جامعة العربي بن مهيدي – أم البواقي–
كلية العلوم الاقتصادية والعلوم التجارية وعلوم التسيير
قسم العلوم الاقتصادية

السنة الجامعية: 2025 – 2026
المستوى: ثانية ماستر
التخصص: اقتصاد كمي
المدة: ساعة ونصف

يوم: 2026/1/17

الاسم واللقب: .....

امتحان الدورة العادية في مقياس تحليل البيانات الضخمة

السؤال الأول: (09 نقاط)

ضع علامة X على الإجابات الصحيحة فيما يلي (يمكن أن تكون أكثر من إجابة صحيحة):

1. ما الذي يعرّف مجموعة البيانات بأنها "بيانات ضخمة" بشكل أساسي؟:

☐ أ. حجمها وتعقيدها يتجاوزان قدرة أدوات المعالجة والتخزين التقليدية ☐ ج. حجمها يتجاوز دائما واحد تيرابايت

☐ ب. لا يمكن تفسيرها إلا من خلال التحليل البشري المباشر ☐ د. يتم إنشاؤها حصريًا من خلال إنترنت الأشياء

2. . إذا أراد بنك استثماري تحليل بيانات الأسواق المالية العالمية في الوقت الفعلي لاتخاذ قرارات استثمارية فورية، أي نوع من 'معالجة البيانات' سيكون الأكثر فائدة؟:

☐ أ. المعالجة غير المهيكلة Unstructured Processing ☐ ج. المعالجة المتدفقة Stream Processing

☐ ب. المعالجة اليدوية Manual Processing ☐ د. المعالجة الدفعية Batch Processing

3. يمكن لإدارة البيانات الضخمة عند القيام بها بشكل صحيح وفعال أن تحقق مجموعة من الفوائد، من بينها:

☐ أ. تخفيض التكاليف ☐ ج. تحقيق مزايا التنافسية

☐ ب. زيادة الأمان ☐ د. تحليل البيانات

4. إذا تعطل أحد الحواسيب في Cluster، يستطيع برنامج Spark استعادة العمل من البيانات المخزنة مؤقتا بفضل خاصية:

☐ أ. Spark Scalability ☐ ج. Spark Work-restoration

☐ ب. RDD Lineage ☐ د. DataFrame Revision

5. أي من قواعد البيانات التالية تعتبر نوع من أنواع قواعد البيانات غير العلائقية (NoSQL)؟:

☐ أ. قواعد البيانات السحابية ☐ ج. قواعد بيانات الوثائق

☐ ب. قواعد البيانات غير المهيكلة ☐ د. قواعد بيانات المفاتيح والقيمة

6. إذا كان لدينا ملف اسمه input.csv موجود في HDFS مساره هو /bigdata/input.csv ونريد نقله إلى المجلد data داخل القرص C في الجهاز، فإننا نستخدم الأمر:

☐ أ. hdfs dfs -copyToLocal /bigdata/input.csv /c:/data ☐ ج. hdfs dfs -get /bigdata/input.csv /c:/data

☐ ب. hdfs dfs -CopyToLocal /c:/data /bigdata/input.csv ☐ د. لا شيء مما سبق

7. لاستخراج حجم ملف موجودة في HDFS مساره هو /salesdata/sales2025/sales2025Q1.txt فإننا نستخدم الأمر:

☐ أ. hdfs dfs -ls/ salesdata/sales2025/sales2025Q1.txt ☐ ج. hadoop fs -ls /salesdata/sales2025

☐ ب. hdfs dfs -ls /salesdata/sales2025 ☐ د. لا شيء مما سبق

8. إذا أردنا ادخال محتوى في ملف موجود في HDFS حيث مساره هو /data/student.csv فإننا نستخدم الأمر:

☐ أ. hdfs dfs -appandToFile - /data/student.csv ☐ ج. hdfs dfs -appandToFile -/data/student.csv

☐ ب. hdfs dfs -appendToFile /data/student.csv ☐ د. hdfs dfs -appendToFile - /data/student.csv

9. إذا أردنا حذف directory اسمه loans موجود في HDFS مساره هو /branchA/loans فإننا نستخدم الأمر:

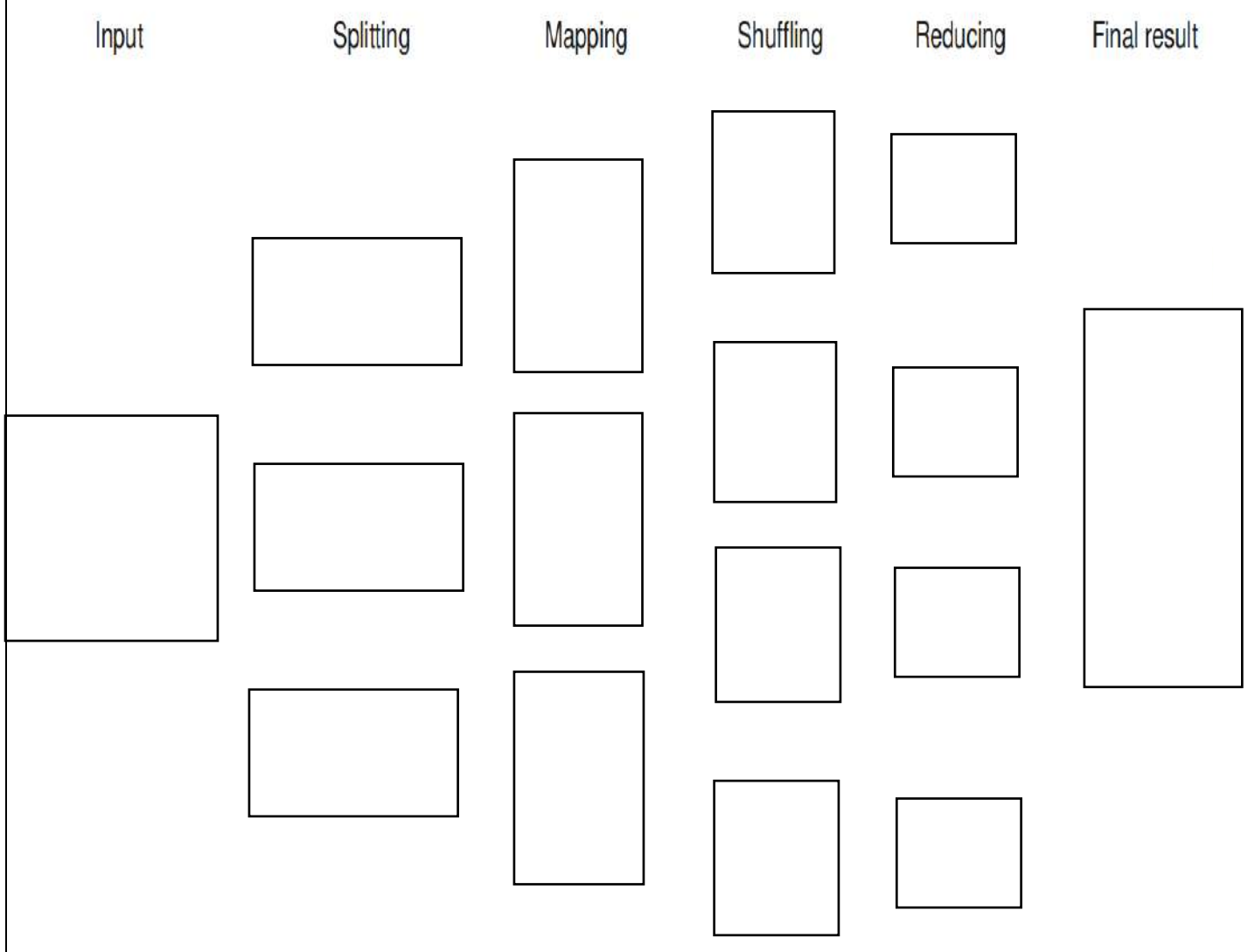
☐ أ. hdfs dfs -rm /branchA/loans ☐ ج. hadoop fs -rm -r /branchA/laens

☐ ب. hdfs dfs -rm -r /branchA/loans ☐ د. hdfs dfs -rm /branchA/leans

السؤال الثاني: (04 نقاط)

وضح من خلال المخطط البياني المرفق (مع ضرورة استخدام أسهم الربط)، آلية عمل MapReduce في حساب عدد كل كلمة موجودة في ملف يحتوي على البيانات التالية:

yarn.hdfs.sql
sql.yarn.sql
rdd.yarn.sql



السؤال الثالث: (07 نقاط)

لدينا ملف اسمه creditrisk.txt موجود في HDFS حيث مساره هو /creditdata/creditrisk.txt يحتوي على بيانات تصنيف مستوى المخاطر لطلبات القروض في أحد البنوك الكبرى، البيانات هي عبارة عن كلمات مفصول بينها بفاصلة منقوطة ";". نريد تنفيذ MapReduce من خلال pyspark لحساب عدد تكرار كلمات التصنيف للثلث الأول من البيانات وإظهار أكثر 10 كلمات تصنيف تكرارا، وكذلك في نفس الكود تحويل نتائج MapReduce إلى جدول (Data Frame) باستخدام مكتبة pandas، بحيث يتم تسمية عمود التصنيفات بـ Rating وعمود التكرارات بـ Number، وكذا التمثيل البياني لجدول نتائج MapReduce من خلال الأعمدة البيانية (bar) أيضا باستخدام مكتبة pandas. وكان لدينا الكود التالي:

```
from pyspark import SparkContext
import pandas as pd
sc = SparkContext(appName="exam")
rdd = sc.textFile("D:/data/creditdata/creditrisk.txt")
words = rdd.flatMap(lambda line: line.split())
third_words = sc.parallelize(words.take(words.count()/2))
pairs = words.map(lambda word: (word, 1))
reducer = pairs.reduceByKey(lambda a, b: a + b)
sort = reducer.sortBy(lambda x: x[1], ascending = True)
for key, value in sort.collect():
    print(key, value)
df = pd.dataframe(reducer.collect(), columns = ["Rating", "Number"])
df.plot.bar(x="Rating", y="Number")
sc.stop()
```

1. يحتوي هذا الكود على مجموعة من الأخطاء، ضع خط تحت الأخطاء الموجودة، ثم قم بإعادة كتابة الكود مع تصحيح كل الأخطاء الموجودة.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

2. إذا أردنا وضع نتائج MapReduce في المسار التالي /creditdata/creditriskresults في الـ HDFS من خلال الكود السابق، أكتب السطر البرمجي الذي يجب إضافته للقيام بذلك.

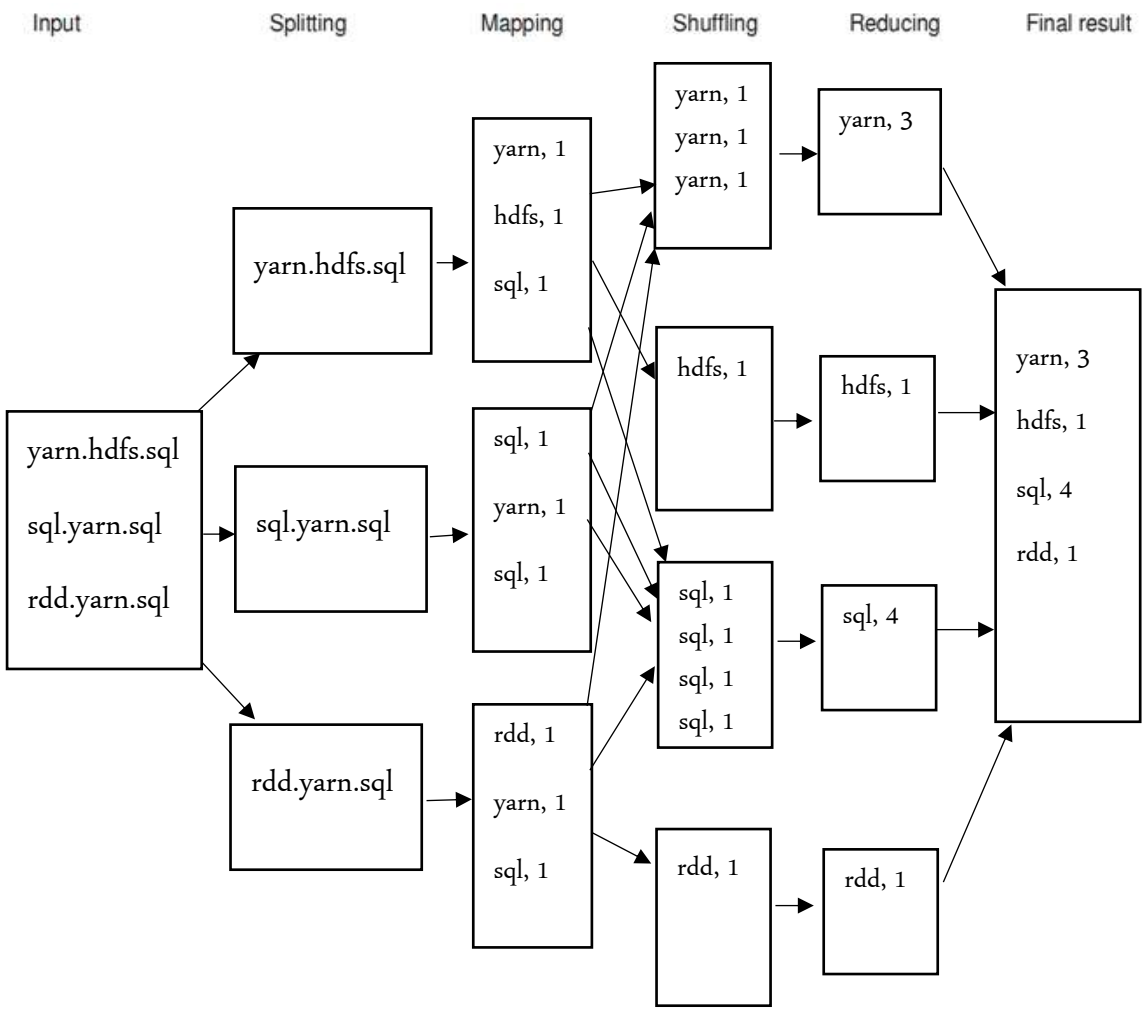
.....

بالتوفيق

الإجابة النموذجية لامتحان الدورة العادية في مقياس تحليل البيانات الضخمة

السؤال الاول		العلامة
	الإجابات الصحيحة:	
	1: أ	1
	2: ج	1
	3: أ ب ج	1
	4: ب	1
	5: ج د	1
	6: د	1
	7: ج ب	1
	8: د	1
	9: ب	1
المجموع		09

السؤال الثاني		النقاط
	التوضيح من خلال المخطط البياني المرفق آلية عمل MapReduce في حساب عدد كل كلمة موجودة في الملف:	4
المجموع		04



النقاط	السؤال الثالث	
2	``` from pyspark import SparkContext import pandas as pd sc = SparkContext(appName="exam") rdd = sc.textFile("D:/data/creditdata/creditrisk.txt") words = rdd.flatMap(lambda line: line.split()) third_words = sc.parallelize(words.take(words.count()/2)) pairs = words.map(lambda word: (word, 1)) reducer = pairs.reduceByKey(lambda a, b: a + b) sort = reducer.sortBy(lambda x: x[1], ascending = True) for key, value in sort.collect(): print(key, value) df = pd.dataframe(reducer.collect(), columns = ["Rating", "Number"]) df.plot.bar(x="Rating", y="Number") sc.stop() ```	1
4	**إعادة كتابة الكود مع تصحيح كل الأخطاء الموجودة:** from pyspark import SparkContext import pandas as pd sc = SparkContext(appName="exam") rdd = sc.textFile("hdfs://localhost:9000/creditdata/creditrisk.txt") words = rdd.flatMap(lambda line: line.split(";")) third_words = sc.parallelize(words.take(words.count()//3)) pairs = third_words.map(lambda word: (word, 1)) reducer = pairs.reduceByKey(lambda a, b: a + b) sort = reducer.sortBy(lambda x: x[1], ascending = False) for key, value in sort.take(10): print(key, value) df = pd.DataFrame(reducer.collect(), columns = ["Rating", "Number"]) df.plot.bar(x="Rating", y="Number") sc.stop()	
1	**كتابة السطر البرمجي الذي يجب إضافته لوضع نتائج MapReduce في المسار /creditdata/creditriskresults:** reducer.saveAsTextFile("hdfs://localhost:9000/creditdata/creditriskresults")	2
07	المجموع	