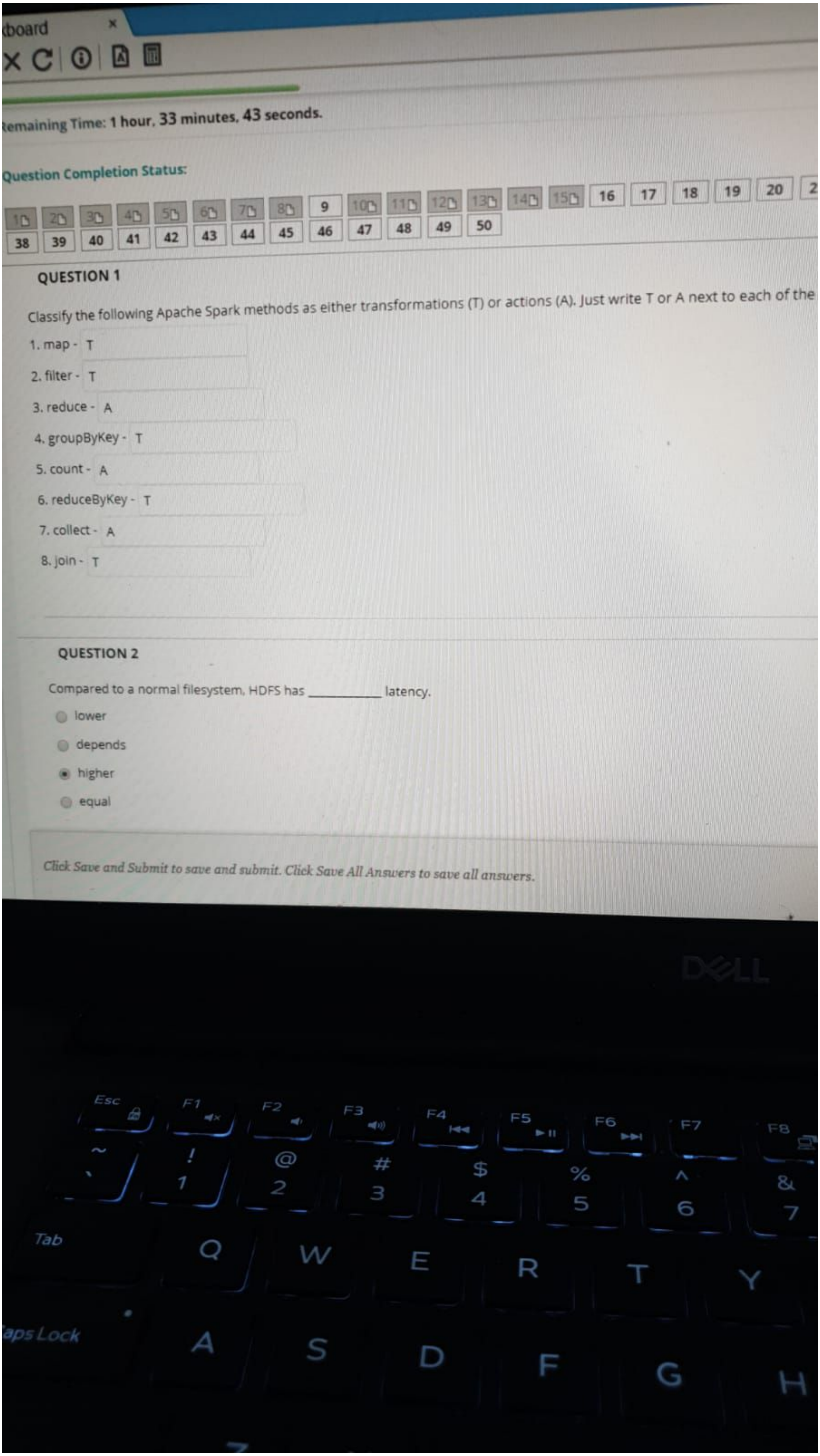


Remaining Time: **1 hour, 33 minutes, 43 seconds.**

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 2

38 39 40 41 42 43 44 45 46 47 48 49 50

QUESTION 1

Classify the following Apache Spark methods as either transformations (T) or actions (A). Just write T or A next to each of the

1. map - T
2. filter - T
3. reduce - A
4. groupByKey - T
5. count - A
6. reduceByKey - T
7. collect - A
8. join - T

QUESTION 2

Compared to a normal filesystem, HDFS has __________ latency.

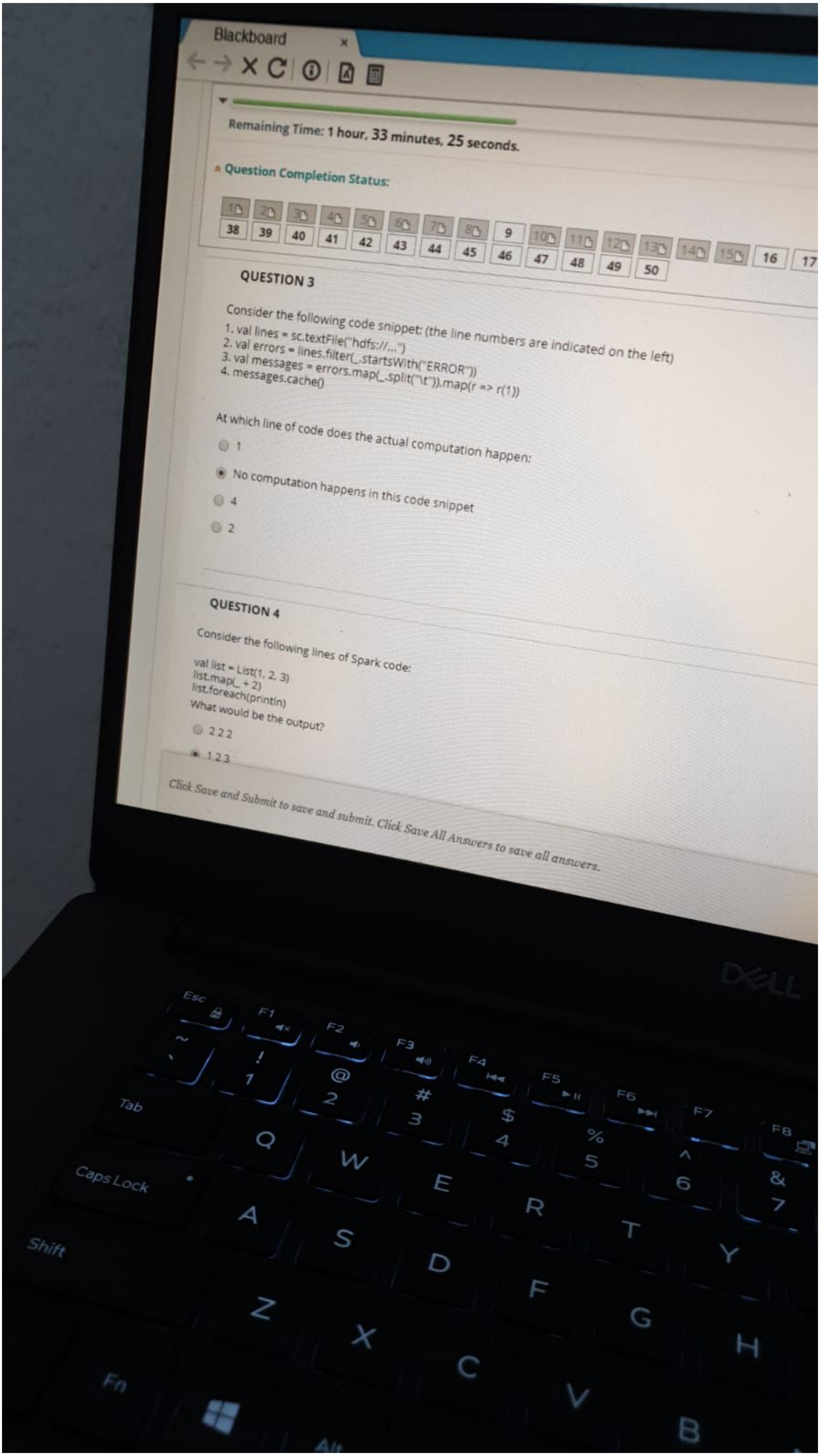
○ lower

○ depends

● higher

○ equal

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: 1 hour, 33 minutes, 25 seconds.

QUESTION 3

Consider the following code snippet: (the line numbers are indicated on the left)

```
1. val lines = sc.textFile("hdfs://...")
2. val errors = lines.filter(_.startsWith("ERROR"))
3. val messages = errors.map(_.split("\t")).map(r => r(1))
4. messages.cache()
```

At which line of code does the actual computation happen:

○ 1

● No computation happens in this code snippet

○ 4

○ 2

QUESTION 4

Consider the following lines of Spark code:

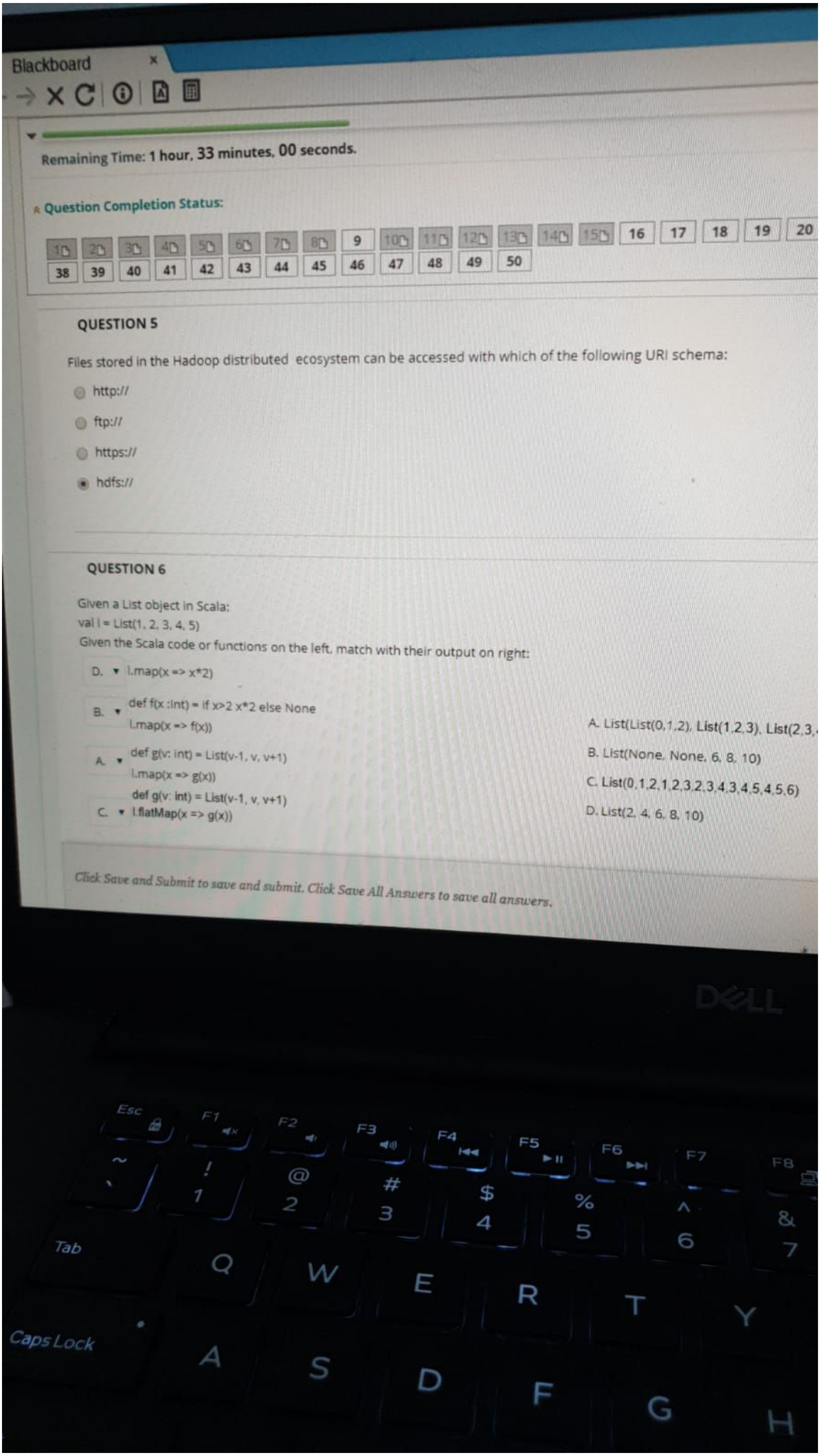
```
val list = List(1, 2, 3)
list.map(_ + 2)
list.foreach(println)
```

What would be the output?

○ 2 2 2

● 1 2 3

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: **1 hour, 33 minutes, 00 seconds.**

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

38 39 40 41 42 43 44 45 46 47 48 49 50

QUESTION 5

Files stored in the Hadoop distributed ecosystem can be accessed with which of the following URI schema:

○ http://

○ ftp://

○ https://

● hdfs://

QUESTION 6

Given a List object in Scala:
val l = List(1, 2, 3, 4, 5)
Given the Scala code or functions on the left, match with their output on right:

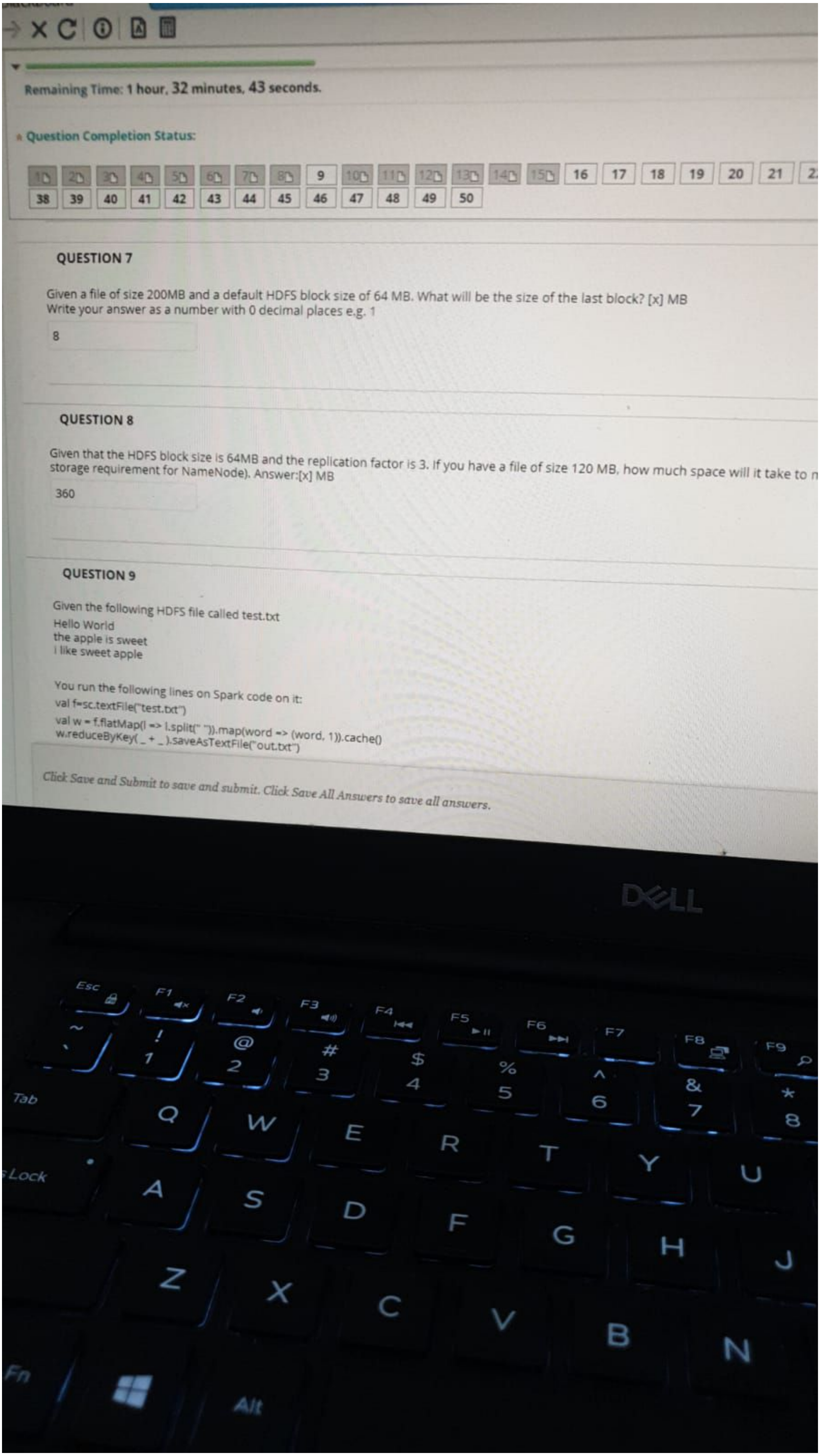
D. ▾	l.map(x => x*2)	

B. ▾	def f(x :Int) = if x>2 x*2 else None
	l.map(x => f(x))
		A. List(List(0,1,2), List(1,2,3), List(2,3,

A. ▾	def g(v: int) = List(v-1, v, v+1)
	l.map(x => g(x))
		B. List(None, None, 6, 8, 10)
		C. List(0,1,2,1,2,3,2,3,4,3,4,5,4,5,6)

C. ▾	def g(v: int) = List(v-1, v, v+1)
	l.flatMap(x => g(x))
		D. List(2, 4, 6, 8, 10)

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: 1 hour, 32 minutes, 43 seconds.

Question Completion Status:

QUESTION 7

Given a file of size 200MB and a default HDFS block size of 64 MB. What will be the size of the last block? [x] MB
Write your answer as a number with 0 decimal places e.g. 1

8

QUESTION 8

Given that the HDFS block size is 64MB and the replication factor is 3. If you have a file of size 120 MB, how much space will it take to n
storage requirement for NameNode). Answer:[x] MB

360

QUESTION 9

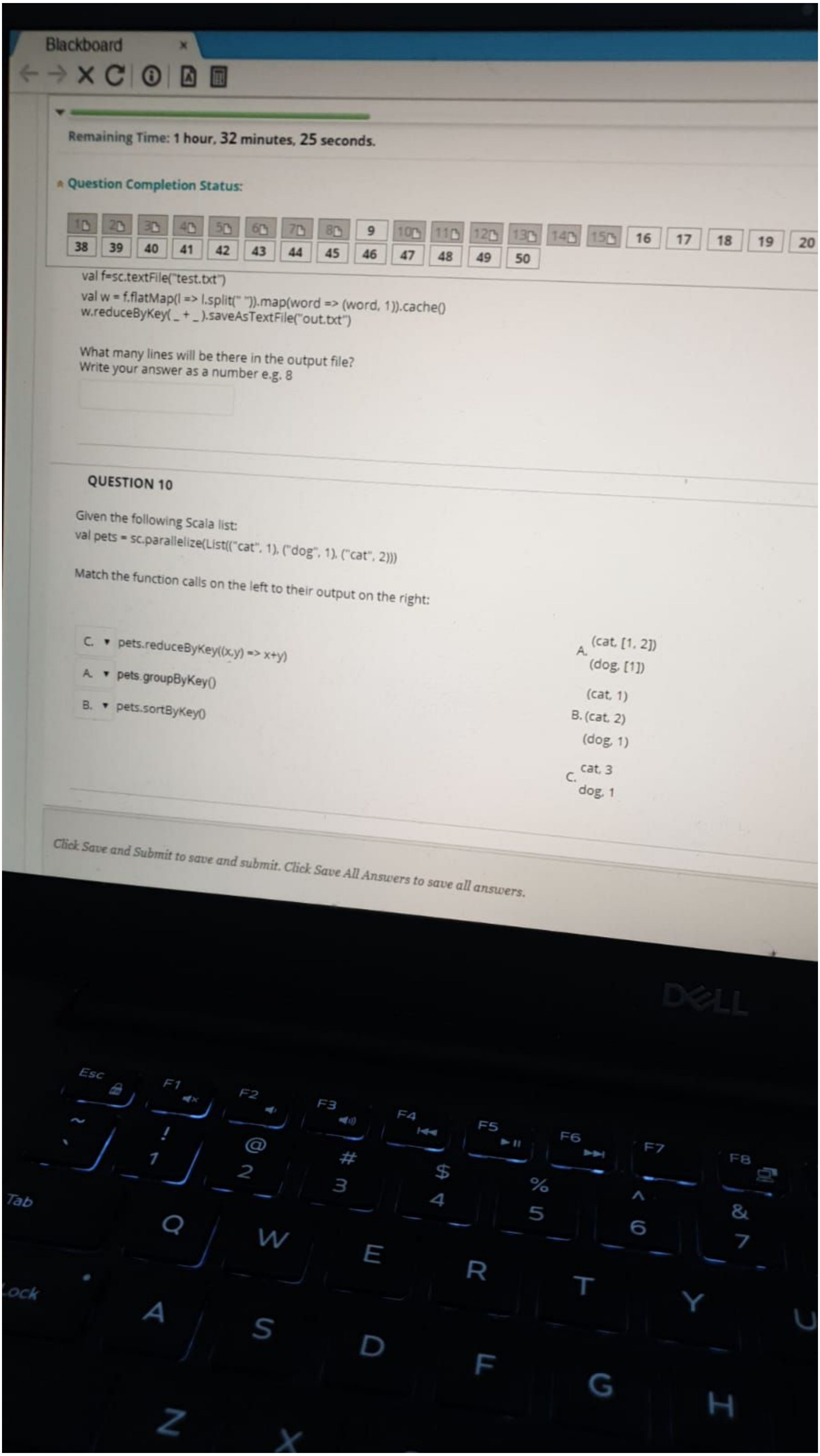
Given the following HDFS file called test.txt

```
Hello World
the apple is sweet
i like sweet apple
```

You run the following lines on Spark code on it:

```
val f=sc.textFile("test.txt")
val w = f.flatMap(l => l.split(" ")).map(word => (word, 1)).cache()
w.reduceByKey( _ + _ ).saveAsTextFile("out.txt")
```

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: 1 hour, 32 minutes, 25 seconds.

Question Completion Status:

```
val f=sc.textFile("test.txt")
val w = f.flatMap(l => l.split(" ")).map(word => (word, 1)).cache()
w.reduceByKey( _ + _ ).saveAsTextFile("out.txt")
```

What many lines will be there in the output file?
Write your answer as a number e.g. 8

QUESTION 10

Given the following Scala list:
val pets = sc.parallelize(List(("cat", 1), ("dog", 1), ("cat", 2)))

Match the function calls on the left to their output on the right:

C. ▾ pets.reduceByKey((x,y) => x+y)

A. ▾ pets.groupByKey()

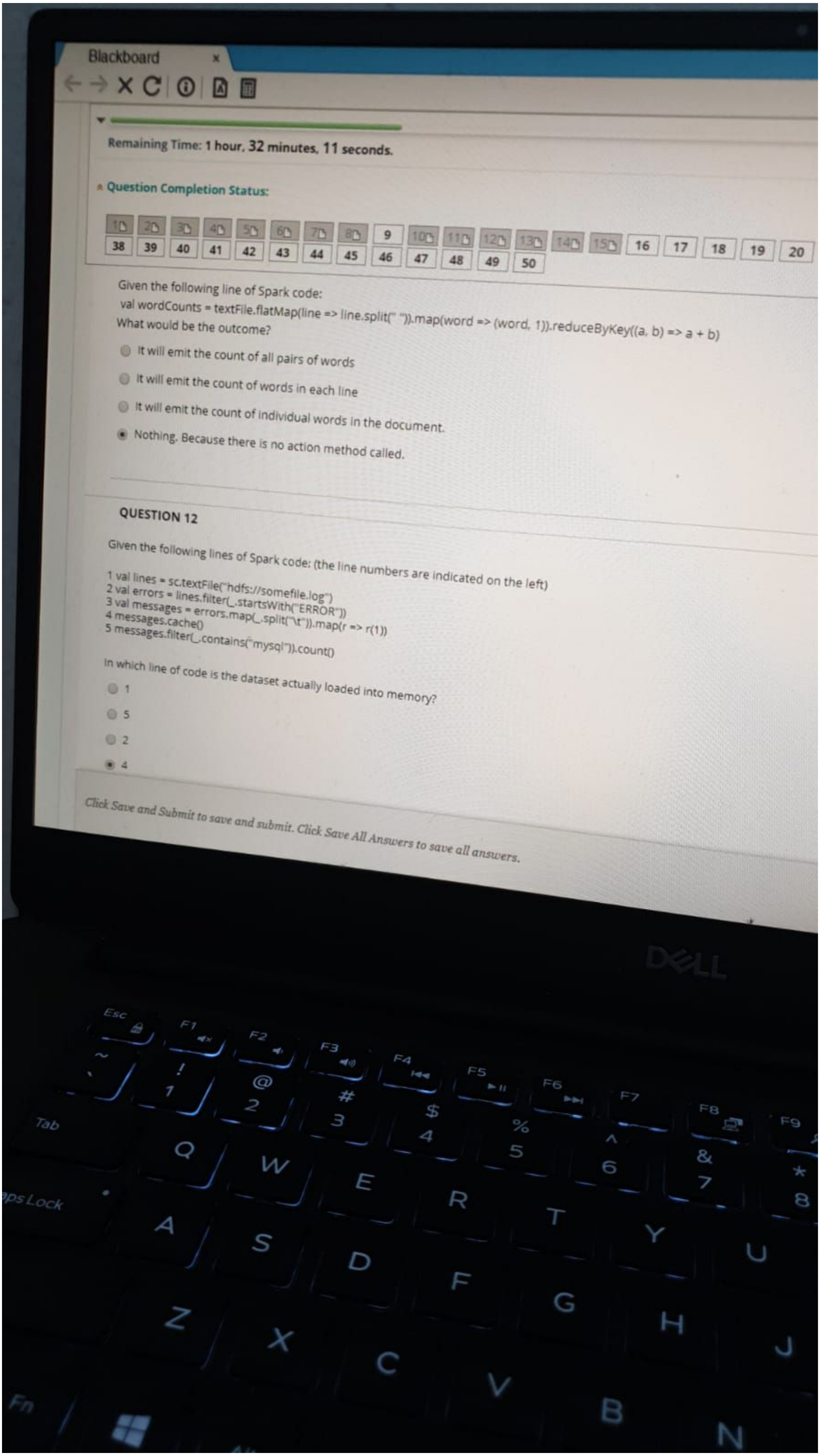
B. ▾ pets.sortByKey()

A. (cat, [1, 2])
(dog, [1])

B. (cat, 1)
(cat, 2)
(dog, 1)

C. cat, 3
dog, 1

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: **1 hour, 32 minutes, 11 seconds.**

Question Completion Status:

Given the following line of Spark code:

```
val wordCounts = textFile.flatMap(line => line.split(" ")).map(word => (word, 1)).reduceByKey((a, b) => a + b)
```

What would be the outcome?

○ It will emit the count of all pairs of words

○ It will emit the count of words in each line

○ It will emit the count of individual words in the document.

● Nothing. Because there is no action method called.

QUESTION 12

Given the following lines of Spark code: (the line numbers are indicated on the left)

```
1 val lines = sc.textFile("hdfs://somefile.log")
2 val errors = lines.filter(_.startsWith("ERROR"))
3 val messages = errors.map(_.split("\t")).map(r => r(1))
4 messages.cache()
5 messages.filter(_.contains("mysql")).count()
```

In which line of code is the dataset actually loaded into memory?

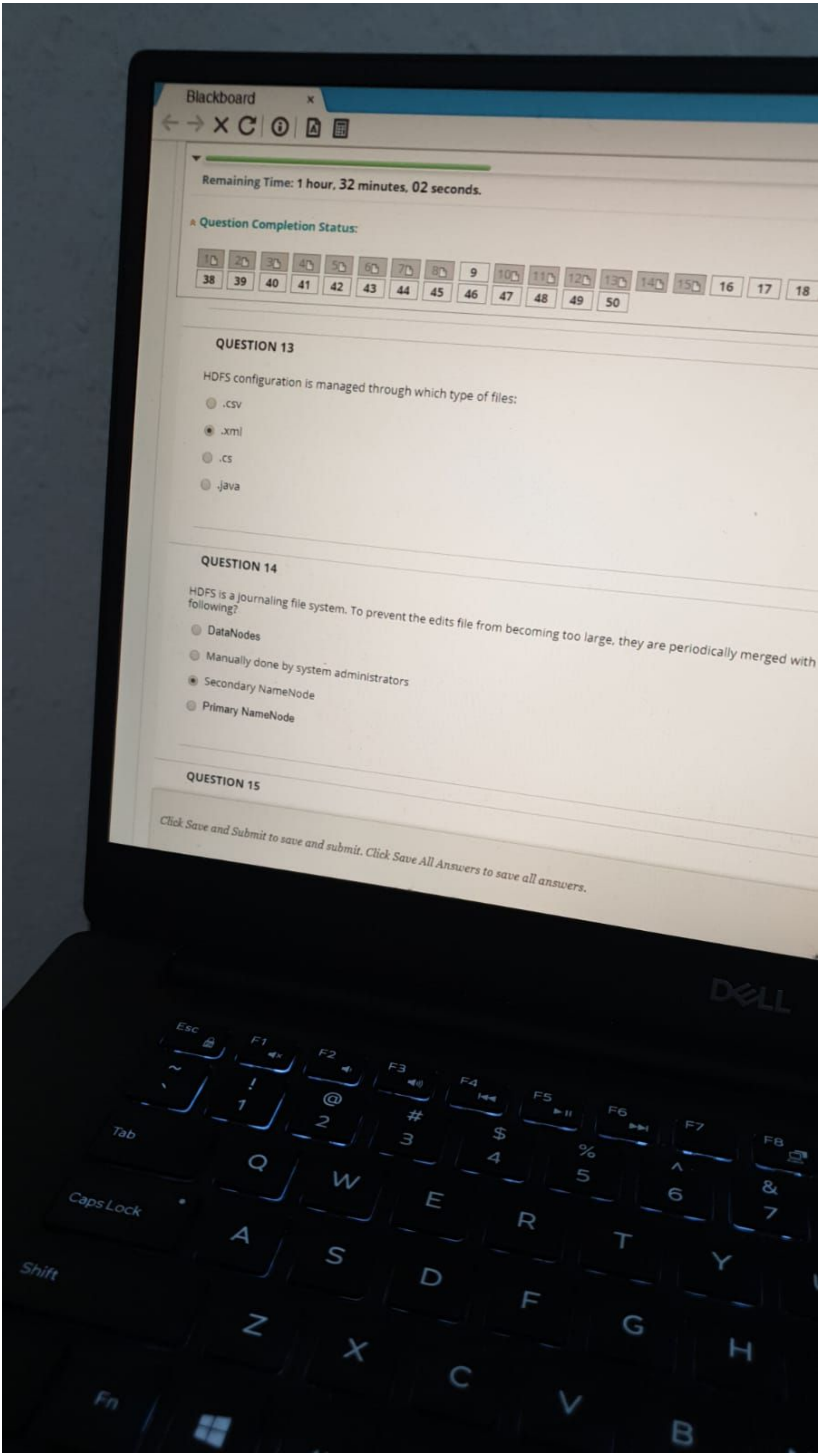
○ 1

○ 5

○ 2

● 4

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: 1 hour, 32 minutes, 02 seconds.

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
38 39 40 41 42 43 44 45 46 47 48 49 50

QUESTION 13

HDFS configuration is managed through which type of files:

○ .csv

● .xml

○ .cs

○ .java

QUESTION 14

HDFS is a journaling file system. To prevent the edits file from becoming too large, they are periodically merged with following?

○ DataNodes

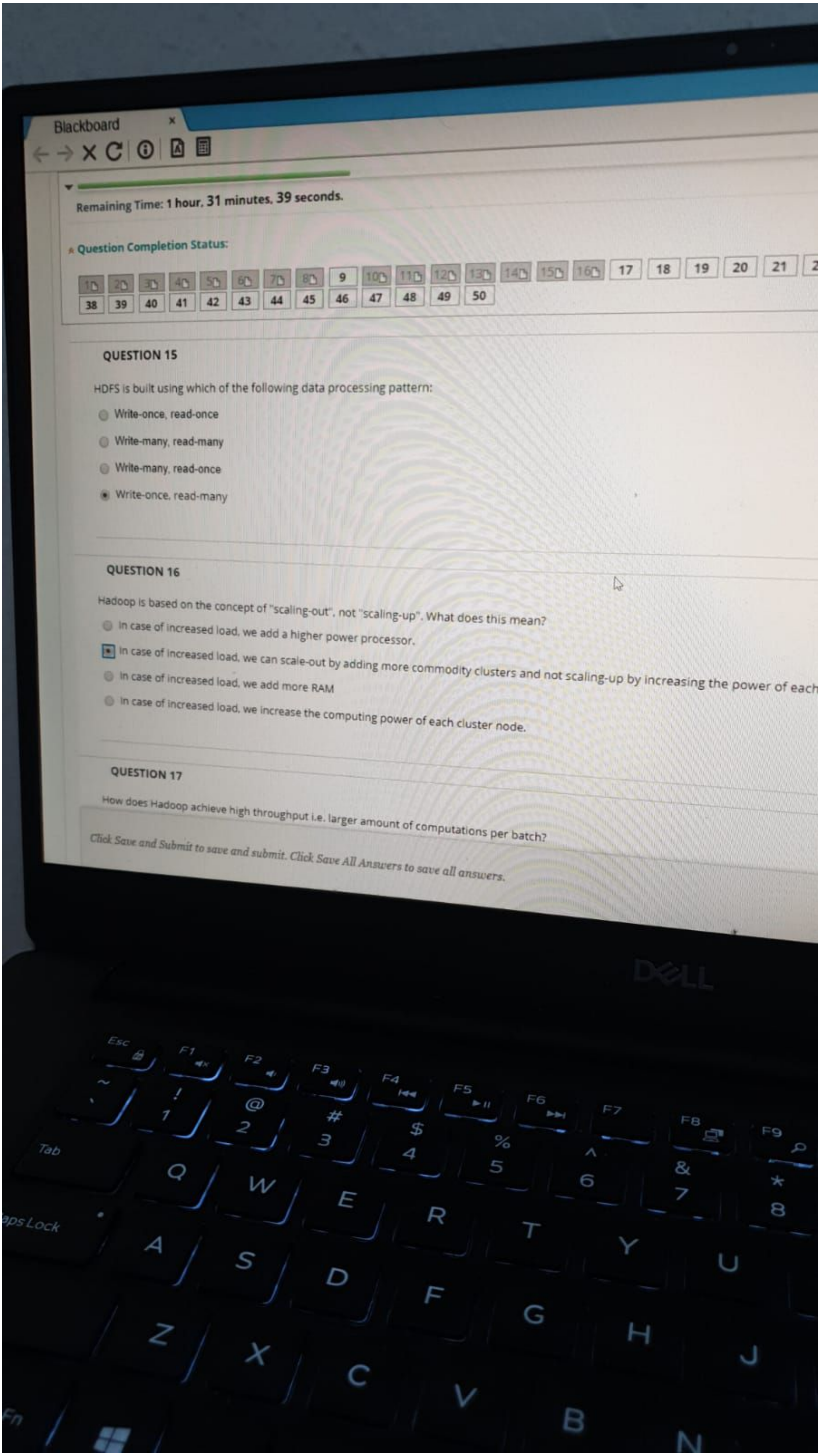
○ Manually done by system administrators

● Secondary NameNode

○ Primary NameNode

QUESTION 15

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: 1 hour, 31 minutes, 39 seconds.

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 2
38 39 40 41 42 43 44 45 46 47 48 49 50

QUESTION 15

HDFS is built using which of the following data processing pattern:

○ Write-once, read-once

○ Write-many, read-many

○ Write-many, read-once

● Write-once, read-many

QUESTION 16

Hadoop is based on the concept of "scaling-out", not "scaling-up". What does this mean?

○ In case of increased load, we add a higher power processor.

● In case of increased load, we can scale-out by adding more commodity clusters and not scaling-up by increasing the power of each

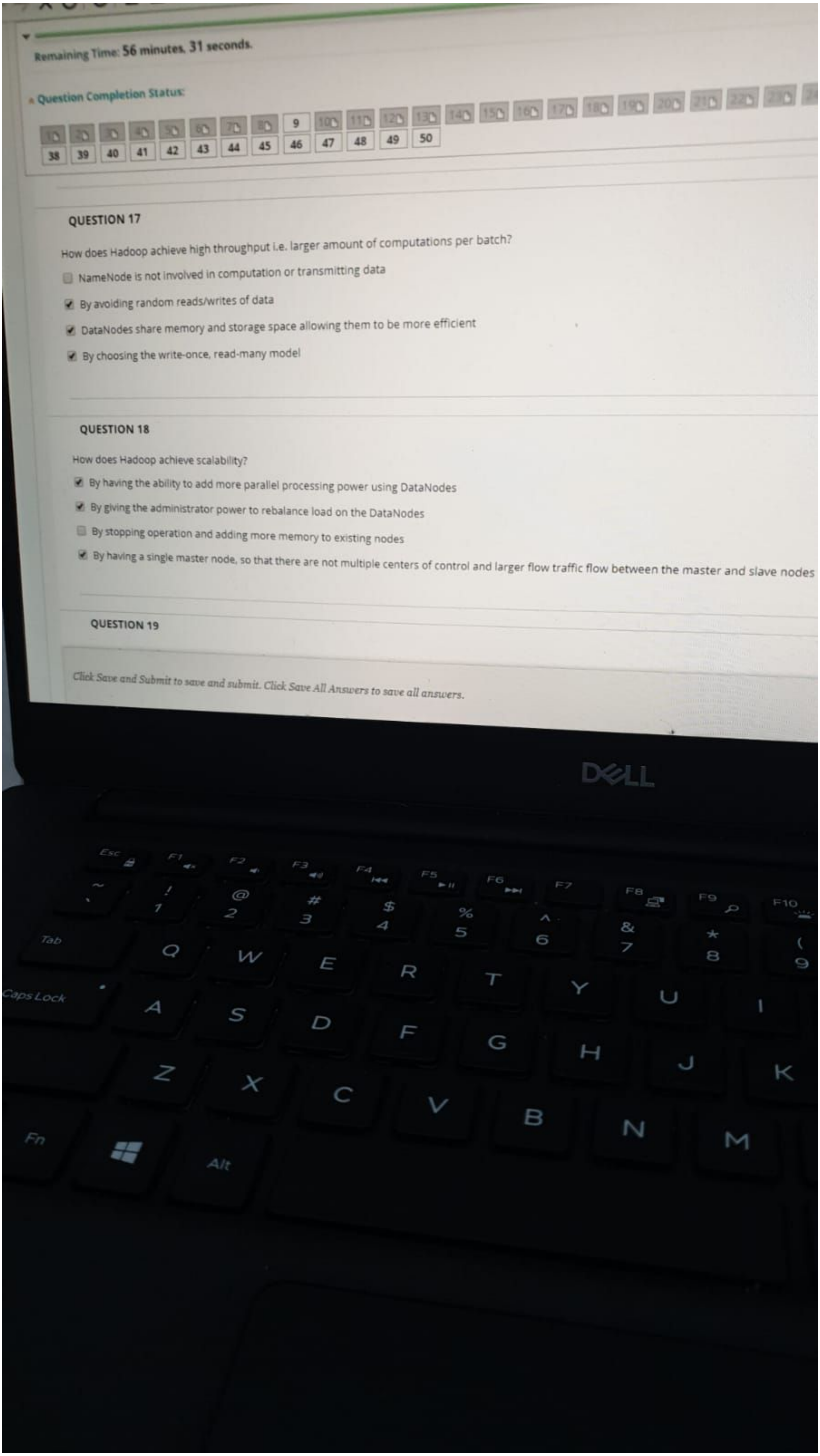
○ In case of increased load, we add more RAM

○ In case of increased load, we increase the computing power of each cluster node.

QUESTION 17

How does Hadoop achieve high throughput i.e. larger amount of computations per batch?

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: 56 minutes, 31 seconds.

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

38 39 40 41 42 43 44 45 46 47 48 49 50

QUESTION 17

How does Hadoop achieve high throughput i.e. larger amount of computations per batch?

□ NameNode is not involved in computation or transmitting data

☑ By avoiding random reads/writes of data

☑ DataNodes share memory and storage space allowing them to be more efficient

☑ By choosing the write-once, read-many model

QUESTION 18

How does Hadoop achieve scalability?

☑ By having the ability to add more parallel processing power using DataNodes

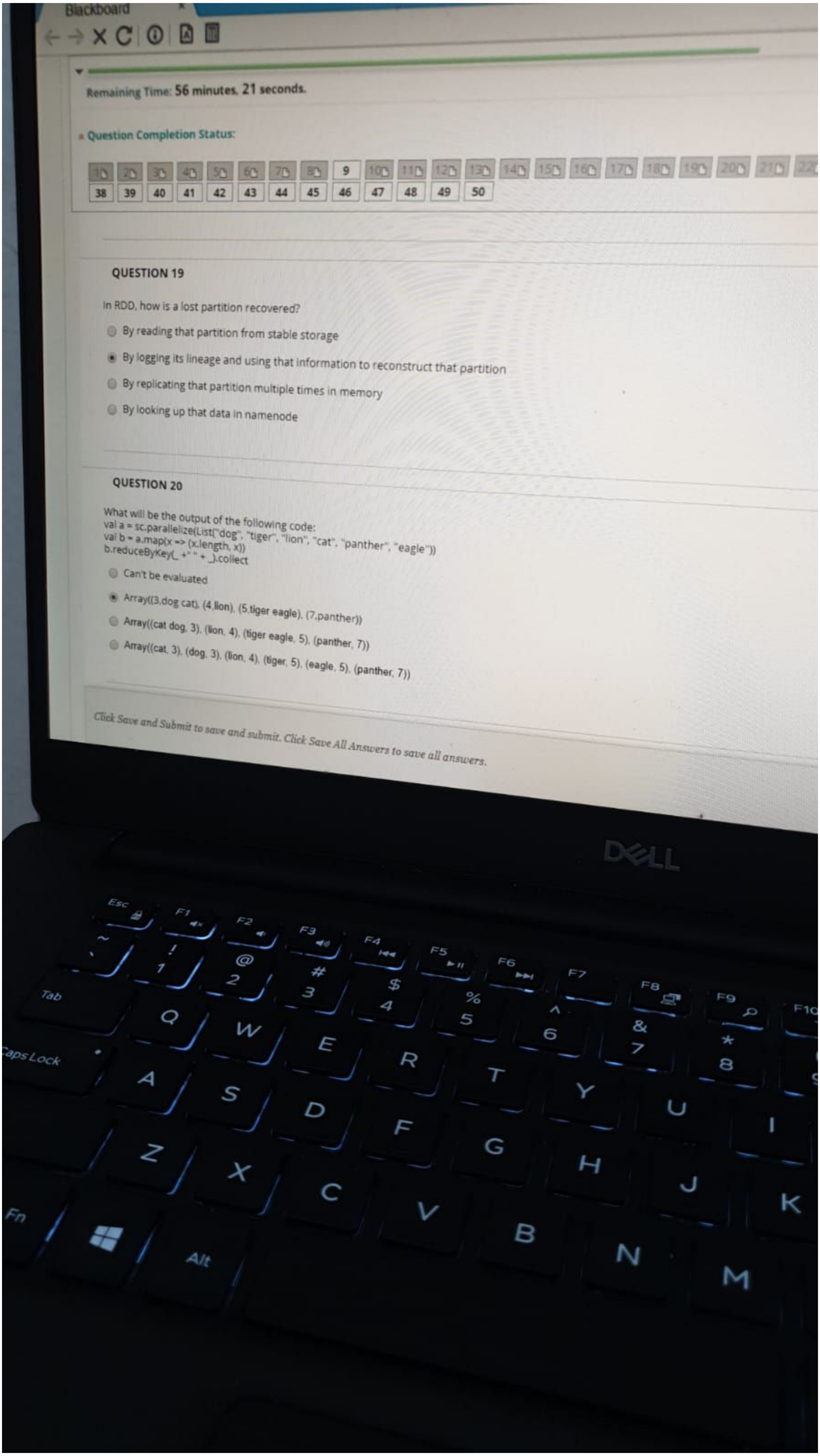
☑ By giving the administrator power to rebalance load on the DataNodes

□ By stopping operation and adding more memory to existing nodes

☑ By having a single master node, so that there are not multiple centers of control and larger flow traffic flow between the master and slave nodes

QUESTION 19

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: **56 minutes, 21 seconds.**

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22

38 39 40 41 42 43 44 45 46 47 48 49 50

QUESTION 19

In RDD, how is a lost partition recovered?

- ○ By reading that partition from stable storage
- ● By logging its lineage and using that information to reconstruct that partition
- ○ By replicating that partition multiple times in memory
- ○ By looking up that data in namenode

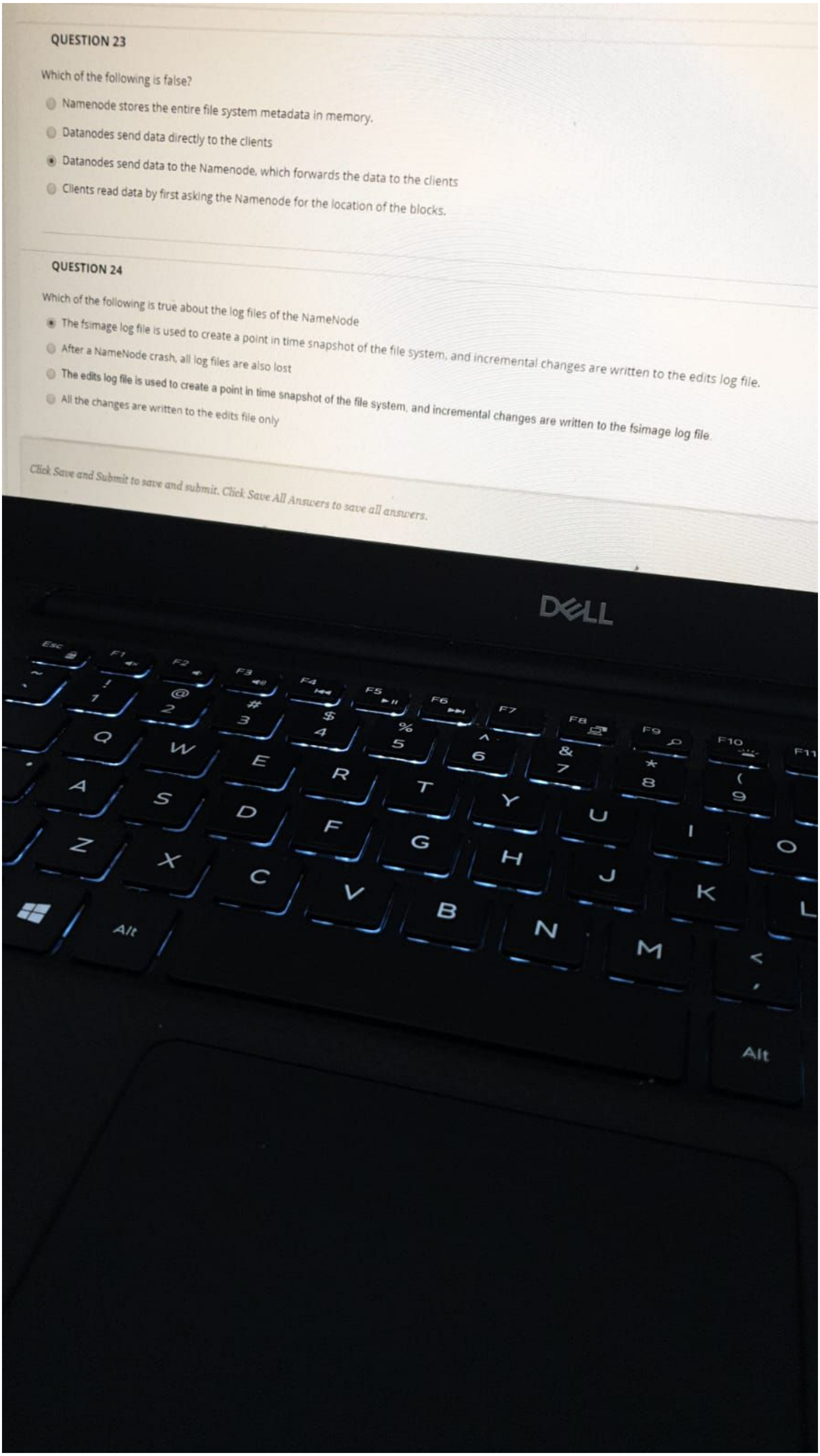
QUESTION 20

What will be the output of the following code:

```
val a = sc.parallelize(List("dog", "tiger", "lion", "cat", "panther", "eagle"))
val b = a.map(x => (x.length, x))
b.reduceByKey(_ +" " + _).collect
```

- ○ Can't be evaluated
- ● Array((3,dog cat), (4,lion), (5,tiger eagle), (7,panther))
- ○ Array((cat dog, 3), (lion, 4), (tiger eagle, 5), (panther, 7))
- ○ Array((cat, 3), (dog, 3), (lion, 4), (tiger, 5), (eagle, 5), (panther, 7))

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



QUESTION 23

Which of the following is false?

○ Namenode stores the entire file system metadata in memory.

○ Datanodes send data directly to the clients

● Datanodes send data to the Namenode, which forwards the data to the clients

○ Clients read data by first asking the Namenode for the location of the blocks.

QUESTION 24

Which of the following is true about the log files of the NameNode

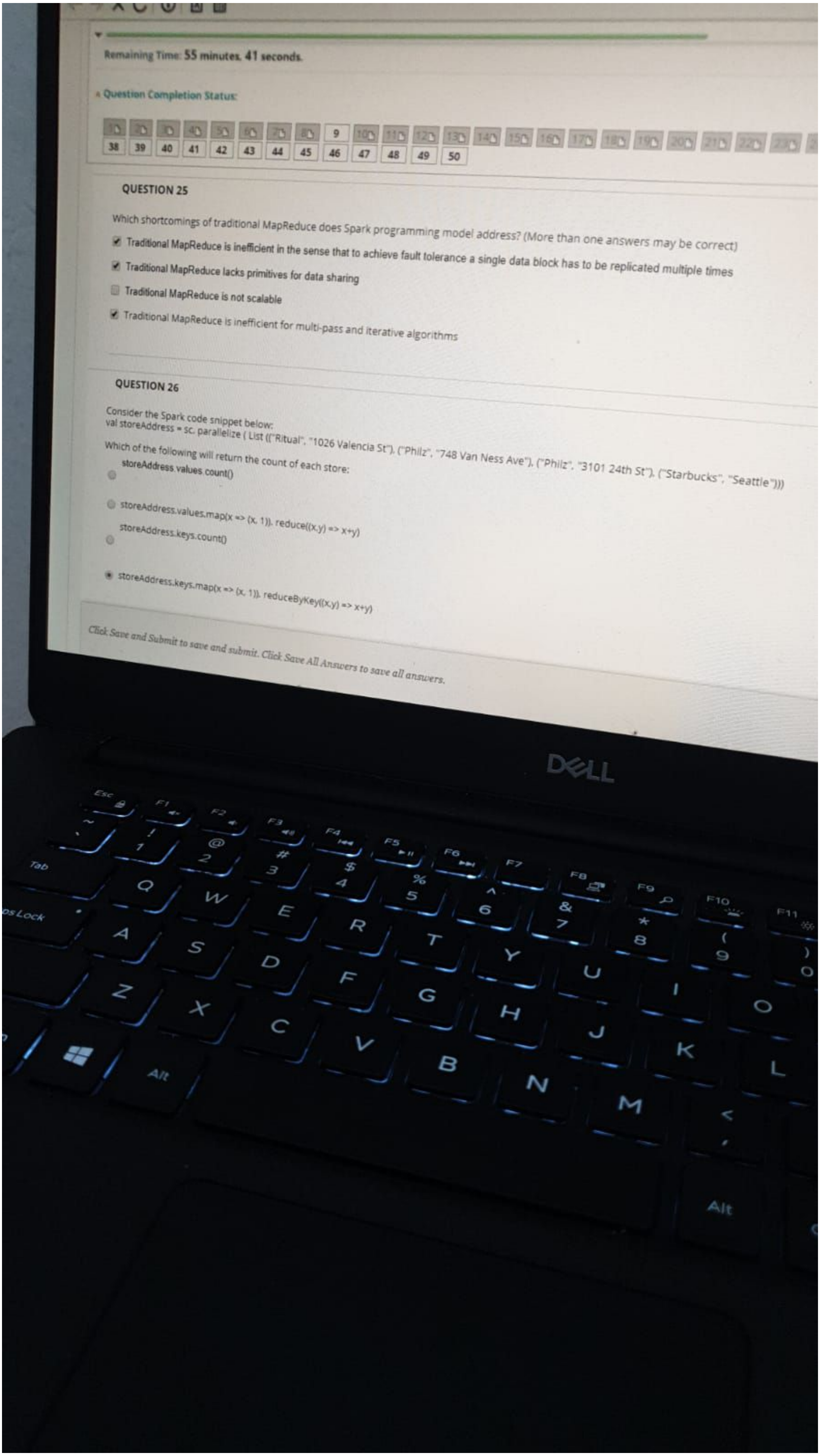
● The fsimage log file is used to create a point in time snapshot of the file system, and incremental changes are written to the edits log file.

○ After a NameNode crash, all log files are also lost

○ The edits log file is used to create a point in time snapshot of the file system, and incremental changes are written to the fsimage log file.

○ All the changes are written to the edits file only

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: 55 minutes, 41 seconds.

Question Completion Status:

QUESTION 25

Which shortcomings of traditional MapReduce does Spark programming model address? (More than one answers may be correct)

☑ Traditional MapReduce is inefficient in the sense that to achieve fault tolerance a single data block has to be replicated multiple times

☑ Traditional MapReduce lacks primitives for data sharing

□ Traditional MapReduce is not scalable

☑ Traditional MapReduce is inefficient for multi-pass and iterative algorithms

QUESTION 26

Consider the Spark code snippet below:
val storeAddress = sc. parallelize (List (("Ritual", "1026 Valencia St"), ("Philz", "748 Van Ness Ave"), ("Philz", "3101 24th St"), ("Starbucks", "Seattle")))

Which of the following will return the count of each store:

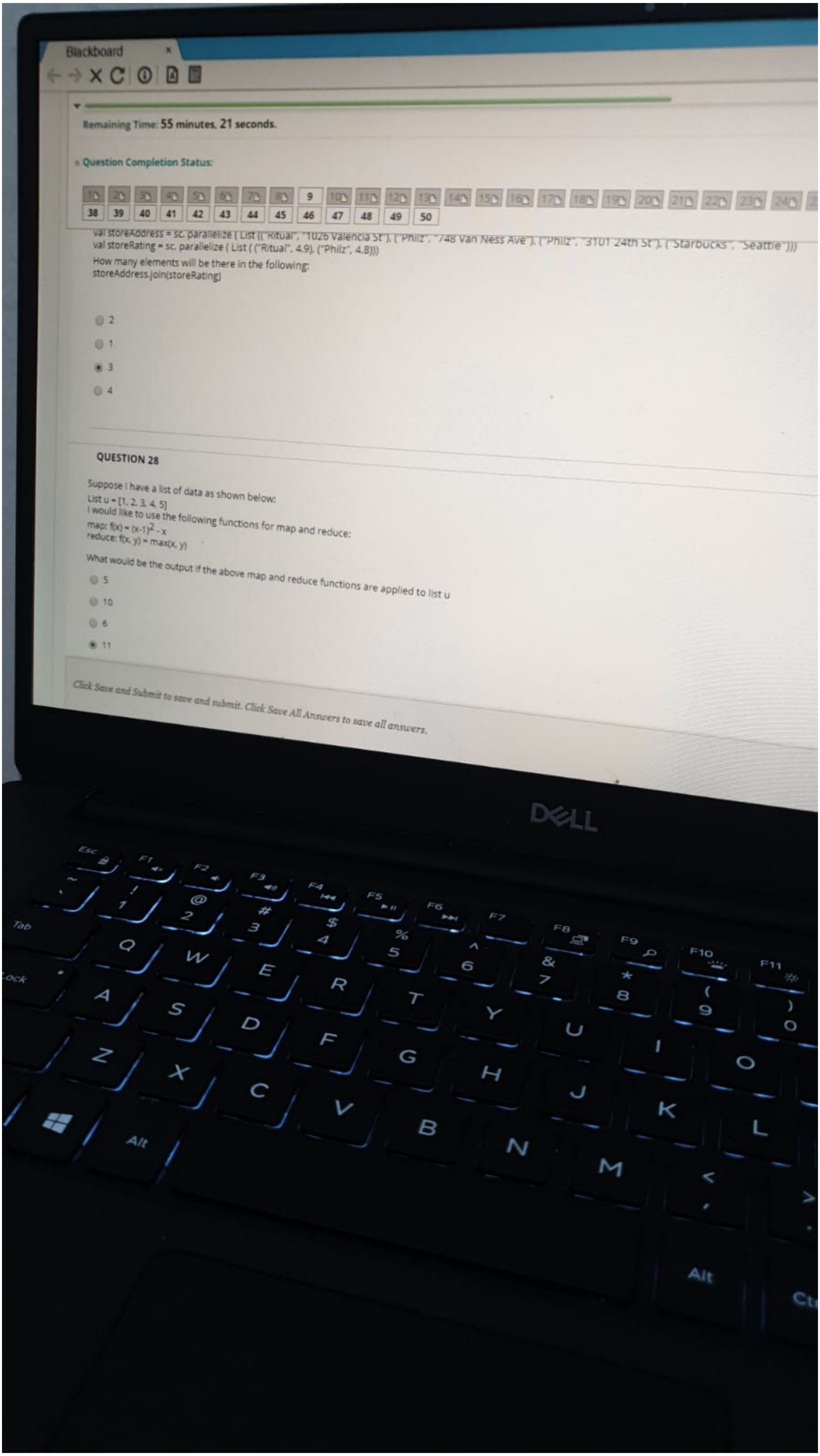
○ storeAddress.values.count()

○ storeAddress.values.map(x => (x, 1)). reduce((x,y) => x+y)

○ storeAddress.keys.count()

● storeAddress.keys.map(x => (x, 1)). reduceByKey((x,y) => x+y)

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: 55 minutes, 21 seconds.

Question Completion Status:

```
val storeAddress = sc. parallelize ( List (("Ritual", "1026 Valencia St"), ("Philz", "748 Van Ness Ave"), ("Philz", "3101 24th St"), ("Starbucks", "Seattle")))
val storeRating = sc. parallelize ( List ( ("Ritual", 4.9), ("Philz", 4.8)))
```

How many elements will be there in the following:
storeAddress.join(storeRating)

○ 2

○ 1

● 3

○ 4

QUESTION 28

Suppose I have a list of data as shown below:
List u = [1, 2, 3, 4, 5]
I would like to use the following functions for map and reduce:
map: $f(x) = (x-1)^2 - x$
reduce: $f(x, y) = max(x, y)$

What would be the output if the above map and reduce functions are applied to list u

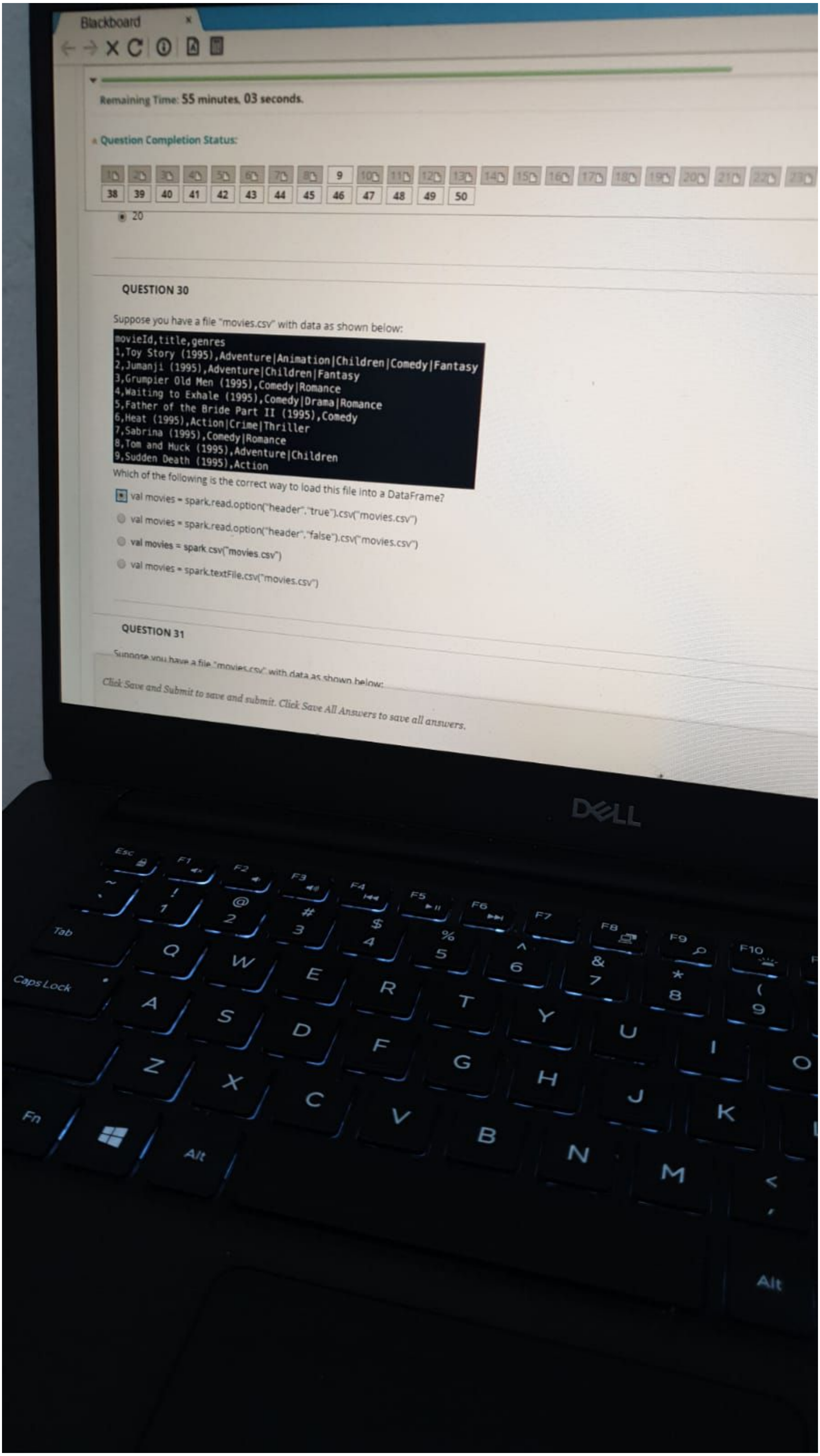
○ 5

○ 10

○ 6

● 11

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: 55 minutes, 03 seconds.

Question Completion Status:

● 20

QUESTION 30

Suppose you have a file "movies.csv" with data as shown below:

```
movieId,title,genres
1,Toy Story (1995),Adventure|Animation|Children|Comedy|Fantasy
2,Jumanji (1995),Adventure|Children|Fantasy
3,Grumpier Old Men (1995),Comedy|Romance
4,Waiting to Exhale (1995),Comedy|Drama|Romance
5,Father of the Bride Part II (1995),Comedy
6,Heat (1995),Action|Crime|Thriller
7,Sabrina (1995),Comedy|Romance
8,Tom and Huck (1995),Adventure|Children
9,Sudden Death (1995),Action
```

Which of the following is the correct way to load this file into a DataFrame?

● val movies = spark.read.option("header","true").csv("movies.csv")

○ val movies = spark.read.option("header","false").csv("movies.csv")

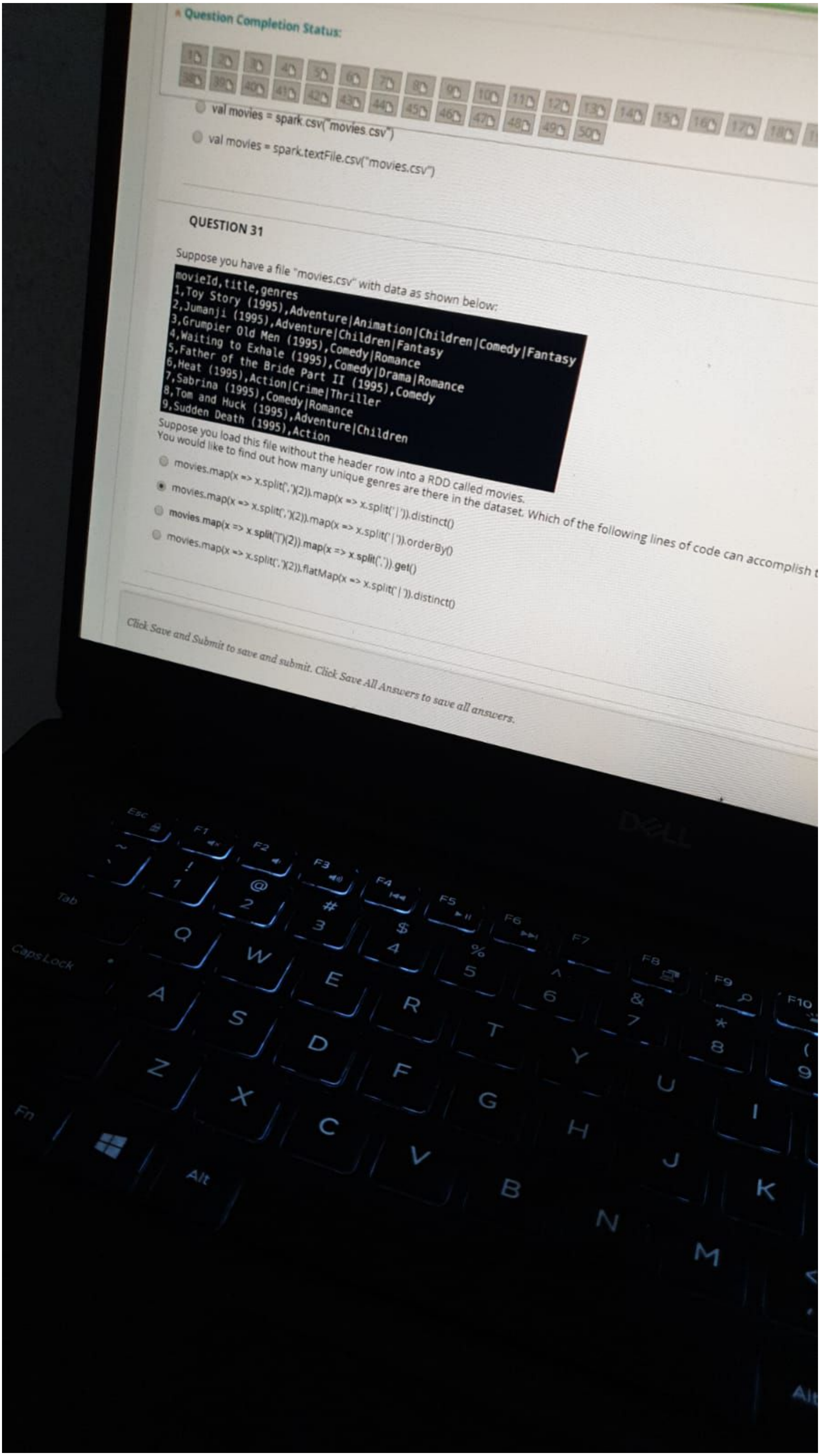
○ val movies = spark.csv("movies.csv")

○ val movies = spark.textFile.csv("movies.csv")

QUESTION 31

Suppose you have a file "movies.csv" with data as shown below:

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



○ val movies = spark.csv("movies.csv")

○ val movies = spark.textFile.csv("movies.csv")

QUESTION 31

Suppose you have a file "movies.csv" with data as shown below:

```
movieId,title,genres
1,Toy Story (1995),Adventure|Animation|Children|Comedy|Fantasy
2,Jumanji (1995),Adventure|Children|Fantasy
3,Grumpier Old Men (1995),Comedy|Romance
4,Waiting to Exhale (1995),Comedy|Drama|Romance
5,Father of the Bride Part II (1995),Comedy
6,Heat (1995),Action|Crime|Thriller
7,Sabrina (1995),Comedy|Romance
8,Tom and Huck (1995),Adventure|Children
9,Sudden Death (1995),Action
```

Suppose you load this file without the header row into a RDD called movies.
You would like to find out how many unique genres are there in the dataset. Which of the following lines of code can accomplish t

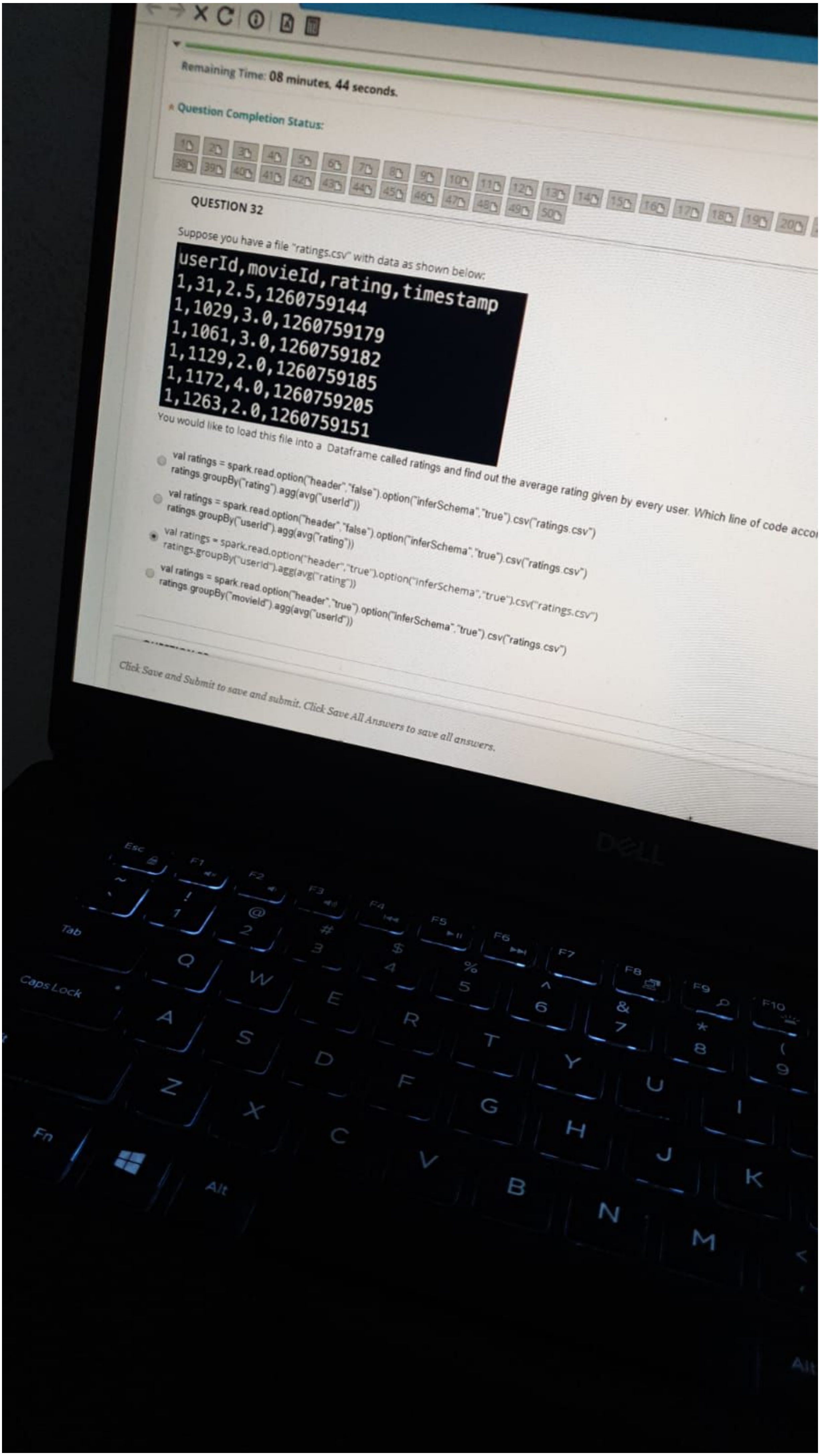
○ movies.map(x => x.split(',')(2)).map(x => x.split('|')).distinct()

● movies.map(x => x.split(',')(2)).map(x => x.split('|')).orderBy()

○ movies.map(x => x.split('|')(2)).map(x => x.split(',')).get()

○ movies.map(x => x.split(',')(2)).flatMap(x => x.split('|')).distinct()

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



QUESTION 32

Suppose you have a file "ratings.csv" with data as shown below:

```
userId,movieId,rating,timestamp
1,31,2.5,1260759144
1,1029,3.0,1260759179
1,1061,3.0,1260759182
1,1129,2.0,1260759185
1,1172,4.0,1260759205
1,1263,2.0,1260759151
```

You would like to load this file into a Dataframe called ratings and find out the average rating given by every user. Which line of code acco

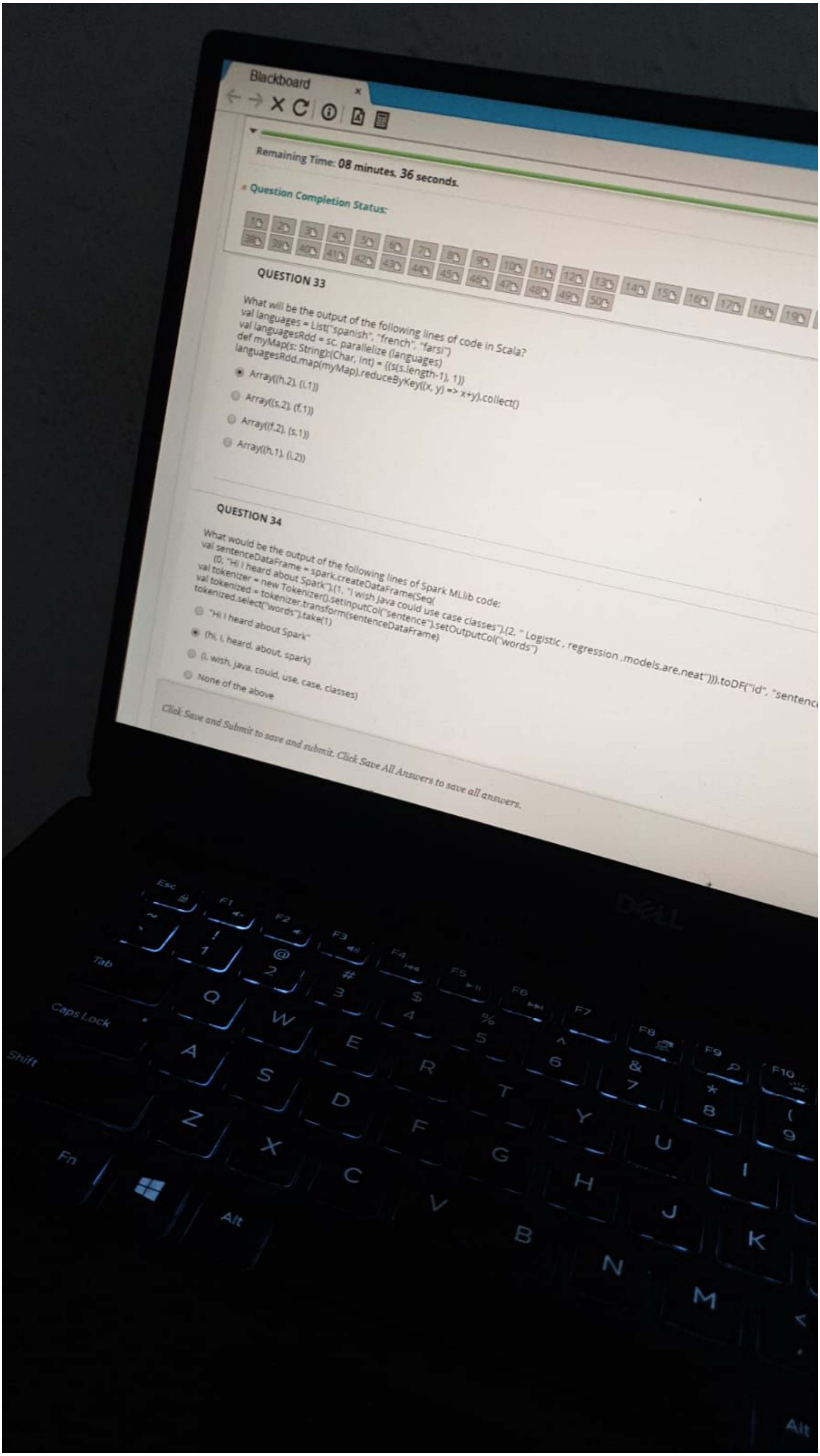
○ val ratings = spark.read.option("header","false").option("inferSchema","true").csv("ratings.csv")
ratings.groupBy("rating").agg(avg("userId"))

○ val ratings = spark.read.option("header","false").option("inferSchema","true").csv("ratings.csv")
ratings.groupBy("userId").agg(avg("rating"))

● val ratings = spark.read.option("header","true").option("inferSchema","true").csv("ratings.csv")
ratings.groupBy("userid").agg(avg("rating"))

○ val ratings = spark.read.option("header","true").option("inferSchema","true").csv("ratings.csv")
ratings.groupBy("movieId").agg(avg("userId"))

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: **08 minutes, 36 seconds.**

Question Completion Status:

QUESTION 33

What will be the output of the following lines of code in Scala?

```
val languages = List("spanish", "french", "farsi")
val languagesRdd = sc. parallelize (languages)
def myMap(s: String):(Char, Int) = ((s(s.length-1), 1))
languagesRdd.map(myMap).reduceByKey((x, y) => x+y).collect()
```

● Array((h,2), (i,1))

○ Array((s,2), (f,1))

○ Array((f,2), (s,1))

○ Array((h,1), (i,2))

QUESTION 34

What would be the output of the following lines of Spark MLlib code:

```
val sentenceDataFrame = spark.createDataFrame(Seq(
  (0, "Hi I heard about Spark"),(1, "I wish Java could use case classes"),(2, " Logistic , regression ,models,are,neat"))).toDF("id", "sentenc
val tokenizer = new Tokenizer().setInputCol("sentence").setOutputCol("words")
val tokenized = tokenizer.transform(sentenceDataFrame)
tokenized.select("words").take(1)
```

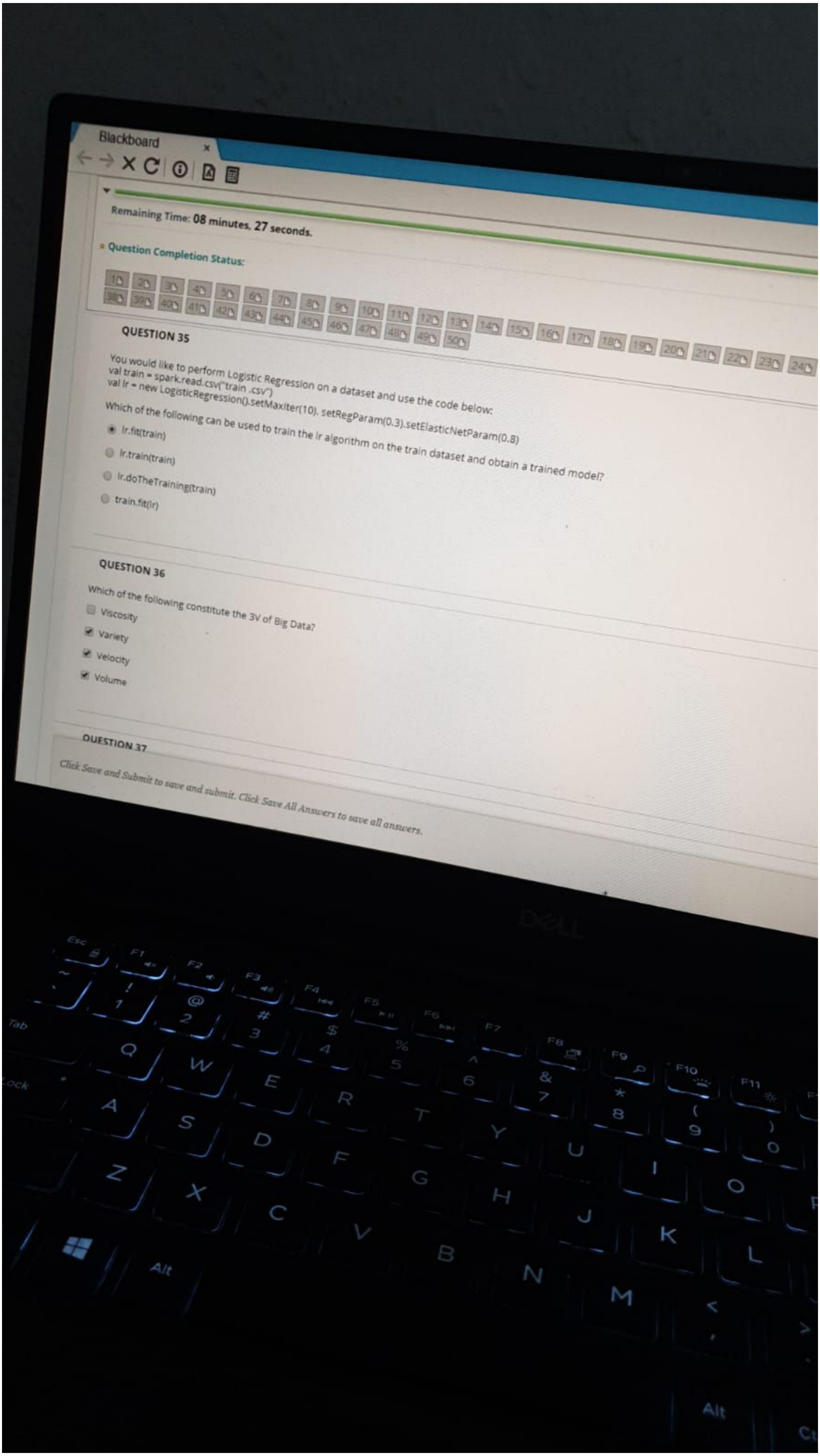
○ "Hi I heard about Spark"

● (hi, i, heard, about, spark)

○ (i, wish, java, could, use, case, classes)

○ None of the above

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: **08 minutes, 27 seconds.**

Question Completion Status:

QUESTION 35

You would like to perform Logistic Regression on a dataset and use the code below:

```
val train = spark.read.csv("train .csv")
val lr = new LogisticRegression().setMaxIter(10). setRegParam(0.3).setElasticNetParam(0.8)
```

Which of the following can be used to train the lr algorithm on the train dataset and obtain a trained model?

● lr.fit(train)

○ lr.train(train)

○ lr.doTheTraining(train)

○ train.fit(lr)

QUESTION 36

Which of the following constitute the 3V of Big Data?

□ Viscosity

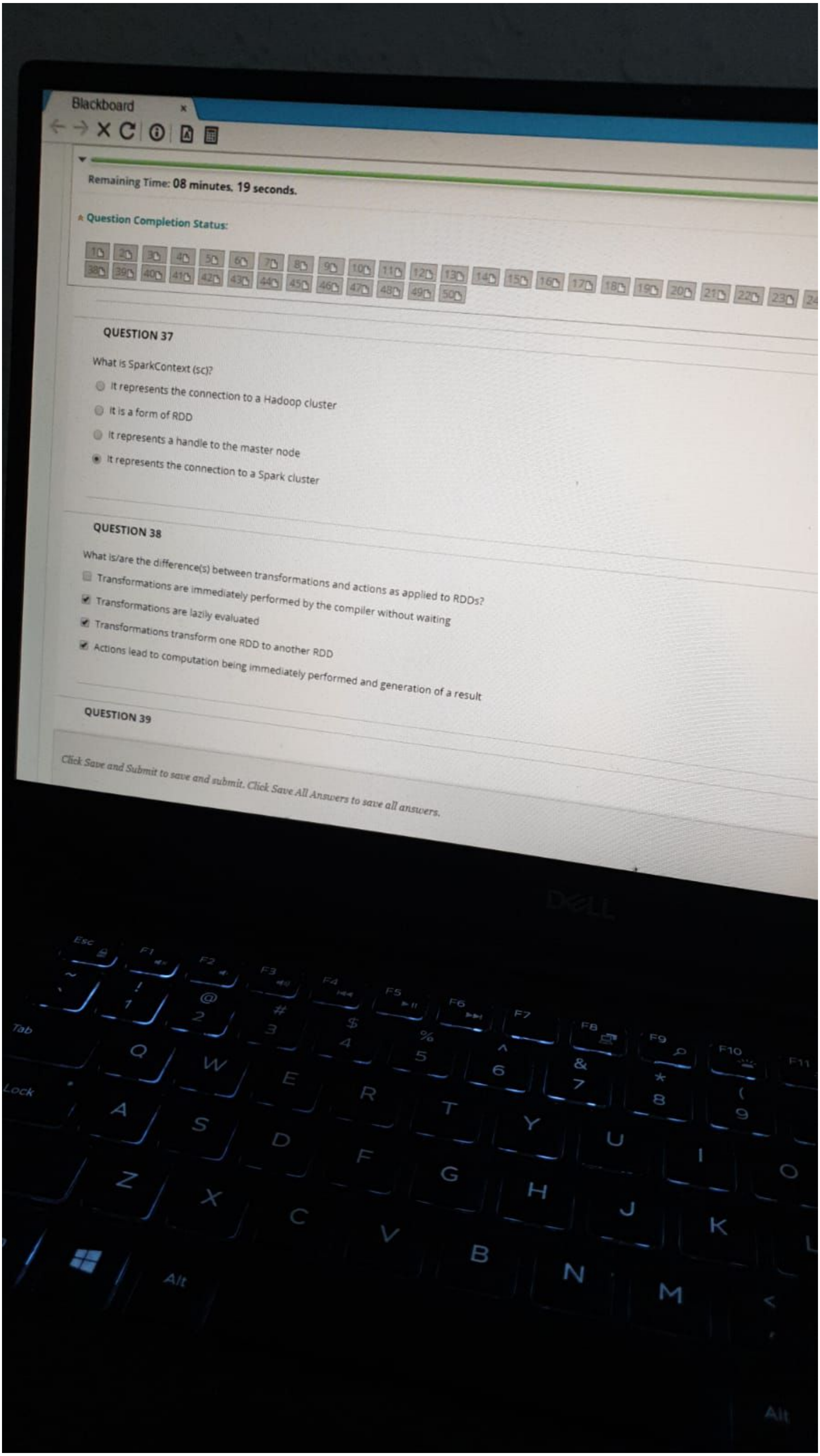
☑ Variety

☑ Velocity

☑ Volume

QUESTION 37

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: **08 minutes, 19 seconds.**

Question Completion Status:

QUESTION 37

What is SparkContext (sc)?

○ It represents the connection to a Hadoop cluster

○ It is a form of RDD

○ It represents a handle to the master node

● It represents the connection to a Spark cluster

QUESTION 38

What is/are the difference(s) between transformations and actions as applied to RDDs?

□ Transformations are immediately performed by the compiler without waiting

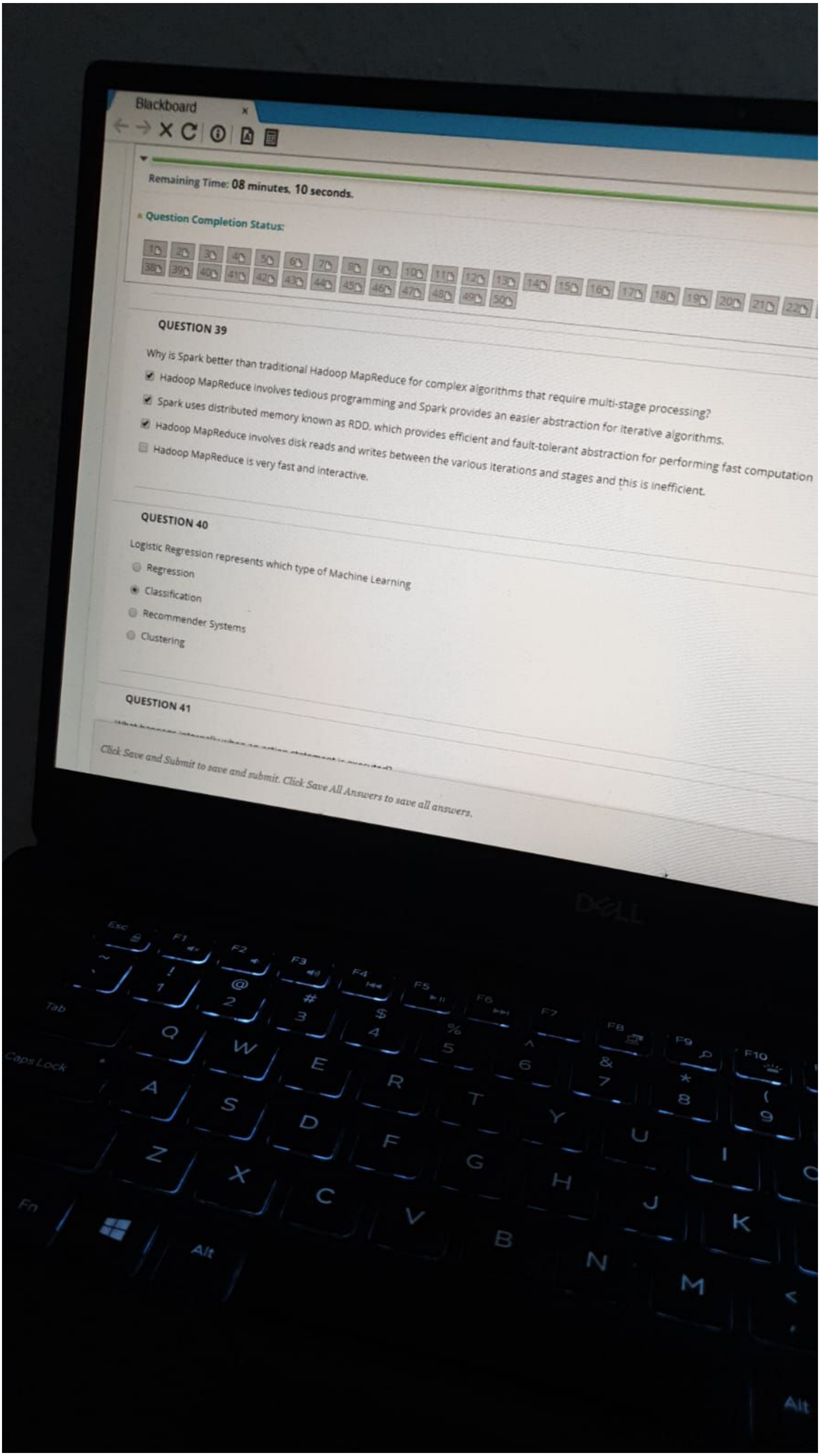
☑ Transformations are lazily evaluated

☑ Transformations transform one RDD to another RDD

☑ Actions lead to computation being immediately performed and generation of a result

QUESTION 39

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: **08 minutes, 10 seconds.**

Question Completion Status:

QUESTION 39

Why is Spark better than traditional Hadoop MapReduce for complex algorithms that require multi-stage processing?

- ☑ Hadoop MapReduce involves tedious programming and Spark provides an easier abstraction for iterative algorithms.
- ☑ Spark uses distributed memory known as RDD, which provides efficient and fault-tolerant abstraction for performing fast computation
- ☑ Hadoop MapReduce involves disk reads and writes between the various iterations and stages and this is inefficient.
- □ Hadoop MapReduce is very fast and interactive.

QUESTION 40

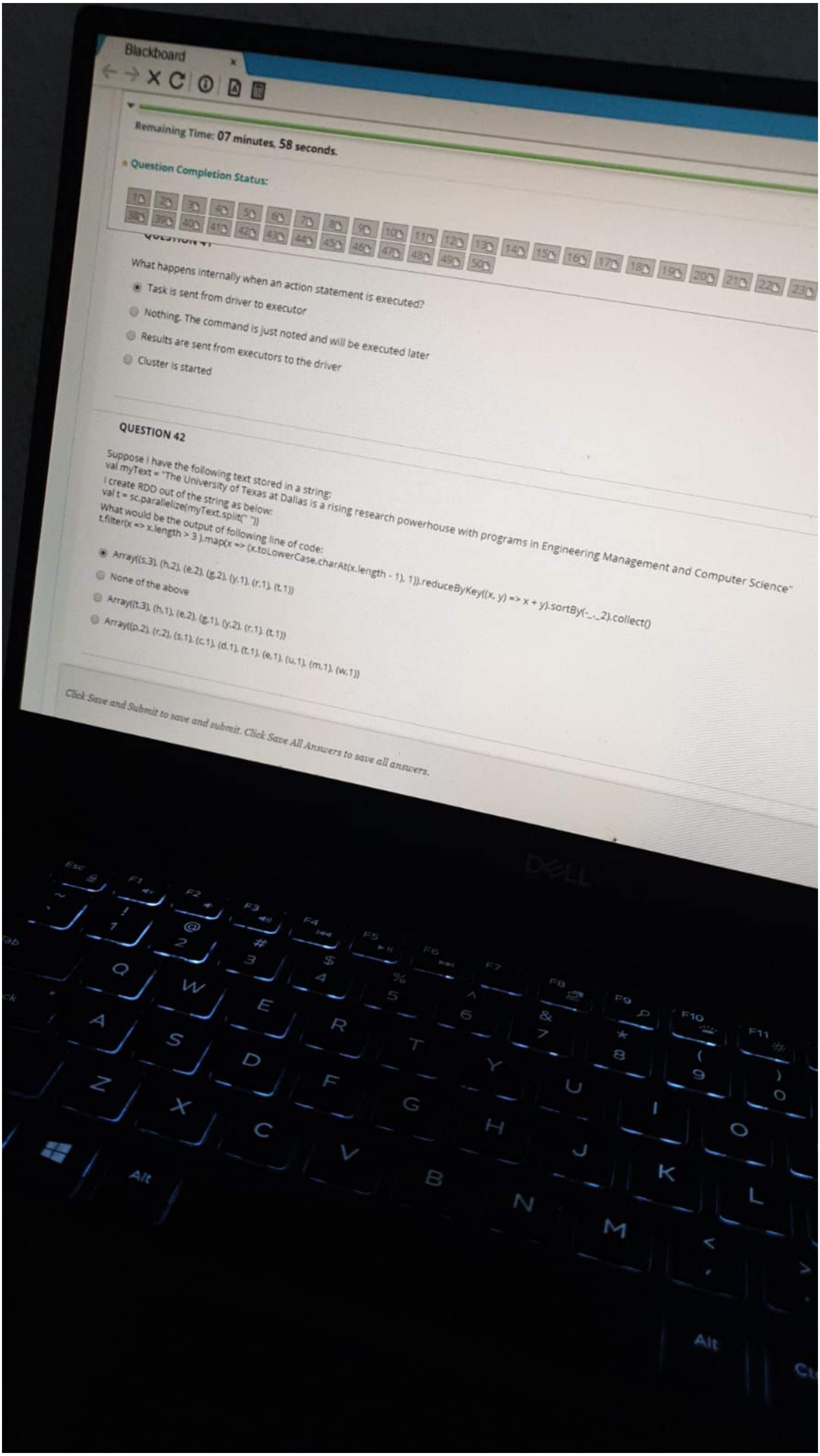
Logistic Regression represents which type of Machine Learning

- ○ Regression
- ● Classification
- ○ Recommender Systems
- ○ Clustering

QUESTION 41

What happens internally when an action statement is executed?

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



Remaining Time: **07 minutes, 58 seconds.**

Question Completion Status:

QUESTION 41

What happens internally when an action statement is executed?

● Task is sent from driver to executor

○ Nothing. The command is just noted and will be executed later

○ Results are sent from executors to the driver

○ Cluster is started

QUESTION 42

Suppose I have the following text stored in a string:
val myText = "The University of Texas at Dallas is a rising research powerhouse with programs in Engineering Management and Computer Science"
I create RDD out of the string as below:
val t = sc.parallelize(myText.split(" "))
What would be the output of following line of code:
t.filter(x => x.length > 3).map(x => (x.toLowerCase.charAt(x.length - 1), 1)).reduceByKey((x, y) => x + y).sortBy(-_._2).collect()

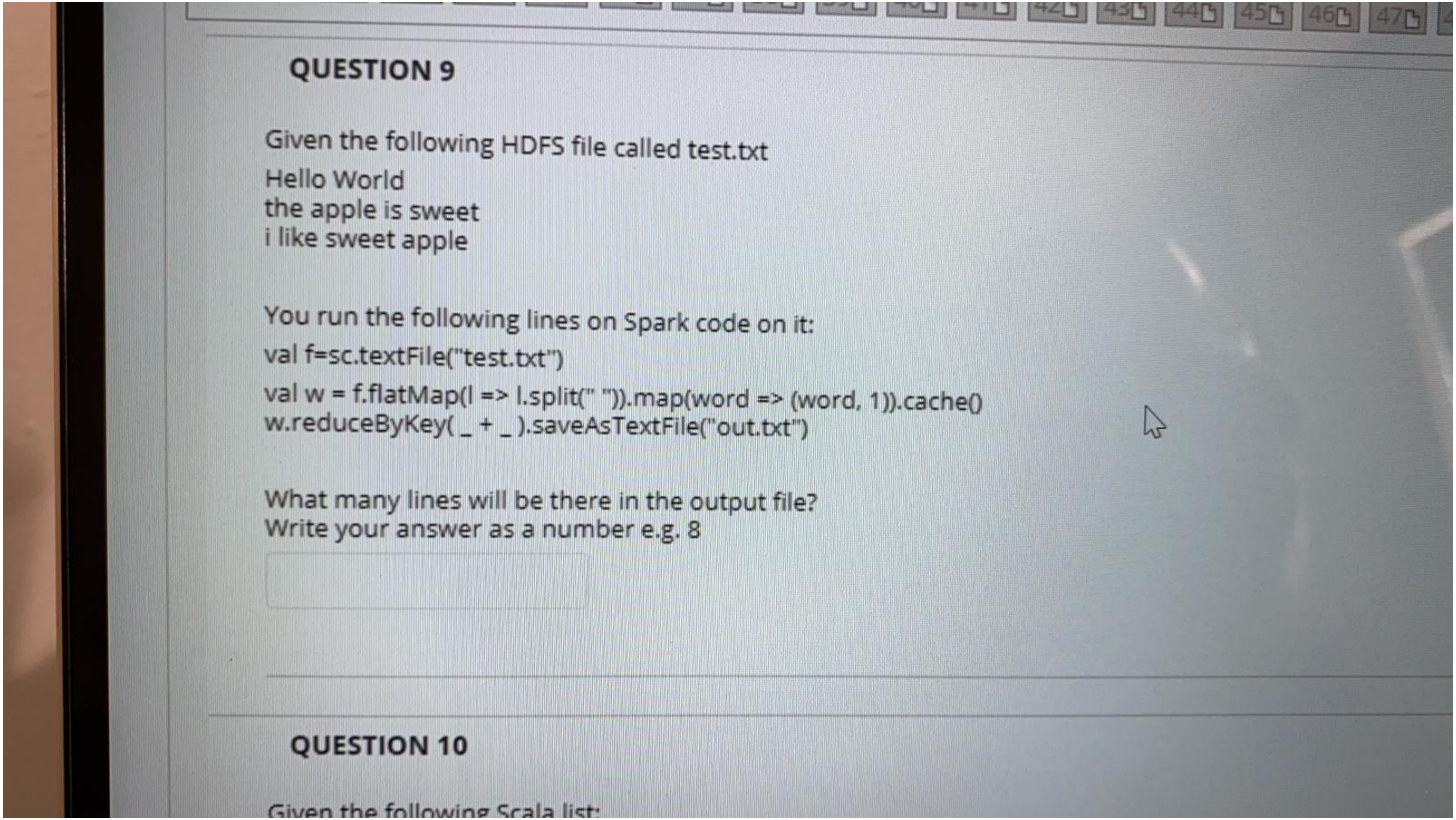
● Array((s,3), (h,2), (e,2), (g,2), (y,1), (r,1), (t,1))

○ None of the above

○ Array((t,3), (h,1), (e,2), (g,1), (y,2), (r,1), (t,1))

○ Array((p,2), (r,2), (s,1), (c,1), (d,1), (t,1), (e,1), (u,1), (m,1), (w,1))

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



QUESTION 9

Given the following HDFS file called test.txt

Hello World
the apple is sweet
i like sweet apple

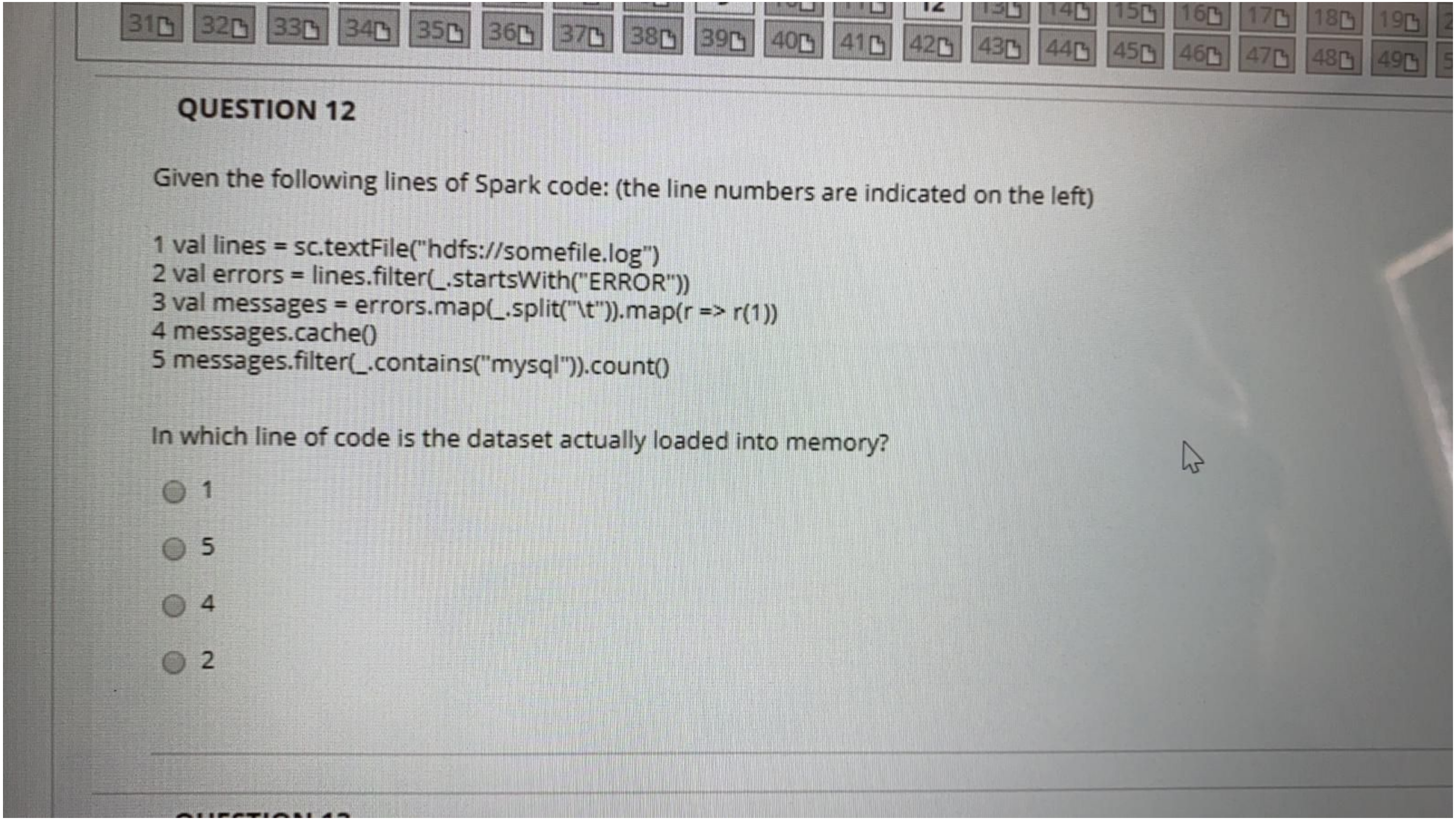
You run the following lines on Spark code on it:

```
val f=sc.textFile("test.txt")
val w = f.flatMap(l => l.split(" ")).map(word => (word, 1)).cache()
w.reduceByKey( _ + _ ).saveAsTextFile("out.txt")
```

What many lines will be there in the output file?
Write your answer as a number e.g. 8

QUESTION 10

Given the following Scala list:



QUESTION 12

Given the following lines of Spark code: (the line numbers are indicated on the left)

```
1 val lines = sc.textFile("hdfs://somefile.log")
2 val errors = lines.filter(_.startsWith("ERROR"))
3 val messages = errors.map(_.split("\t")).map(r => r(1))
4 messages.cache()
5 messages.filter(_.contains("mysql")).count()
```

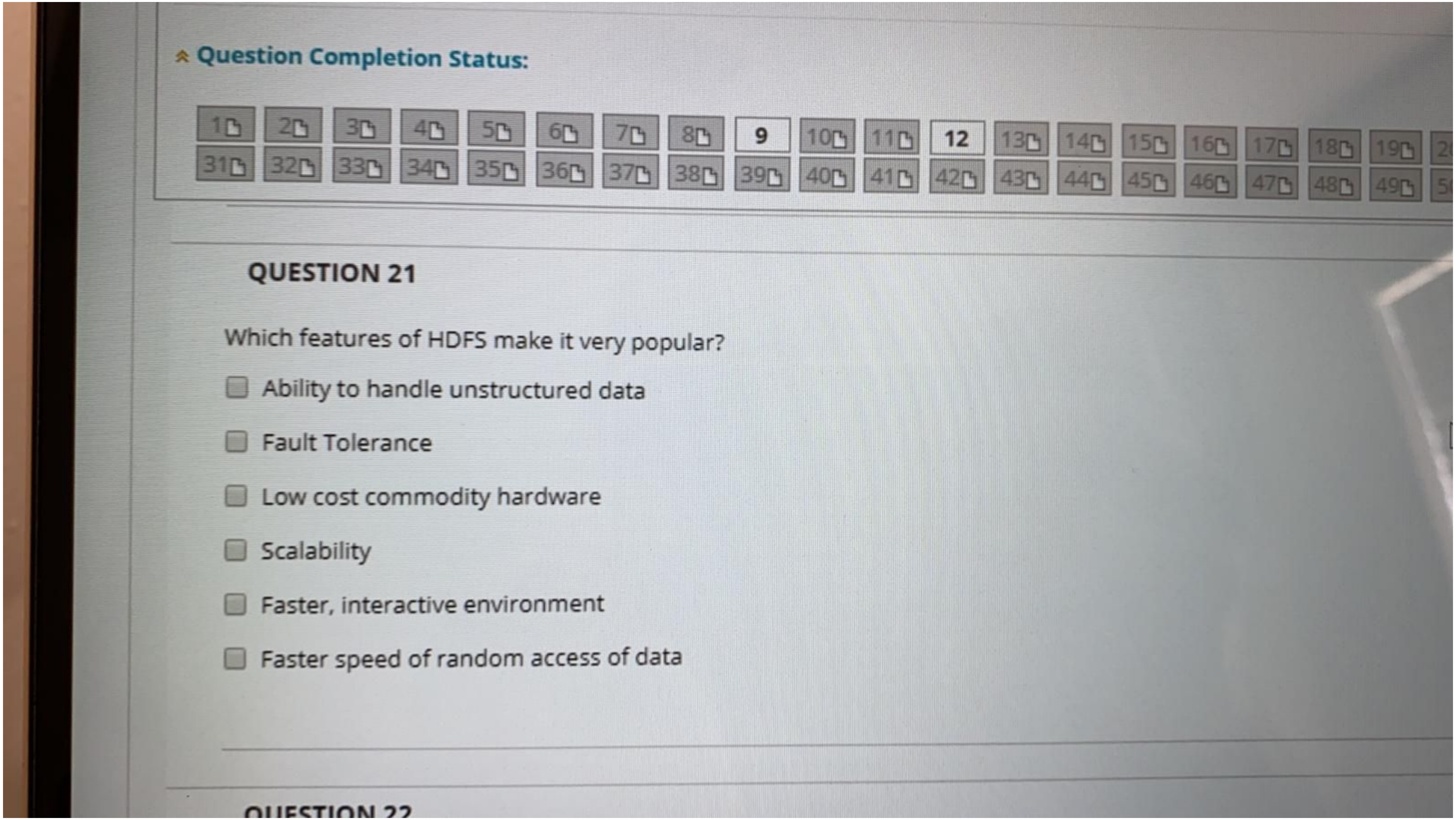
In which line of code is the dataset actually loaded into memory?

○ 1

○ 5

○ 4

○ 2



Question Completion Status:

QUESTION 21

Which features of HDFS make it very popular?

□ Ability to handle unstructured data

□ Fault Tolerance

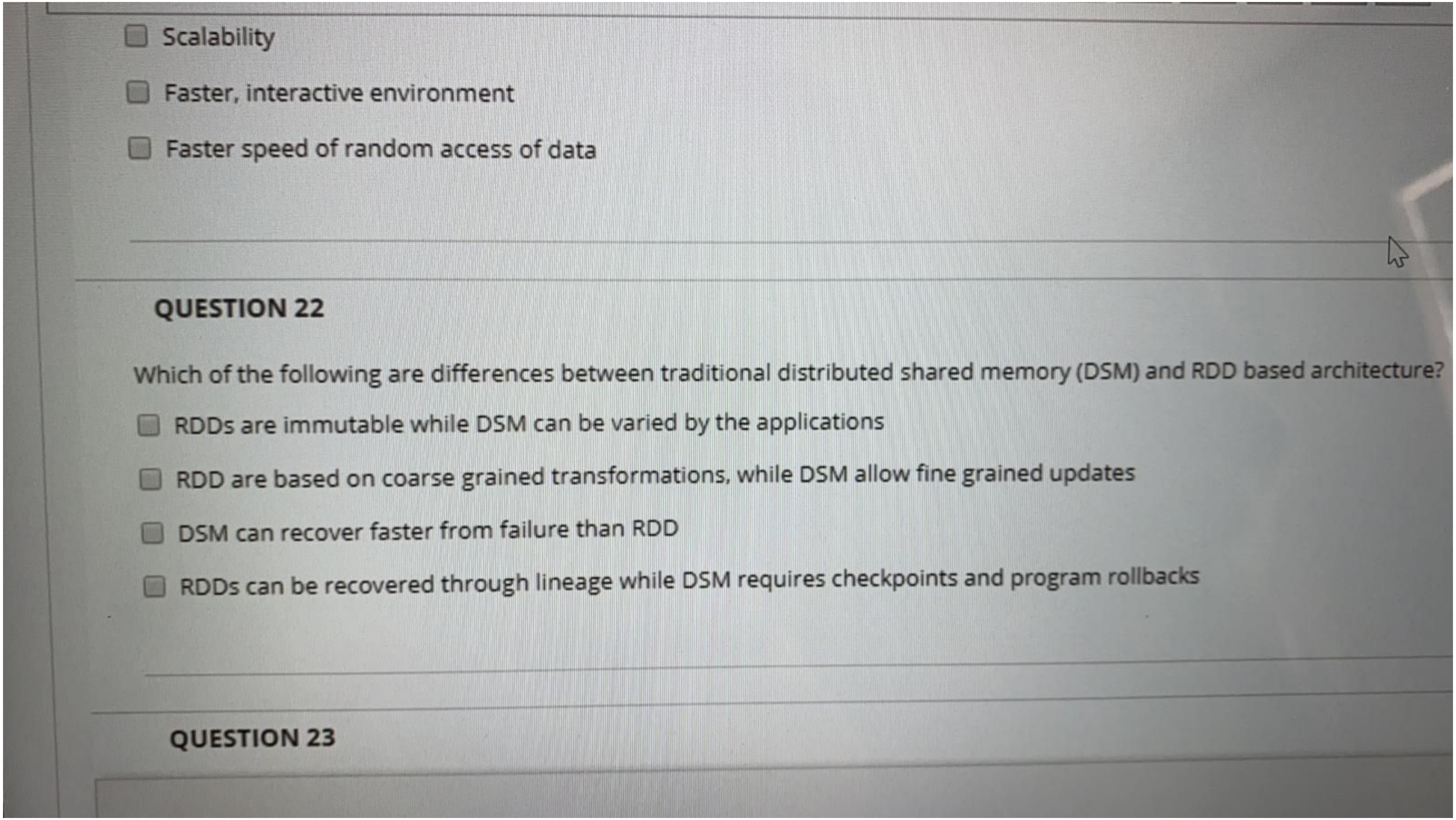
□ Low cost commodity hardware

□ Scalability

□ Faster, interactive environment

□ Faster speed of random access of data

QUESTION 22



□ Scalability

□ Faster, interactive environment

□ Faster speed of random access of data

QUESTION 22

Which of the following are differences between traditional distributed shared memory (DSM) and RDD based architecture?

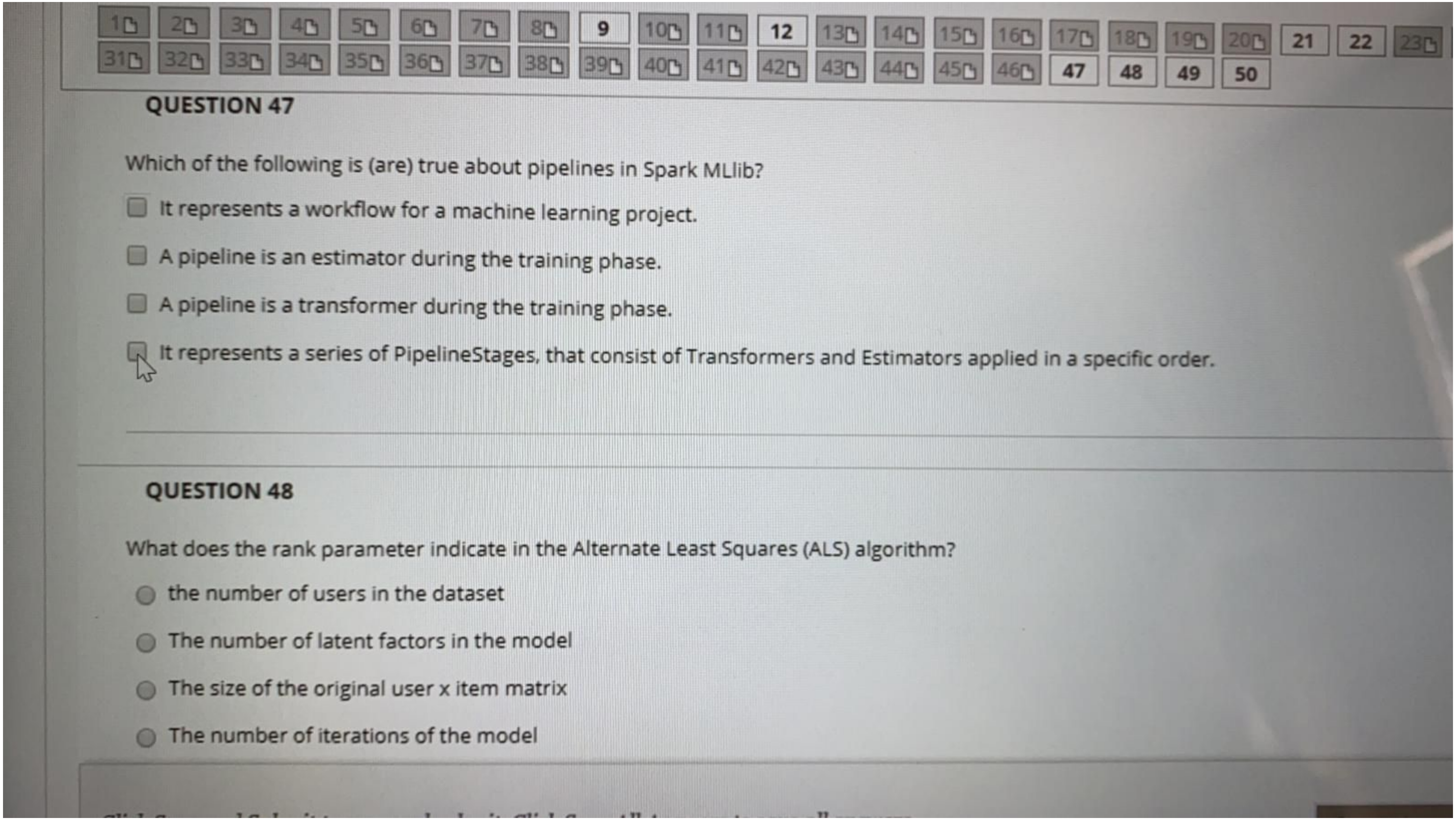
□ RDDs are immutable while DSM can be varied by the applications

□ RDD are based on coarse grained transformations, while DSM allow fine grained updates

□ DSM can recover faster from failure than RDD

□ RDDs can be recovered through lineage while DSM requires checkpoints and program rollbacks

QUESTION 23



QUESTION 47

Which of the following is (are) true about pipelines in Spark MLlib?

□ It represents a workflow for a machine learning project.

□ A pipeline is an estimator during the training phase.

□ A pipeline is a transformer during the training phase.

□ It represents a series of PipelineStages, that consist of Transformers and Estimators applied in a specific order.

QUESTION 48

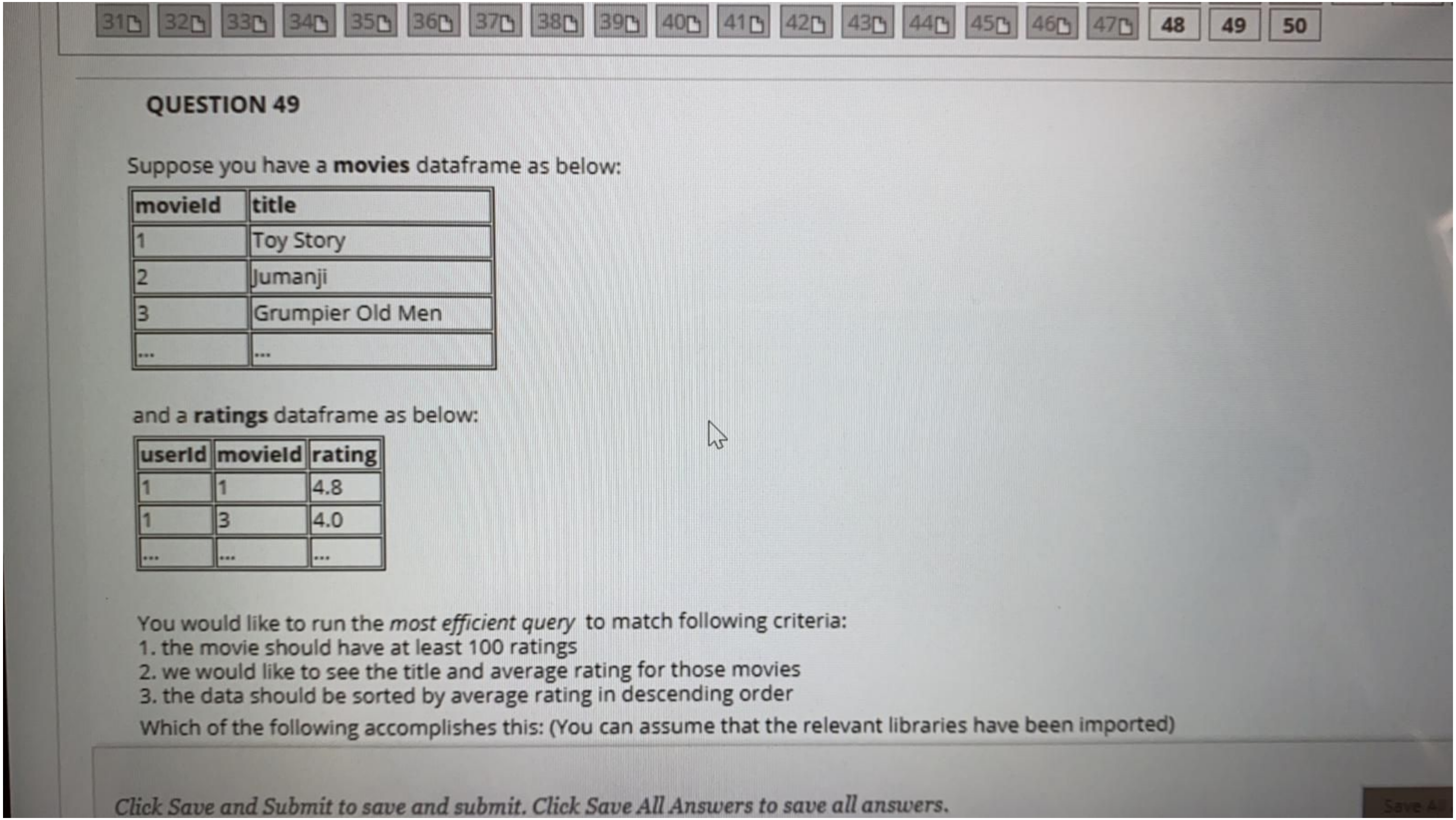
What does the rank parameter indicate in the Alternate Least Squares (ALS) algorithm?

○ the number of users in the dataset

○ The number of latent factors in the model

○ The size of the original user x item matrix

○ The number of iterations of the model



QUESTION 49

Suppose you have a **movies** dataframe as below:

movieId	title
1	Toy Story
2	Jumanji
3	Grumpier Old Men
...	...

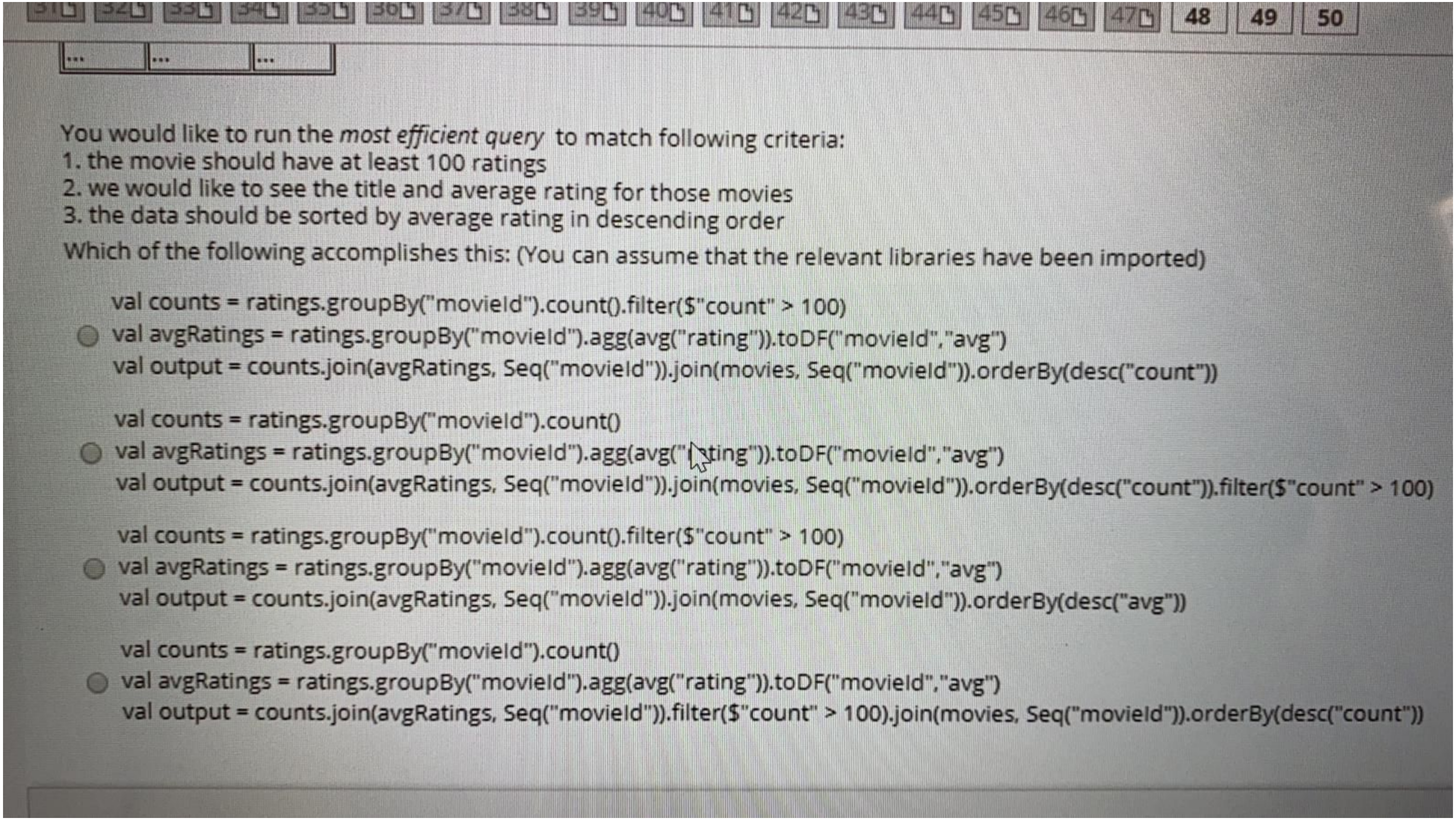
and a **ratings** dataframe as below:

userId	movieId	rating
1	1	4.8
1	3	4.0
...	...	...

You would like to run the *most efficient query* to match following criteria:
1. the movie should have at least 100 ratings
2. we would like to see the title and average rating for those movies
3. the data should be sorted by average rating in descending order

Which of the following accomplishes this: (You can assume that the relevant libraries have been imported)

Click Save and Submit to save and submit. Click Save All Answers to save all answers.



You would like to run the *most efficient query* to match following criteria:
1. the movie should have at least 100 ratings
2. we would like to see the title and average rating for those movies
3. the data should be sorted by average rating in descending order

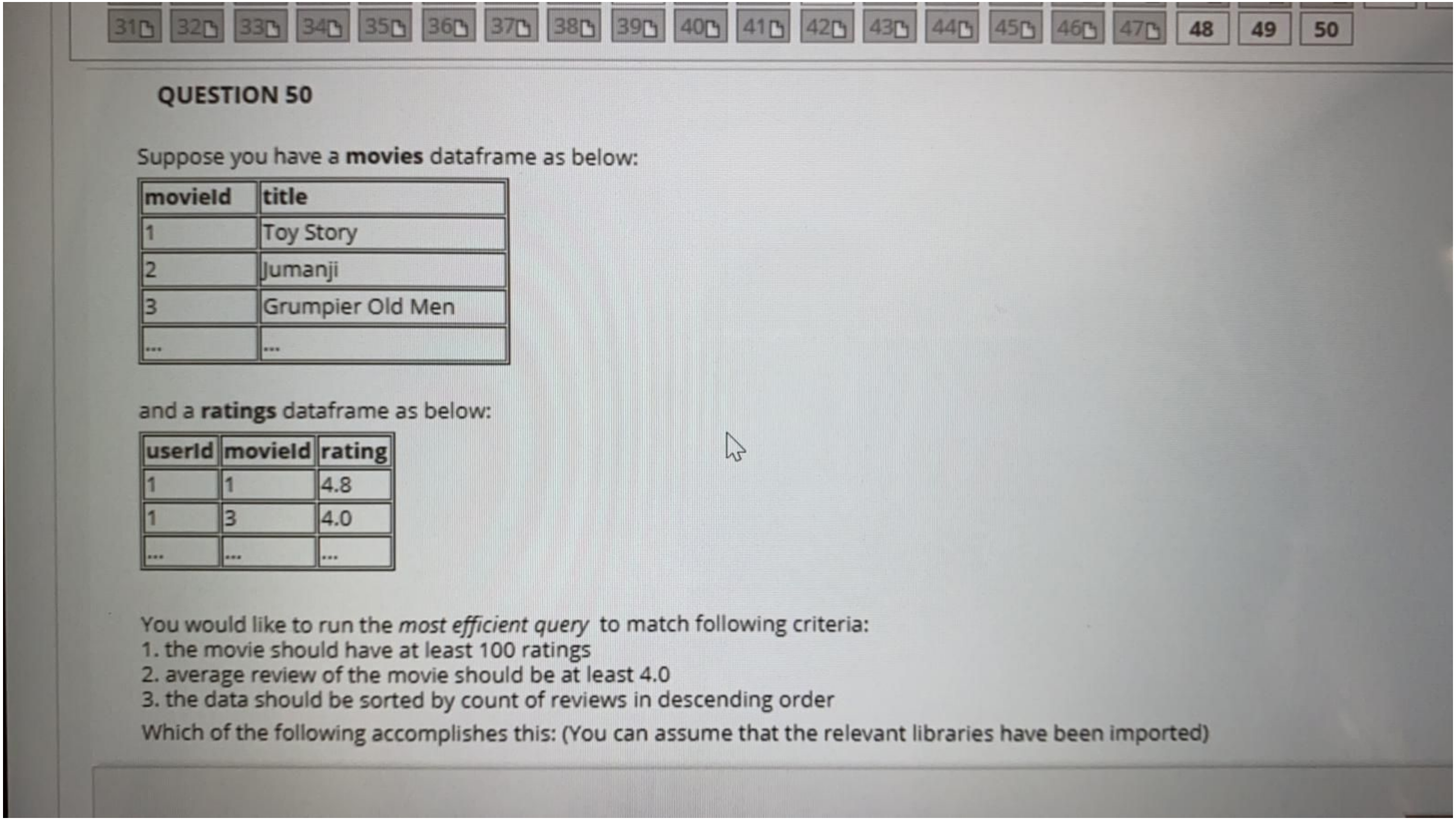
Which of the following accomplishes this: (You can assume that the relevant libraries have been imported)

○
```
val counts = ratings.groupBy("movieId").count().filter($"count" > 100)
val avgRatings = ratings.groupBy("movieId").agg(avg("rating")).toDF("movieId","avg")
val output = counts.join(avgRatings, Seq("movieId")).join(movies, Seq("movieId")).orderBy(desc("count"))
```

○
```
val counts = ratings.groupBy("movieId").count()
val avgRatings = ratings.groupBy("movieId").agg(avg("rating")).toDF("movieId","avg")
val output = counts.join(avgRatings, Seq("movieId")).join(movies, Seq("movieId")).orderBy(desc("count")).filter($"count" > 100)
```

○
```
val counts = ratings.groupBy("movieId").count().filter($"count" > 100)
val avgRatings = ratings.groupBy("movieId").agg(avg("rating")).toDF("movieId","avg")
val output = counts.join(avgRatings, Seq("movieId")).join(movies, Seq("movieId")).orderBy(desc("avg"))
```

○
```
val counts = ratings.groupBy("movieId").count()
val avgRatings = ratings.groupBy("movieId").agg(avg("rating")).toDF("movieId","avg")
val output = counts.join(avgRatings, Seq("movieId")).filter($"count" > 100).join(movies, Seq("movieId")).orderBy(desc("count"))
```



QUESTION 50

Suppose you have a **movies** dataframe as below:

movieId	title
1	Toy Story
2	Jumanji
3	Grumpier Old Men
...	...

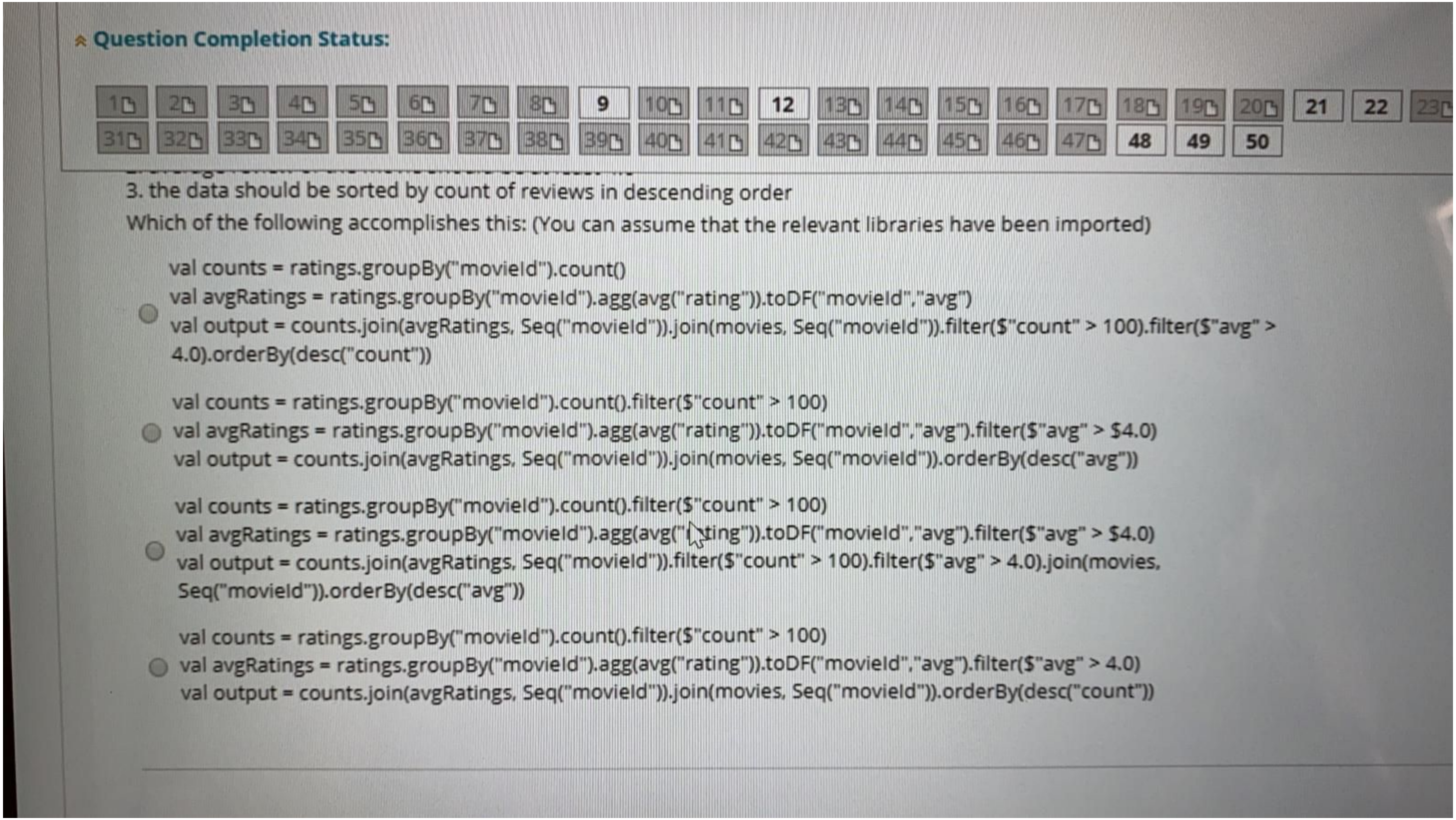
and a **ratings** dataframe as below:

userId	movieId	rating
1	1	4.8
1	3	4.0
...	...	...

You would like to run the *most efficient query* to match following criteria:

1. the movie should have at least 100 ratings
2. average review of the movie should be at least 4.0
3. the data should be sorted by count of reviews in descending order

Which of the following accomplishes this: (You can assume that the relevant libraries have been imported)



3. the data should be sorted by count of reviews in descending order

Which of the following accomplishes this: (You can assume that the relevant libraries have been imported)

○
```
val counts = ratings.groupBy("movieId").count()
val avgRatings = ratings.groupBy("movieId").agg(avg("rating")).toDF("movieId","avg")
val output = counts.join(avgRatings, Seq("movieId")).join(movies, Seq("movieId")).filter($"count" > 100).filter($"avg" >
4.0).orderBy(desc("count"))
```

○
```
val counts = ratings.groupBy("movieId").count().filter($"count" > 100)
val avgRatings = ratings.groupBy("movieId").agg(avg("rating")).toDF("movieId","avg").filter($"avg" > $4.0)
val output = counts.join(avgRatings, Seq("movieId")).join(movies, Seq("movieId")).orderBy(desc("avg"))
```

○
```
val counts = ratings.groupBy("movieId").count().filter($"count" > 100)
val avgRatings = ratings.groupBy("movieId").agg(avg("rating")).toDF("movieId","avg").filter($"avg" > $4.0)
val output = counts.join(avgRatings, Seq("movieId")).filter($"count" > 100).filter($"avg" > 4.0).join(movies,
Seq("movieId")).orderBy(desc("avg"))
```

○
```
val counts = ratings.groupBy("movieId").count().filter($"count" > 100)
val avgRatings = ratings.groupBy("movieId").agg(avg("rating")).toDF("movieId","avg").filter($"avg" > 4.0)
val output = counts.join(avgRatings, Seq("movieId")).join(movies, Seq("movieId")).orderBy(desc("count"))
```