DAY-6(Apache Spark)

1. Working with RDDs:
   a) Write a Python program to create an RDD from a local data source.
   b) Implement transformations and actions on the RDD to perform data processing tasks.
   c) Analyze and manipulate data using RDD operations such as map, filter, reduce, or aggregate.

2. Spark DataFrame Operations:
   a) Write a Python program to load a CSV file into a Spark DataFrame.
   b)Perform common DataFrame operations such as filtering, grouping, or joining.
   c) Apply Spark SQL queries on the DataFrame to extract insights from the data.

3. Spark Streaming:
   a) Write a Python program to create a Spark Streaming application.
   b) Configure the application to consume data from a streaming source (e.g., Kafka or a socket).
   c) Implement streaming transformations and actions to process and analyze the incoming data stream.

4. Spark SQL and Data Source Integration:
   a) Write a Python program to connect Spark with a relational database (e.g., MySQL, PostgreSQL).
   b)Perform SQL operations on the data stored in the database using Spark SQL.
   c) Explore the integration capabilities of Spark with other data sources, such as Hadoop Distributed File System (HDFS) or Amazon S3.

Submission Guidelines:

- Answer all the questions in a single Jupyter Notebook file (.ipynb).
- Include necessary code, comments, and explanations to support your answers and implementation.
- Ensure the notebook runs without errors and is well-organized.
- Create a GitHub repository to host your assignment files.
- Rename the Jupyter Notebook file using the format "date_month_topic.ipynb" (e.g., "12_July_Spark.ipynb").
- Place the Jupyter Notebook file in the repository.
- Commit and push any additional files or resources required to run your code (if applicable) to the repository.
- Ensure the repository is publicly accessible.
- Submit the link to your GitHub repository as the assignment submission.

Grading Criteria:

1. Understanding and completeness of answers: 40%
2. Clarity and depth of explanations: 25%

3. Correct implementation and evaluation of optimizer techniques: 15%
4. Analysis and comparison of different optimizers: 10%
5. Proper code implementation and organization: 10%

Note:- Create your assignment in Jupyter notebook and upload it to GitHub & share that uploaded assignment file link through your dashboard. Make sure the repository is public.