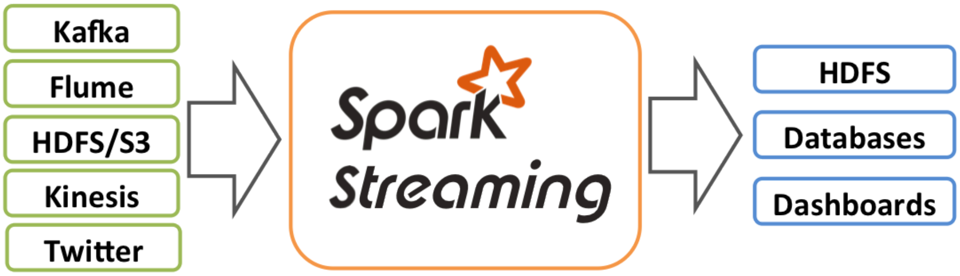
Practical Assignment No. 5	
Title:	**Apache Spark**
Problem Statement:	Demonstrate application of Apache spark to analyse streaming data from social media.
Objective:	To understand Apache Spark machine learning techniques for developing big data processing applications.
Outcome:	CO606.4: Utilize Apache Spark machine learning techniques for developing big data processing applications.
Software or Hardware Requirements:	Anaconda/Java/GCC
Theory:	**Apache Spark** In the era of Big Data, a significant portion of information is generated in real time from social media platforms such as Twitter, Facebook, Instagram, and YouTube. Analyzing this streaming data helps organizations understand trends, sentiments, and user behavior. **Apache Spark** provides a powerful framework for processing and analyzing such continuous streams of data in a scalable and fault-tolerant manner. **Apache Spark** is an open-source, distributed computing system that provides an in-memory data processing capability, making it faster than traditional batch-processing frameworks like Hadoop MapReduce. For real-time analysis, Spark offers a component called **Spark Streaming**, and in newer versions, **Structured Streaming**, which enables developers to process live data streams.  Kafka Flume HDFS/S3 Kinesis Twitter Spark Streaming HDFS Databases Dashboards

Working Principle:

1. **Data Ingestion:**
   Streaming data is collected from social media platforms using APIs (e.g., Twitter API) or message queues like **Apache Kafka**, **Flume**, or **Socket connections**. These tools push continuous streams of data into Spark.

2. **Stream Processing:**
   Spark Streaming divides the incoming data stream into small **micro-batches**. Each batch is processed using Spark's core engine with transformations (like map, reduce, filter) and actions.
   In **Structured Streaming**, data is treated as an unbounded table that continuously appends new records.

3. **Data Analysis:**
   The real-time data is analyzed to extract useful insights such as:

   - Trending topics or hashtags

   - Sentiment analysis (positive, negative, neutral tweets)

   - User activity analysis or engagement metrics

4. **Output and Visualization:**
   The analyzed results can be stored in databases like **MongoDB**, **HDFS**, or displayed through dashboards using **Grafana**, **Tableau**, or **Spark's web UI**.

Advantages of Using Apache Spark for Streaming Data:

- **Real-time processing:** Handles continuous data streams efficiently.

- **Scalability:** Can process data across multiple nodes in a cluster.

- **Fault tolerance:** Uses RDD lineage to recover data in case of failure.

- **Integration:** Works seamlessly with Kafka, Flume, HDFS, and cloud services.

- **Unified engine:** Can handle batch, streaming, machine learning, and graph processing in one framework.

Applications:

	● Monitoring **social media sentiment** for brand reputation. ● Detecting **trending topics or hashtags** in real-time. ● Performing **real-time recommendation** or advertisement targeting. ● Tracking **public opinion** or crisis communication through live feeds.
Input/Datasets/Test Cases:	(Social Media Website details)
Results:	Execute code for analysing and streaming data from social media. Take a print of this code with output for submission as part of results.
Analysis and conclusion:	Write your own analysis of output and conclusion(Minimum 1 statement of Analysis, Minimum 1 Statement Conclusion)
References:	Reference /Links(min Any 2, include dataset ref.). Write references in IEEE format.