Задание для 2 курса

Цель

- Знакомство с базовой концепцией технологии обработки больших данных MapReduce, которая позволяет разбивать данные на небольшие фрагменты, каждый из которых может обрабатываться параллельно и независимо друг от друга;
- Решение реальной практической задачи с применением указанной технологии MapReduce на базе библиотеки pySpark языка Python, построенной на базе фреймворка Appache Spark, упрощающего построение приложений на основе подхода MapReduce.

Описание практического задания

Предлагается поэтапная последовательность действий по следующим шагам:

1. Знакомство с материалом (набор лекции с озвучкой в PowerPoint) и самопроверка по тестовым вопросам.
2. Самостоятельная локальная установка (на личных компьютерах) необходимого набора программного обеспечения для обеспечения возможности написания программ в среде pySpark. Инструкция по установке приведена в конце данного документа.
3. Разработка программы на языке Python с использованием pySpark, преобразующей исходные данные (набор строк, описывающих сделки с разными финансовыми инструментами (акциями/облигациями)), в данные, содержащие описание сделок в формате японских свечей (японскую свечу можно рассматривать как некоторый набор статистик по сделкам, совершенным за определённый момент времени).

Формат входных данных

На вход подаётся .csv-файл (разделители - запятые). Пример файла можно скачать по ссылке: https://m.cs.msu.ru/s/mA4xyL6ytqAEE8f

В первой строке файла указаны названия колонок. Названия следующие (запятая - разделитель отдельных названий):

#SYMBOL,SYSTEM,MOMENT,ID_DEAL,PRICE_DEAL,VOLUME,OPEN_POS,DIRECTION

Пояснения к названиям колонок, которые следует использовать в задаче:

1. #SYMBOL – название финансового инструмента (акции/облигации);
2. MOMENT – время (дата) сделки;
3. PRICE_DEAL – цена сделки.

Далее в файле указываются строки, описывающие происходящие сделки (разделитель значений в строке - запятая).

Пример строки в файле:
SVH1,F,20110111100000080,255223067,30.46000,1,8714,S

Данный файл отсортирован по дате и времени.
Дата и время указываются в формате YYYYMMDDhhmmssfff, где YYYY - год, MM - месяц, DD - день, hh - часы, mm - минуты, ss - секунды, f – миллисекунды.

Параметры программы

Также, в параметрах программы передается (опционально) путь к конфигурационному файлу config.xml. Этот файл может содержать описание следующих параметров (после знака = указаны значения параметров, используемые по умолчанию; после // указан комментарий, поясняющий назначение параметра; каждый параметр может быть опущен в файле конфигурации, при этом будет использовано значение по умолчанию):
candle.width = 300000 // "ширина" свечи в числе миллисекунд;
candle.date.from = 19000101 // первый день периода времени (ГГГГММДД);
candle.date.to = 20200101 // первый день после последнего дня периода (ГГГГММДД);
candle.time.from = 1000 // время (ЧЧММ) начала первой свечи;
candle.time.to = 1800 // время (ЧЧММ) после окончания последней свечи.

Формат выходных данных

Набор файлов. Каждый файл имеет расширение .csv и содержит в имени название инструмента, по которому этот файл построен (название инструмента можно получить из колонки #SYMBOL в исходном наборе данных). Соответственно, для каждого инструмента отдельно строится файл.

Каждый файл представляет собой описание японских свечей для указанного инструмента в формате .csv ***без шапки***(т.е. без первой строки с заголовком).
Данный файл состоит из строк – описаний свечей для соответствующего инструмента (название инструмента содержится в имени файла – см. выше).
Формат строки следующий:
SYMBOL,MOMENT,OPEN,HIGH,LOW,CLOSE

Здесь:
1. ***MOMENT*** – время начала свечи;
2. ***OPEN*** – цена первой сделки за свечу;
3. ***HIGH*** – максимальная цена за свечу;
4. ***LOW*** – минимальная цена за свечу;
5. ***CLOSE*** – цена последней сделки за свечу.

Пример строки в выходном файле (обратите внимание, что ***точность определяется одним знаком после запятой***):
GDH1,20110111100000000,1407.0,1407.0,1379.0,1379.3

Параметры свечей задаются в файле конфигурации (см. выше).

Свечи "начинаются" в моменты времени, кратные "ширине". Отсчет времени для вычисления кратности начинается в 00:00 рассматриваемого дня.

На выходе необходимо получить директорию с файлами. Имена файлов должны содержать SYMBOL в качестве префикса.

Пояснения

1. Свечи необходимо строить от candle.time.from до candle.time.to каждого рассматриваемого дня;
2. Рассматривать переход через сутки не нужно (программа будет тестироваться на свечах, построенных в рамках одного рабочего дня, 10:00 – 18:00, либо меньшего периода времени);
3. Рассматривать случай, когда последняя свеча не помещается целиком в рассматриваемый промежуток времени, не нужно (считаем, что в рассматриваемый период времени укладывается целое число свеч, и в момент времени candle.time.to должна начаться новая свеча, которую мы не рассматриваем);
4. Рассматривать случай, когда candle.time.from не кратно размеру свечи, не нужно. Считаем, что candle.time.from всегда кратно размеру свечи.
5. В ситуациях, когда рассматриваются записи, в которых совпадают названия инструментов и моменты времени, но цены различны, для разрешения неоднозначности необходимо дополнительно рассматривать поле ID_DEAL. При одинаковых моментах времени для цены открытия (OPEN) выбирается цены с ***наименьшим*** ID_DEAL, для цены закрытия (CLOSE) - с ***наибольшим*** ID_DEAL.

Начало работы с PySpark

- Для программирования примера используется Python 3.10. Для его установки на свой компьютер потребуется предварительно установить Java.
- Используется Spark API для языка Python. API оформлено в виде библиотеки pySpark. Про неё можно почитать по ссылкам:
  https://spark.apache.org/docs/latest/api/python/index.html
  https://spark.apache.org/docs/latest/api/python/getting_started/index.html
- Конкретная версию pySpark может варьироваться, предлагается текущую на данный момент версию 3.5.0. Её установить можно по инструкции, описанной по следующей ссылке:
  https://spark.apache.org/docs/latest/api/python/getting_started/install.html
  Рекомендуется использовать автоматическую установку через команду:
  pip install pyspark