Prezado candidato.

Gostaríamos de fazer um teste que será usado para sabermos a sua proficiência nas habilidades para a vaga. O teste

consiste em algumas perguntas e exercícios práticos sobre Spark e as respostas e códigos implementados devem ser

armazenados no GitHub. O link do seu repositório deve ser compartilhado conosco ao final do teste.

Quando usar alguma referência ou biblioteca externa, informe no arquivo README do seu projeto. Se tiver alguma dúvida, use o bom senso e se precisar deixe isso registrado na documentação do projeto.

1. Qual o objetivo do comando **cache** em Spark?

O objetivo do comando Cache, consiste em criar a criar a persistência do RDD, com esse comando vc acessa as informações já com as transformações que foram realizadas.

2- O mesmo código implementado em Spark é normalmente mais rápido que a implementação equivalente em MapReduce. Por quê?

No spark os dados são trabalhados diretamente na memória assim evitando um consumo de recurso maior com operações de entrada e saída no acesso aos dados.

3- Qual é a função do **SparkContext** ?

Criar a conexão do cluster com o programa criado.

4-Explique com suas palavras o que é **Resilient Distributed Datasets** (RDD).

São dataSets que podem ser criados pelo programa ou vir através de uma fonte de dados externa por exemplo, ao qual [e armazenado em memória e distribuído entre o cluster para processamento.

5-**GroupByKey** é menos eficiente que **reduceByKey** em grandes dataset. Por quê?

No **GroupByKey** os dadossão pre-trabalhados e agrupados antes de serem enviados para a partição, em grandes datasets fazer isso antes do reduceByKey pode causar processamento excessivo assim comprometendo a performance

Explique o que o código Scala abaixo faz.

val textFile = sc . textFile ( "hdfs://..." )

val counts = textFile . flatMap ( line => line . split ( " " ))

. map ( word => ( word , 1 ))

. reduceByKey ( \_ + \_ )

counts . saveAsTextFile ( "hdfs://..." )

Criou o Rdd com o arquivo de texto no caminho especificado, separando por espaço, atribui o valor 1, criou o Map e o reduce e gravou o resultado como arquivo no HDFS