**Министерство науки и высшего образования Российской Федерации**

**Федеральное государственное автономное**

**образовательное учреждение**

**высшего образования**

**«КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

ИНСТИТУТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И

ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

КАФЕДРА ТЕХНОЛОГИЙ ПРОГРАММИРОВАНИЯ

Направление: 01.03.04 – «Прикладная математика»

Курсовой проект по лабораторному практикуму ИКТ

**Прогнозирование фондового рынка**

**Работа завершена:**

“\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Р.Р.Степанов)

**Работа допущена к защите:**

Матренина Ольга Михайловна

к.н.

“\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (О.М.Матренина)

**Казань-2019**

**Содержание**

Содержание………………………………………………………….2

Введение……………………………………………………………..3

Приложение на JavaFX……………………………………………...4

Работа с базой данных……………………………………………....7

Скачивание данных с finam.ru………………………………………10

Заключение…………………………………………………………...12

Список литературы и сайтов………………………………………...13

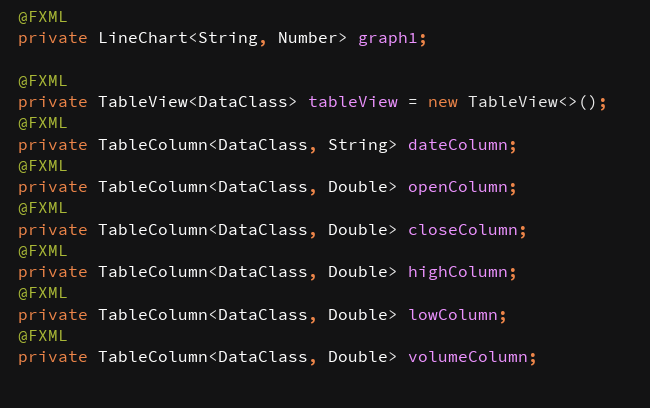
**Введение**

Так или иначе, фондовая биржа постепенно проникает в жизнь обычных людей. Несколько десятков лет назад, все, что было связано с биржей, считалось уделом избранным, чем- то сложным и запутанным. Согласно данным Московской биржи и сайта rbc.ru количество уникальных брокерских и инвестиционных счетов растет достаточно быстро, а значит возникает потребность в инвестиционных консультантах, либо приложениях-помощников. Я создал приложение, которое на основании данных за прошедшее время формирует графики, которые с определенной точностью могут спрогнозировать дальнейшие движения цен акций, облигаций, валютных пар. В приложении используются открытые данные, которые загружаются с finam.ru

**Глава 1. Приложение на JavaFX**

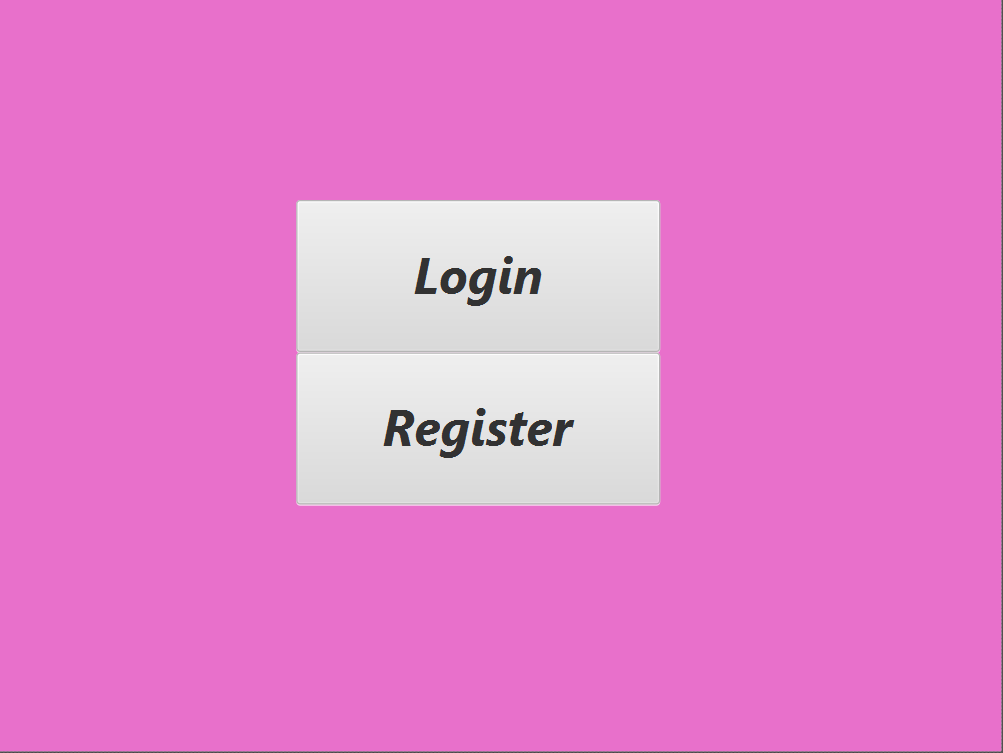
JavaFX – огромный набор инструментов для визуализации приложений, построения графиков, таблиц, кроссплатформенных приложений. Конкурентным преимуществом приложения на JavaFX является кроссплатформенность, оно будет работать везде, где установлена исполняемая среда Java (JRE). Изначально, JavaFX являлся частью выпуска Java8, в последующих версиях он был удален и перенесен в отдельный модуль в Java11. FX позволяет работать с мультимедией, двухмерной и трехмерной графикой. Практически все элементы FX можно «украсить» с помощью CSS, которые позволяет прикреплять стили.

Приложение на JavaFX состоит из двух компонентов, описание визуализации в формате fxml и Controller’a, в котором будет прописывается действия каждого элемента, его заполнение и т.д. Для связи этих двух элементов в fxml нужно прописать его контроллер. Каждый элемент из fxml декларируются с помощью аннотации @FXML (рис.1)



* 1. **Создание окна авторизации.**

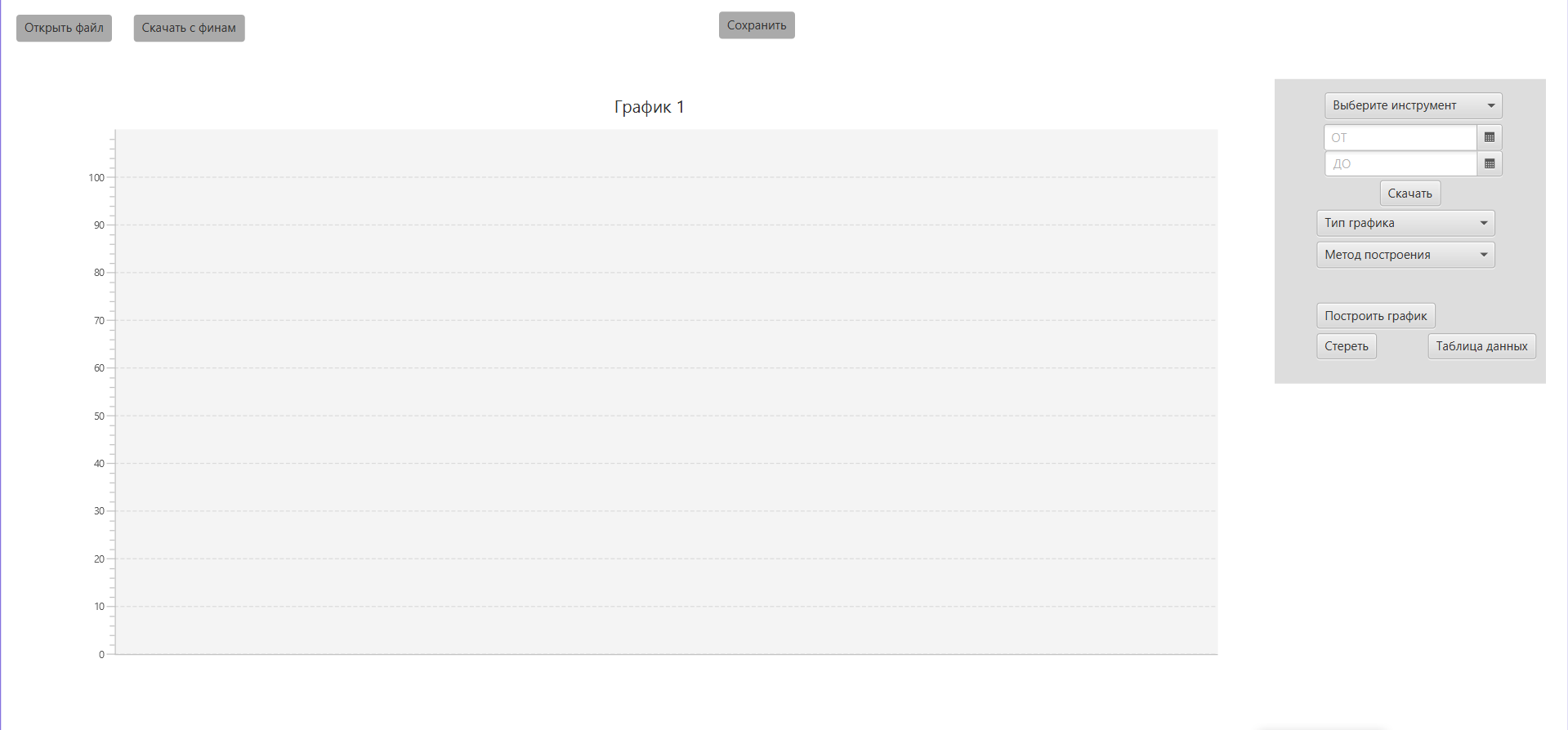
Окно регистрации (рис.2) я сделал максимально простым.

 Рис.2

За основу был взят AnchorPane. AnchorPane позволяет привязывать объекты на сцене, что позволяет менять размеры окна и созранять внутренние пропорции расположения объектов внутри окна. Кнопки регистрации и входа было созданы как обычные Button

**1.2 Создание основного рабочего окна (рис.3)**

За основу был так же взят AnchorPane для удобного масштабирования.  
График был создан с помощью LineChart.

Рис.3

Для заполнения графика используется метод Seris, который представляет собой набор данных по оси Х и У, где соответствующему значению x, сопоставляет значение y.

Значение х может быть как числовым, так и текстовым, например, датами.

Для этого нужно определить ось Х как CategoryAxis.

switch (comboBuildType.getValue()){  
 case "linear":{  
 series1.getData().add(new XYChart.Data<>(new SimpleDateFormat("yyyymmdd").parse(array1[**0**]).toString()**,** MathMethods.*linearFun*(-**1**)))**;** series1.getData().add(new XYChart.Data<>(new SimpleDateFormat("yyyymmdd").parse(array1[array1.length-**1**]).toString()**,** MathMethods.*linearFun*(array1.length)))**;** series1.setName(DataManager.*getInstrument*() + " " + comboType.getValue()+" "+comboBuildType.getValue())**;** graph1.getData().add(series1)**;** break**;** }

С помощью конструкции switch…case, в зависимости от значения ComboBox(еще одного модуля FX который является всплывающим списком),

реализует алгоритм заполнения. Непосредственно заполнение значений осуществляется через seris.getData().add(). График гибко настраивается, для этого вызываются методы из класса графика: отображение легенды, настройка цвета графика, подпись к каждому графику и т.д.

Для выбора даты я использовал встроенный компонент DatePicker, который уже визуализирует календарь и возвращает дату с помощью getValue().

Так же в моем проекте я использовал визуализацию данных из csv файла, создав таблицу TableView<DataClass>, где DataClass - это пользовательский класс, который хранит в себе данные о каждом дне торгов

String date;

double open;

double close;

double high;

double low;

double volume;

Так происходит инициализация столбцов графика, с указанием типа данных внутри ячеек. setItems(daysData) непосредственно заполняет таблицу значениями из daysData, который является ArrayList’ом. При инициализации столбцов графика, после указания типа ячеек, указывается поле из ArrayList из которого берутся значения.

dateColumn.setCellValueFactory(new PropertyValueFactory<DataClass, String>("date"));

openColumn.setCellValueFactory(new PropertyValueFactory<DataClass, Double>("open"));

closeColumn.setCellValueFactory(new PropertyValueFactory<DataClass, Double>("close"));

highColumn.setCellValueFactory(new PropertyValueFactory<DataClass, Double>("high"));

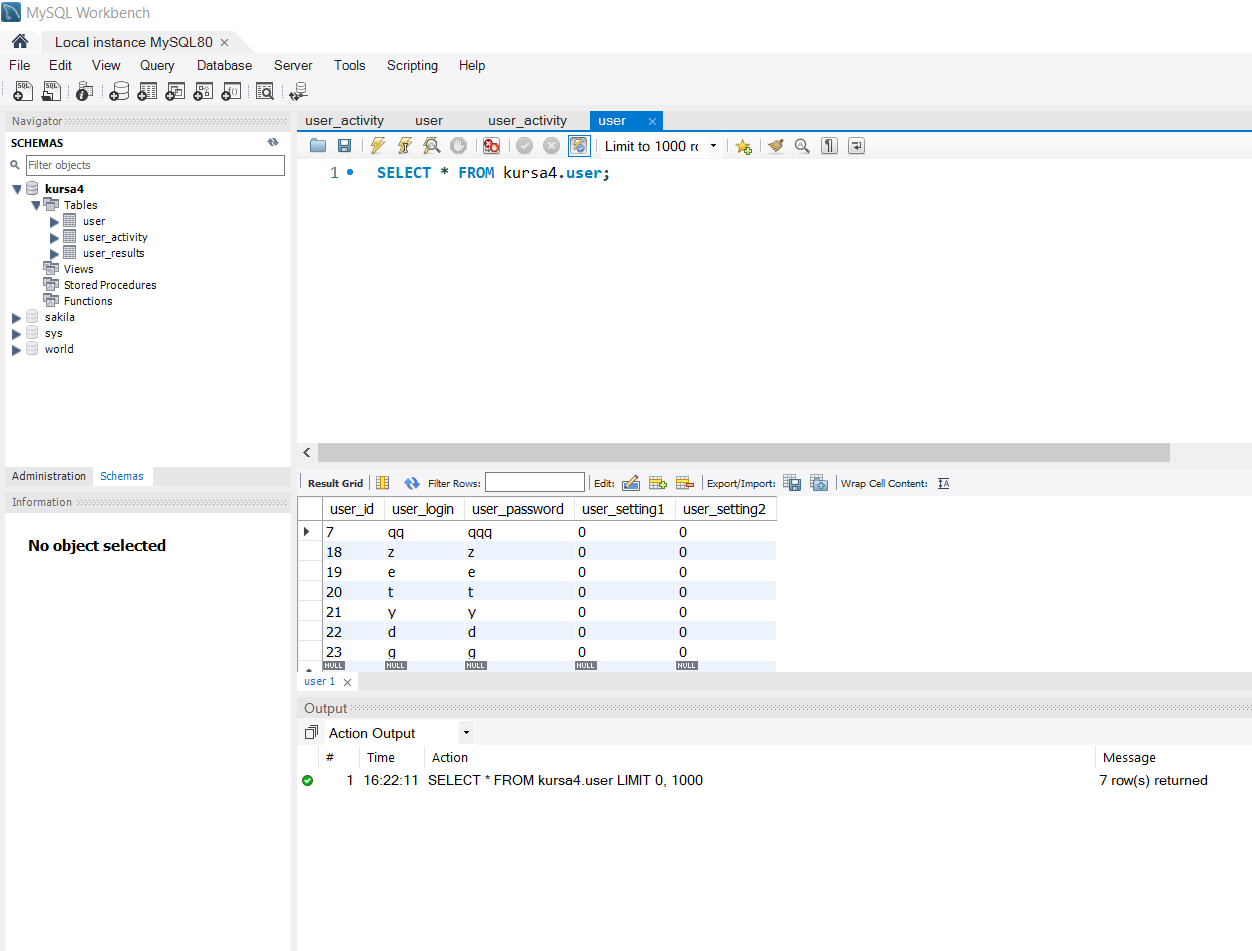
lowColumn.setCellValueFactory(new PropertyValueFactory<DataClass, Double>("low"));

volumeColumn.setCellValueFactory(new PropertyValueFactory<DataClass, Double>("volume"));

**Глава 2. Работа с базой данных**

**2.1 Начало работы**

Для работы над проектом я выбрал одну из самых популярных и поддерживаемых БД, а именно MySql. Одним из ее преимуществ является наличие графического интерфейса для удобства ее создания, администрирования, редактирования и т.д. (рис.4)



Для взаимодействия с базой данных я сделал отдельный класс DBManager. Для работы с БД из Java необходимо скачать класс Driver, который непосредственно связывает java с компонентами БД. Работа с MySql начинается с подключения

conn = DriverManager.getConnection("jdbc:mysql://localhost/?useUnicode=true&useJDBCCompliantTimezoneShift=true&useLegacyDatetimeCode=false&serverTimezone=UTC", <username>, <password>);

Первый параметр-адрес нахождения БД, второй и третий параметры, соответственно, логин и пароль. Далее при любой работе с БД я передаю только объект Connection.

**2.2 Формирование запросов и получение ответов**

String sql = "INSERT INTO kursa4.user (user\_login, user\_password) Values (?, ?)";

PreparedStatement statement = getConn().prepareStatement(sql);

statement.setString(1, user.getLogin());

statement.setString(2, user.getPassword());

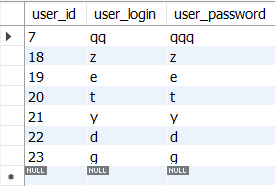
statement.executeUpdate();

Создается строка запроса sql на языке БД MySql. Для заполнения переменных можно использовать либо PreparedStatement, либо «склеивать» строку и вставлять переменные из среды java. Затем строка запроса посылается для исполнения в БД методом executeUpdate.

Для получения ответа из БД используется executeQuery(), который является объектом класса ResultSet

**2.3 Регистрация и вход**

Для регистрации и входа была создана таблица user с полями



user\_id - ненулевое поле с автоматической генерацией порядкового номера

user\_login – ненулевое, уникальное поле

user\_password – так же ненулевое поле, не уникальное, т.к пароли у пользователей могут совпадать

В проекте не требовалось создавать защищенную систему, поэтому все данные передаются в не зашифрованном виде. Условием успешной регистрации является только отсутствие введенного логина в БД. Условием успешного входа является получения не пустого ответа на запрос

String sql = "select \* " +

"FROM kursa4.user " +

"where user\_login=? and user\_password=?";

PreparedStatement statement = getConn().prepareStatement(sql);

statement.setString(1, user.getLogin());

statement.setString(2, user.getPassword());

ResultSet rs = statement.executeQuery();

//В случае ошибки

if (rs.next()) {

System.out.println("ДОБРО ПОЖАЛОВАТЬ");

conn.close();

return true;

}

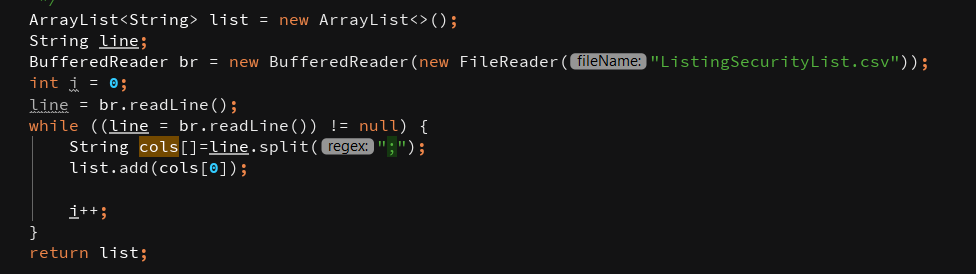
При нахождении двух совпадающих полей, одно из которых уникальное, ответ всегда будет не пуст, а значит такая запись есть, пользователь успешно авторизируется, иначе, ответ будет пустой т.е. либо пользователь не зарегистрован, либо пароль неверный.

**2.4 Логирование**

При успешном входе, из БД получаю уникальный id пользователя и отправляю запрос в связанную таблицу о том, что пользователь с данным id успешно авторизировался. При корректном закрытии программы, в ту же таблицу отправляю результаты работы программы и время выхода пользователя

**Глава 3. Скачивание данных с finam.ru**

Для работы с Web я создал отдельный класс DownloadFromWeb.

Сначала формируется ссылка по заданным параметрам. Сначала получаем название инструмента из списка Московской биржи 

Затем считывается желаемый период загрузки данных из DatePicker из пакета JavaFX.

В классе getUrl формируется ссылка для скачивания. Основную проблему составляет параметр «em» это внутренний идентификатор инструмента внутри базы finam.ru, этих данных нету в открытом доступе, для их получения пришлось извлекать их для каждого инструмента.

Второй сложностью было-преодоление ошибки 403, которая являлась «заглушкой» для программного получения данных. Эту проблему решат всего одна сточка кода

conn.setRequestProperty("User-Agent", "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.11 (KHTML, like Gecko) Chrome/23.0.1271.95 Safari/537.11");

Java устанавливает атрибуты обычного браузера и кажется, что мы обычный пользовтель сайта.

Ниже приведен код с комментариями для непосредственной загрузке файла их ссылки

// Т.к. запрос у нас GET, то сразу принимаем входящие данные.

// Т.к. файл у нас бинарный, открываем ReadableByteChannel и создаем файл

ReadableByteChannel rbc = Channels.newChannel(conn.getInputStream());

FileOutputStream fos = new FileOutputStream("input.csv");

// Перенаправляем данные из ReadableByteChannel прямо канал файла.

long filePosition = 0;

long transferedBytes = fos.getChannel().transferFrom(rbc, filePosition, Long.MAX\_VALUE);

while (transferedBytes == Long.MAX\_VALUE) {

filePosition += transferedBytes;

transferedBytes = fos.getChannel().transferFrom(rbc, filePosition, Long.MAX\_VALUE);

}

**Заключение**

В ходе разработки приложения была создана программа для автоматического чтения исходных статистических данных с сайта finam.ru.

Вывод полученных данных на график и в таблицу, для визуального представления и анализа. Так же были написаны методы для использования статистических функций, для прогнозирования тренда выбранных инструментов.

При создании приложения был получен большой опыт использования JavaFX, базы данных MySql, практика в использовании объектно- ориентированного программирования, взаимодействие с web-объектами.

В программе реализованы следующие возможности

* Загрузка данных с сайта finam.ru и с компьютера пользователя
* Создание учетных записей пользовтеля
* Логирование действий пользователей
* Набор статистических функций для прогнозирования цены выбрнных инструментов

Программа будет полезна для математической консультации начинающих инвесторов, для минимизации рисков со сделками ценных бумаг, валюты.

Программа обладает широкими возможностями модернизации и расширения функционала, оптимизацией сложных вычислений и улучения пользовательского интерфейса

**Список литературы и источников**

1. Брюс Эккель «Философия Java» (4-е полное издание) [Печатное издание] / 2009. – 231 с. Санкт- Петербург.   
2. Васильев А. Н. - Java. Объектно-ориентированное программирование. [Печатное издание] / 2011. – 400с. Санкт-Петербург.   
3. Математическая статистика. Горяинов В.Б., Павлов И.В., Цветкова Г.М. и др. [Печатное издание] / Изд-во МГТУ им. Н.Э. Баумана, 2001. - 424 с.

4. <https://habr.com/ru/post/332700/>

5. [ru.stackoverflow.com](https://ru.stackoverflow.com/)

6. <https://habr.com/ru/post/130057/>

7. <https://devcolibri.com/javafx-введение-в-charts/>

8. <https://tproger.ru/translations/java-jdbc-example/>

9. <https://docs.oracle.com/javafx/2/charts/line-chart.htm>

10. https://www.baeldung.com/java-download-file