Ревью на работу Анастасии Сизовой. Подготовил: Абдуллаев Сахиб

Понятность работы:

Благодаря подробному описанию к работе становится понятно какой шаг и для чего именно производится. Приводятся комментарии по необходимым изменениям. Однако не совсем понятно зачем в данной части использовалась связь с питоном, если можно было прописать код полностью в sql.

Содержание отчета:

В отчете содержатся данные по использованному программному обеспечению, что позволяет сразу сориентироваться нужно ли что-либо доставлять. Также расписана последовательность действий: схематично и ниже более подробно (это удобно для проверки, не приходится обращаться к условиям домашнего задания).

Полнота работы:

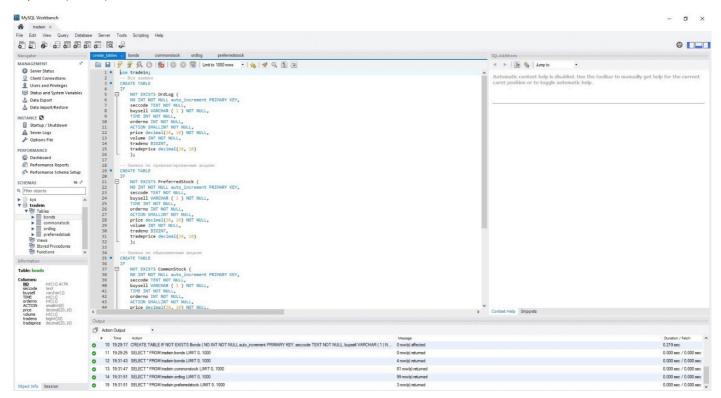
Все поставленные в задании вопросы выполнены (за исключением того, что не протестирована самостоятельно придуманная ошибка): База данных создана, таблицы по привилегированным и обыкновенным акциям и бондам заполнены с помощью классификатора, загруженного с сайта Московской Биржи, найден тикер с наибольшим количеством исполненных рыночных заявок — GAZP.

Результаты прогонки:

Прогонка предоставленных скриптов произошла успешно, без возникновения проблем. Результаты прогонки представлены ниже.

Оценивание работы:

Учитывая, перечисленные выше положительные стороны и недостатки работы, рекомендую поставить оценку хорошо (4 из 5).



```
File Edit Shell Debug Options Window Help

Python 3.7.0 (1v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD6 1)] on vin32

Type "copyright", "credits" or "license()" for more information.

>>>

Thkep, no kotopomy было сделано больше всего сделок: GAZP, в количестве: 19

>>>

Lice Col.4
```

```
tikers.py - C:\Users\Sahib\Desktop\tikers.py (3.7.0)
                                                                                                                                         X
File Edit Format Run Options Window Help
# Соединяем Питона с БД
import mysql.connector
import decimal
‡ Коннектимся к нашей БД
conn = mysql.connector.connect(
    user="root",
    password="891109",
    host="localhost"
curs = conn.cursor()
# Использование БД, созданной в MySQL
curs.execute("""USE tradein;""")
# Открываем файл OrderLog для импорта данных в БД
data = open('TradeLog20160114.txt').read()
# Формируем список из первых 100 записей (для экономии времени и места :) )
lines = data.split('\n')[1:100]
# Заполняем таблицу OrdLog дамными из файла OrderLog20160114.txt и делаем проверку
# повторного запуска скрипта без занесения старых записей в БД
for i, elem in enumerate(lines):
    try:
        t = "
        for field in elem.split(',')[1:]:
            try:
                float (field)
            except Exception:
# Eсли поле пустое, то ставим 0
if field == '':
                    t += '0, '
                    t += f"'{field}', "
           else:
        t += f'{field}, '
request = f"""INSERT INTO ordlog(seccode, buysell, time, orderno, action, price, volume, tradeno, tradeprice) VALUES ({t[:-2]});"'
        curs.execute (request)
    except Exception as e:
        pass
conn.commit()
‡ Делаем запрос на получение всех данных из таблицы OrdLog
curs.execute("""SELECT * from ordlog;""")
# Получаем результат сделанного запроса
rows = curs.fetchall()
‡ ----Разбиваем таблицу OrdLog на 3 новых-----
‡ Открываем файл с тикерами
tickers_text = open('ListingSecurityList_new.csv', encoding = "cp1251").read()
tickers = {}
for elem in tickers_text.split('\n')[:-1]:
t = elem enlit/
                                                                                                                                        Ln: 19 Col: 31
```