

Тестовое задание JAVA-разработчика

Исходные данные представлены выгрузками Московской биржи (описание API: <https://fs.moex.com/files/6523>):

- информация о ценных бумагах (securities_*.xml);
- история торгов за произвольную дату (history_*.xml).

Поле для связи объектов: SECID. В приложении допускается существование ценной бумаги без истории, истории без описания бумаги - нет.

Необходимо реализовать сервис, покрывающий задачи:

1. Импорт объектов из приложенных файлов (состав сохраняемых данных определить самостоятельно);
2. Предоставить API для CRUD операций по объектам;
3. При ручном сохранении ценной бумаги проводить валидацию передаваемых данных в поле name - только кириллица, цифры и пробел;
4. Отдельным методом реализовать вывод таблицы с данными из тегов:
 - secid
 - regnumber
 - name
 - emitent_title
 - tradedate
 - numtrades
 - open
 - close

Предусмотреть возможность задать параметрами столбцы для сортировки и фильтрацию по полям emitent_title и tradedate;

5. Выбранную архитектуру решения и инструкцию по запуску приложения описать в Readme файле.

Дополнительные задачи «со звездочкой» в порядке уменьшения их значимости при оценке:

1. Реализовать хранение и работу с данными в БД;
2. Реализовать MVC приложение, позволяющее через интерфейс импортировать файлы и работать с таблицей п.4 и CRUD операциями. Не

стоит акцентировать внимание на ux/ui, нам интересно посмотреть на ваш навык работы с html/js;

3. Доработать импорт данных: при импорте истории по отсутствующей ценной бумаге выполнять rest запрос к API биржи (роут метода http://iss.moex.com/iss/securities.xml?q=SEARCH_STRING). Обратить внимание на оптимизацию обращений;
4. Реализовать фоновое выполнение парсинга данных по расписанию.

Исходный код необходимо выложить на github (или аналогичном сервисе) или прислать архивом с комментариями.