

## Apache Iceberg

YouTube playlist by Dremio.

[https://www.youtube.com/playlist?list=PL-glUf9e9CCtGr\\_zYdWieJhiqBG\\_5qSPa](https://www.youtube.com/playlist?list=PL-glUf9e9CCtGr_zYdWieJhiqBG_5qSPa)