

Avaliação - Spark Big Data Processing

Iniciado: 2 ago em 16:04

Instruções do teste

Pergunta 1

1 pts

Marque as alternativas **verdadeiras** sobre o API Catalog.

- ☐ Pertence a classe SparkContext
- ☒ Pertence a classe SparkSession
- ☐ Gerenciar todos os arquivos no HDFS
- ☒ Gerenciar as tabelas do Metastore

Pergunta 2

1 pts

Preencher as lacunas do comando abaixo, responsável por contar as palavras de um RDD avaliacao_log, que contém o log de uma aplicação formado por um conjunto de linhas separadas por /n (Quebra de linha ou Enter) e as linhas contém palavras separadas por espaço.

Comando:

avaliacao_log. (lambda line: line.split(" ")).

(lambda word: word.lower()).

(lambda word: (word,1)).

(lambda k1,k2: k1+k2)

Obs.: Preencher as lacunas sem espaços, " " ou ' '.

Pergunta 3**1 pts**

Marque as alternativas **verdadeiras** sobre Dataset.

- ☒ Criar com base de dados de um Dataframe
- ☒ Transformações são tipadas
- ☐ Transformações não são tipadas
- ☒ Scala pode fazer uso deste tipo de dado
- ☐ Python pode fazer uso deste tipo de dado
- ☒ Java pode fazer uso deste tipo de dado
- ☒ Criar com base de dados de um RDD

Pergunta 4**1 pts**

Preencher as lacunas do comando abaixo, responsável por criar o campo "ano" no dataframe avaliacao_cliente, onde neste campo irá converter o campo "data" que está no formato de dia/mês/ano, para apenas ano.

Comando:

```
avaliacao_cliente.withColumn("ano", from_unixtime  
  
(unix_timestamp(col("data"), "dd/MM/yyyy"), "yyyy"))
```

Obs.: Preencher as lacunas sem espaços, " " ou ' '.

Pergunta 5**1 pts**

Preencher as lacunas do comando abaixo, responsável por visualizar a média dos salários com duas casas decimais de cada setor, ordenado por ordem alfabética do dataframe `avaliacao_cliente`.

Comando:

```
avaliacao_cliente.  ("setor").agg(format_number(  
 ("salario"),2)).sort( ("setor"))
```

Obs.: Preencher as lacunas sem espaços, " " ou ' '.

Pergunta 6**1 pts**

Marque as alternativas **verdadeiras** sobre Spark Application.

- ☐ É possível fazer o deploy em modo cluster com uso do spark-shell
- ☒ É obrigatório criar uma Sessão no Spark para fazer uso do Spark SQL
- ☒ É possível fazer o deploy em modo cluster com uso do spark-submit
- ☐ Não é obrigatório criar uma Sessão no Spark para fazer uso do Spark SQL
- ☒ É possível enviar um job localmente com spark-submit

Pergunta 7**1 pts**

Marque as alternativas **verdadeiras** sobre Spark Streaming.

- ☐ Recebe Stream de dados na entrada e envia para a engine do spark em Stream
- ☒ É uma sequência de RDDs
- ☐ É uma sequência de Dataframe
- ☒ Recebe Stream de dados na entrada e envia para a engine do spark em batch
- ☐ É uma extensão da API do Spark SQL

Pergunta 8**1 pts**

Em uma aplicação Spark era necessário ler os dados de um tópico do Kafka, para isso, foi adicionado a seguinte dependência no SBT:

```
libraryDependencies += "org.apache.spark" % "spark-streaming-kafka-0-10" %  
"2.4.4"
```

Marque as alternativas **verdadeiras** que podem ser usadas nesta dependência.

- ☒ Suporte para a linguagem Scala

☒ Todas as versões do Spark maior ou igual a 2.4.4

☐ Broker do Kafka na versão 0,5

☒ Broker do Kafka na versão 0,20

☐ Suporte para a linguagem Python

Pergunta 9

1 pts

Marque as alternativas **verdadeiras** sobre Struct Streaming.

☒ Ler um dado em batch e salvar como stream

☒ Suporte para a linguagem Python

☒ Possui as mesmas transformações de DataFrame

☐ Ler um dado em stream e salvar como bath

☒ Suporte para a linguagem Scala

☐ Possui as mesmas transformações de RDD

Pergunta 10

1 pts

Marque as alternativas **verdadeiras** sobre UDF.

- ☐ São otimizadas pelo Spark Catalyst
- ☒ Não são otimizadas pelo Spark Catalyst
- ☒ Pode ser usado como função no spark.sql
- ☐ Usar sempre para substituir uma função existente no Spark SQL
- ☒ Usar apenas quando não existe uma função no Spark SQL

Salvo em 16:19

Enviar teste