✖

Spark Lesson 2

5 questions

1 point

1.
How can you create an RDD? Mark all that apply

☑ Apply a transformation to an existing RDD

☑ Reading from HDFS

□ Calling collect() on an existing RDD

☑ Reading from a local file available both on the driver and on the workers

1 point

2.
How does Spark make RDDs resilient in case a partition is lost?

○ By default keeps multiple copies in memory across different nodes

○ By default keeps multiple copies in memory on the same node

● Tracks the history of each partition and reruns what is needed to restore it

○ Tracks the history of each partition and reads it back from disk

1 point

3.
Which of the following sentences about flatMap and map are true?

□ flatMap accepts a function that returns multiple elements, those elements are then flattened out into a continuous RDD.

☑ map transforms elements with a 1 to 1 relationship, 1 input - 1 output

□ any flatMap transforms each input element in the same number of X output elements, so the size of the output RDD is X times the size of the input RDD

■ if you use flatMap with this function:

```
def my_func(a):
    return [a, a+1]

```

on a RDD that contains only the numbers 2 and 8, and collect the output RDD to the Driver, the output would be:

```
[[2, 3], [8, 9]]
```

1
point

4.
Check all wide transformations

■ groupByKey

■ reduceByKey

■ repartition

□ flatMap

□ shuffle

1
point

5.
Check all true statements about shuffle

□ groupByKey and reduceByKey have similar performance because both trigger a shuffle

■ Repartition, even if it triggers a shuffle, can improve performance of your pipeline by balancing the data distribution after a heavy filtering operation

■ A shuffle operation always works in memory

☑ I understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account. Learn more about Coursera's Honor Code

mingda zhang

Submit Quiz

👍 👎 ⚑