NOTE: “Exam version 1 above” refers to the exam of feb 2022

Q1.

This is the same as Q1 in Exam version 1 above, just with real numbers instead of variable names.

Q2.

The first 4 are transformations, the rest actions. In order, the dependencies: narrow, possibly wide because of that True flag, narrow, narrow.

Q3.

This is easy if you paid attention when coding, and did a lot of debugging in the shell (pyspark), at every stage of a program. For example, line 1 creates an RDD where a record has type: tuple of 2 strings; line 8 returns a plain Python list.

Q4.

This is like Q4 (c) in Exam version 1 above.

(a) This is structured data, so the obvious idea would be to make a Bigtable out of it (you'd have to roughly explain how). However, *"this data will be processed once a day"*, which means a sale (data point) doesn't have to be read/written quickly, so doesn't appear to need indexing. So, you can store it with less effort in big files (semistructured data: csv, json), ideally one file per day to help the processing.

(b) The pictures are pretty large: up to a chunk size. You could store each in one file, then build a database which helps to find these files: a Bigtable holding the coordinates, timestamp, and file path. For the record, images in general can also be stored inside the Bigtable, as a stream of bytes inside a cell (these images are a little large for that, but the block size of the Bigtable can be configured). The assignment doesn't say how often and how quickly the images must be located (often and quickly, like in Google Maps?), but this makes sense in any case.

Q5.

You simply need to choose a row key (a unique string or number), and probably two column families (keywords, pages) with very many columns. Many students weren't able to represent a Bigtable in any form (as a table, or as a map).

Q6.

This one's theory. Kafka and Spark Streaming don't really overlap in functionality: they do different things altogether, so a lot can be said here. Spark doesn't store data except until the end of the program (Kafka does provide a form of storage, and an orderly fashion to provide only the relevant part of the data to a consumer application).

Q7.

This you essentially got as the KNN assignment already! You could try to do regression instead of classification (very similar algorithm, except now all "class" names are replaced with numerical "values". For the new point, compute the most likely value, namely: the average of the values of the k nearest points. (It's unlikely that you'll get a task so close to an assignment though.)