*Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования*

***«Московский государственный технический университет имени Н.Э. Баумана»   
(МГТУ им. Н.Э. Баумана)***

***Кафедра ИУ3, 3 курс, 6 семестр.***

**Отчёт**

**по лабораторной работе №2**

# “[Создание фигуры диаграммы на JavaFX](http://www.agentlab.ru/confluence/pages/viewpage.action?pageId=54001678)

# [с использованием Xtend2](http://www.agentlab.ru/confluence/pages/viewpage.action?pageId=54001678)”

**по курсу**

**“Разработка программного обеспечения”**

Выполнила:

Емельянова А.А.

Группа:

ИУ3-61

Проверил:

Иванов А.М.

**Москва, 2014**

# Цель работы

Цель работы — формирование у студентов ряда навыков:

* Использование передовых технологий разработки GUI толстых клиентов
* Использование DSL языков на примере Xtend2
* Основы работы в системе контроля версий Git

# Задание

Создать Eclipse-проект с векторной фигуркой  в соответствии со своим вариантом задания и требованиями, добавить его в репозитарий системы контроля версий.

Вариант 9:



# Настройка инструментов

Предварительные требования:

* Eclipse с установленным e(fx)clipse с предыдущей лабораторной работы

Последовательность настройки:

1. Создать аккаунт на Github и выслать логин на [amivanoff@gmail.com](mailto:amivanoff@gmail.com) для предоставления прав на запись

2. Настроить подключение Eclipse к Github

* Использование руководство [Настройка EclipseIDE для доступа к репозитариюGitHubпопротоколу HTTP](http://www.agentlab.ru/confluence/pages/viewpage.action?pageId=41058472)
  + Указать другой адрес репозитария, скопировав его со страницы <https://github.com/agentlab/jfxeditor>

3. Установить в Eclipse поддержку языка Xtend2

* См. <https://www.eclipse.org/xtend/download.html>
  + тамвыбрать "Latest Release"

Для создания фигуры по заданию была использован метод с добавлением структурыGridPane иHBox, внутри которого была реализована группа с добавлением в нее полилинии и текста. Описаниеметодов создания формы находится в файле ClassFigure.xtend. Описание стилей для компонентов формы находится в файле ClassFigure.css.

Листинг:

**package ru.agentlab.jfxed.figures;**

**import ru.agentlab.jfxed.figures.clazz.ClassFigure;**

**import ru.agentlab.jfxed.figures.clazz.ClassKok;**

**import javafx.application.Application;**

**import javafx.event.EventHandler;**

**import javafx.scene.Node;**

**import javafx.scene.Scene;**

**import javafx.scene.input.MouseEvent;**

**import javafx.scene.layout.AnchorPane;**

**import javafx.scene.layout.VBox;**

**import javafx.stage.Stage;**

**public class BootFigures extends Application {**

**ClassFigure sourceToMove;**

**double sceneStartX;**

**double sceneStartY;**

**double figureStartTranslateX;**

**double figureStartTranslateY;**

**public static void main(String[] args) {**

**launch(args);**

**}**

**@Override**

**public void start(Stage primaryStage) throws Exception {**

**primaryStage.setTitle("Hello World!");**

**AnchorPane root = new AnchorPane();**

**final ClassFigure source = new ClassFigure();**

**final ClassKok source1 = new ClassKok();**

**root.getChildren().add(source.getRoot());**

**root.getChildren().add(new VBox());**

**root.getChildren().add(source1.getRoot());**

**root.getChildren().add(new VBox());**

**Scene scene = new Scene(root, 800, 600);**

**//scene.getStylesheets().addAll(ClassFigure.class.getResource("ClassFigure.css").toExternalForm());**

**primaryStage.setScene(scene);**

**primaryStage.show();**

**/\*source.getRoot().setOnMousePressed(new EventHandler<MouseEvent>() {**

**@Override**

**public void handle(MouseEvent event) {**

**System.out.println("setOnMousePressed");**

**sceneStartX = event.getSceneX();**

**sceneStartY = event.getSceneY();**

**figureStartTranslateX = ((Node) event.getSource()).getTranslateX();**

**figureStartTranslateY = ((Node) event.getSource()).getTranslateY();**

**}**

**});**

**source.getRoot().setOnMouseDragged(new EventHandler<MouseEvent>() {**

**@Override**

**public void handle(MouseEvent event) {**

**System.out.println("setOnMouseDragged");**

**System.out.println(event.getSceneX() + " " + event.getSceneY());**

**source.getRoot().setTranslateX(figureStartTranslateX + event.getSceneX() - sceneStartX);**

**source.getRoot().setTranslateY(figureStartTranslateY + event.getSceneY() - sceneStartY);**

**}**

**});\*/**

**}**

**}**

**package ru.agentlab.jfxed.figures**

**import javafx.scene.Node**

**import javafx.scene.layout.Pane**

**import javafx.scene.input.MouseEvent**

**class BaseFigure {**

**protected Pane root**

**var double sceneStartX = 0**

**var double sceneStartY = 0**

**var double figureStartTranslateX = 0**

**var double figureStartTranslateY = 0**

**var Node node = null;**

**def public Pane getRoot() {**

**return root**

**}**

**def void initHandlers() {**

**root.onMousePressed = [ MouseEvent event |**

**println("setOnMousePressed")**

**sceneStartX = event.getSceneX()**

**sceneStartY = event.getSceneY()**

**node = event.source as Node**

**figureStartTranslateX = node.translateX**

**figureStartTranslateY = node.translateY**

**]**

**root.onMouseDragged = [ MouseEvent event |**

**println("setOnMouseDragged");**

**println(event.sceneX + " " + event.sceneY);**

**root.translateX = figureStartTranslateX + event.sceneX - sceneStartX**

**root.translateY = figureStartTranslateY + event.sceneY - sceneStartY**

**]**

**}**

**}**

.ClassFigure{

-fx-background-color: yellow ;

-fx-background-radius: 20;

-fx-border-color: black;

-fx-border-width: 1;

-fx-border-radius: 20;

-fx-padding: 0;

}

.Figure{

-fx-background-color: yellow ;

-fx-background-radius: 0;

-fx-border-color: black;

-fx-border-width: 0;

-fx-border-radius: 0;

-fx-padding: 5 5 5 5;

}

.Figure2{

-fx-background-color: none ;

-fx-background-radius: 0;

-fx-border-color: black;

-fx-border-width: 0;

-fx-border-radius: 0;

-fx-padding: 0 0 0 0;

-fx-stroke-type: outside

}

.ClassFigure-Name-Section{

-fx-padding: 15 50 35 50;

-fx-border-color: black;

-fx-border-width: 0;

}

.Circle1 {

-fx-alignment: top-left;

-fx-padding: 0 0 0 -10;

-fx-background-color: none ;

-fx-background-radius: 100;

-fx-border-color: black;

-fx-border-width: 1 1 1 0;

-fx-border-radius: 100;

-fx-stroke-type: inside

}

.Circle2 {

-fx-alignment: top-left;

-fx-padding: 5;

-fx-background-color: none ;

-fx-background-radius: 100;

-fx-border-color: black;

-fx-border-width: 0 1 0 0;

-fx-border-radius: 200;

}

.Circle {

-fx-alignment: top-right;

-fx-padding: 5;

-fx-background-color: none;

-fx-background-radius: 100;

-fx-border-color: black;

-fx-border-width: 0;

-fx-border-radius: 100;

-fx-stroke-type: inside

}

.ClassCircle-Text{

-fx-alignment: center;

-fx-font: bold 1pt *"Arial"*;

-fx-stroke-type: outside;

-fx-stroke-width: 0;

}

.ClassFigure-Text{

-fx-alignment: center;

-fx-font: bold 20pt *"Arial"*;

-fx-stroke-type: outside;

-fx-stroke-width: 0;

}

.ClassFigure-Methods-List{

-fx-padding: 10;

-fx-spacing: 10;

}

**package ru.agentlab.jfxed.figures.clazz**

**import javafx.scene.layout.HBox**

**import javafx.scene.layout.VBox**

**import javafx.scene.text.Text**

**import ru.agentlab.jfxed.figures.BaseFigure**

**import javafx.scene.shape.\***

**import javafx.scene.shape.Arc**

**class ClassFigure extends BaseFigure {**

**String name = "Event";**

**new (){**

**root = new VBox**

**initHandlers()**

**val s = ClassFigure.getResource("ClassFigure.css").toExternalForm()**

**root.stylesheets += s**

**root.styleClass += "ClassFigure";**

**root.children += new HBox => [**

**styleClass += "ClassFigure-Name-Section"**

**children += new Text => [**

**text = name**

**styleClass += "ClassFigure-Text"**

**root.children += new HBox => [**

**children += new HBox => [**

**styleClass += "Figure"**

**]**

**children += new HBox => [**

**styleClass += "Figure2"**

**children += new HBox => [**

**styleClass += "Circle2"**

**translateX = 5;**

**]**

**]**

**children += new HBox => [**

**styleClass += "Circle1"**

**children += new Text => [**

**text = " "**

**styleClass += "ClassCircle-Text"**

**]**

**]**

**/\*children += new HBox => [**

**styleClass += "Circle1"**

**children += new HBox => [**

**children += new HBox => [**

**styleClass += "Circle1"**

**]**

**]**

**]\*/**

**styleClass += "Circle"**

**]**

**]**

**]**

**}**

**}**

.ClassFigure{

-fx-background-color: YELLOW;

-fx-background-radius: 5;

-fx-border-color: black;

-fx-border-width: 1;

-fx-border-radius: 5;

-fx-alignment: CENTER;

-fx-padding: 20 40 20 40;

}

.ClassFigure-Rect{

-fx-fill: CYAN;

-fx-stroke: #FFFFFF;

-fx-stroke-type: inside;

-fx-border-radius: 5;

-fx-background-radius: 5;

-fx-arc-height: 5;

-fx-arc-width: 5;

-fx-padding: 5;

}

.ClassFigure-Name-Section{

-fx-alignment: center;

-fx-padding: 5;

-fx-border-color: black;

-fx-border-width: 0 0 1 0;

-fx-border-radius: 5 5 0 0;

}

.ClassFigure-Line{

}

/\* .ClassFigure-Text{

-fx-font: bold 14pt "Arial";

-fx-stroke-type: outside;

-fx-stroke-width: 0;

}\*/

.ClassFigure-Methods-List{

-fx-padding: 6;

-fx-spacing: 70;

}

**package ru.agentlab.jfxed.figures.clazz**

**import javafx.scene.layout.HBox**

**import javafx.scene.layout.VBox**

**import javafx.scene.text.Text**

**import ru.agentlab.jfxed.figures.BaseFigure**

**class ClassKok extends BaseFigure {**

**String name = " ";**

**new (){**

**root = new VBox**

**initHandlers()**

**val s = ClassKok.getResource("ClassKok.css").toExternalForm()**

**root.stylesheets += s**

**//root.styleClass += "ClassFigure";**

**/\* root.children += new HBox => [**

**styleClass += "ClassFigure-Name-Section"**

**children += new Text => [**

**text = name**

**styleClass += "ClassFigure-Text"**

**]**

**]**

**root.children += new VBox => [**

**styleClass += "ClassFigure-Methods-List"**

**children += new Text => [**

**text = " Event "**

**]**

**]\*/**

**}**

**}**

Результат компиляции:

