**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«Национальный исследовательский ядерный университет «МИФИ»

**Обнинский институт атомной энергетики –**

филиал федерального государственного автономного образовательного учреждения высшего

образования "Национальный исследовательский ядерный университет «МИФИ»

(ИАТЭ НИЯУ МИФИ)

Отделение Интеллектуальные кибернетические системы

**Лабораторная работа № 3**

**Средства визуализации данных в языках программирования**

Выполнил:

студент гр. ИС-М18 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Герасимов В.В.

Принял:

профессор ОИКС \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Сальников Н.Л.

Обнинск, 2018

**Цель работы**: выбрать интересующий язык программирования и изучить средства визуализации данных, реализованных в данном языке с помощью подключаемых библиотек. Выбрано:Java **[1]** и JFreeChart **[2]**.

**Выполнение работы**:

Для выполнения работы необходимо найти и подключить библиотеку к Java проекту. Будем использовать Maven **[3]** для автоматического поиска зависимостей проекта и удобства. В *pom.xml* добавим следующие строки для подключения библиотеки:

<**dependency**>  
 <**groupId**>org.jfree</**groupId**>  
 <**artifactId**>jfreechart</**artifactId**>  
 <**version**>1.5.0</**version**>  
</**dependency**>

Библиотека успешно подключена к проекту, и их классы теперь можно использовать у себя.

В Java необходимо создать класс, наследующийся от J*Frame*, чтобы создать пользовательский интерфейс, где и будем визуализировать информацию. В *JFreeChart* уже этот класс существует, будем его использовать – *ApplicationFrame.*

Чтобы визуализировать что-то, нам нужны эти данные. В *JFreeChart* оперируют специальным интерфейсом – *Dataset*. От него наследуются и другие интерфейсы для данных, например, *CategoryDataset,* который используется в графиках типа *Bar* и *Line*; *PieDataset* для графика типа *Pie*.

Чтобы построить график типа *Bar* необходимо использовать статичный метод ChartFactory.*createBarChart(…)*, в который передается набор данных *CategoryDataset,* заголовок и другие параметры.

Чтобы построить график типа *Pie* необходимо использовать статичный метод ChartFactory.*createPieChart(…)*, в который передается набор данных *PieDataset,* заголовок и другие параметры.

Чтобы построить график типа *Line* необходимо использовать статичный метод ChartFactory.*createLineChart(…)*, в который передается набор данных *CategoryDataset,* заголовок, названия осей Х и Y, другие параметры.

Чтобы построить график типа 3D*Pie* необходимо использовать статичный метод ChartFactory.*createPieChart3D(…)*, в который передается набор данных *PieDataset,* заголовок и другие параметры. В отличие от графика типа *Pie* этот график будет выглядеть покрасивее за счет своей объемности.

Чтобы построить график типа *MultiplyPieChart* необходимо использовать статичный метод ChartFactory. *createMultiplePieChart3D(…)*, в который передается набор данных *PieDataset,* заголовок и другие параметры. В этом случае наборы данных необходимо разделить на кластеры по какому-либо признаку, например по месяцам: данные на Январь 2018, данные на Июнь 2018.

Также ко всем объектам графика необходимо прикреплять панель, где эти сами графики будут располагаться ChartPanel.

Для непосредственного показа графика необходимо создать объекты, наследуемые от *ApplicationFrame* (те самые классы, которые описывал сверху), далее в метод setVisible(..) передать true для показа фрейма. Можно еще выполнить метод pack(), чтобы развернуть фрейм.

По итогу создадим pull request в главный репозитории **[4]** и ждем решения владельца репозитория **[4]** о принятии / непринятии изменении.

**Вывод**: в результате выполнения работы была изучена библиотека JFreeChart **[2]** для визуализации данных в Java **[1].** Были продемонстрированы примеры построения график всех основных типов, реализованных в библиотеке: *Bar*, *Line*, *Pie*, *MultiPie*.

**Список используемых источников**

1. Java. [Электронный ресурс]. URL: https://java.com/ru/ [Доступ от 29.11.2018].
2. JFreeChart. [Электронный ресурс]. URL: http://www.jfree.org/jfreechart/ [Доступ от 29.11.2018].
3. Maven. [Электронный ресурс]. URL: https://maven.apache.org [Доступ от 29.11.2018].
4. Рабочий проект. [Электронный ресурс]. URL: <https://github.com/mikhail-turicyn/IS-M18> [Доступ от 29.11.2018]