

# Разработка объектно-ориентированного приложения

Сидоров М.Е.

Приложение позволяет рассчитать значения некоторой заданной функции и отобразить ее на графике. При этом можно изменять интервал построения графика по оси абсцисс, Рисунок 1.

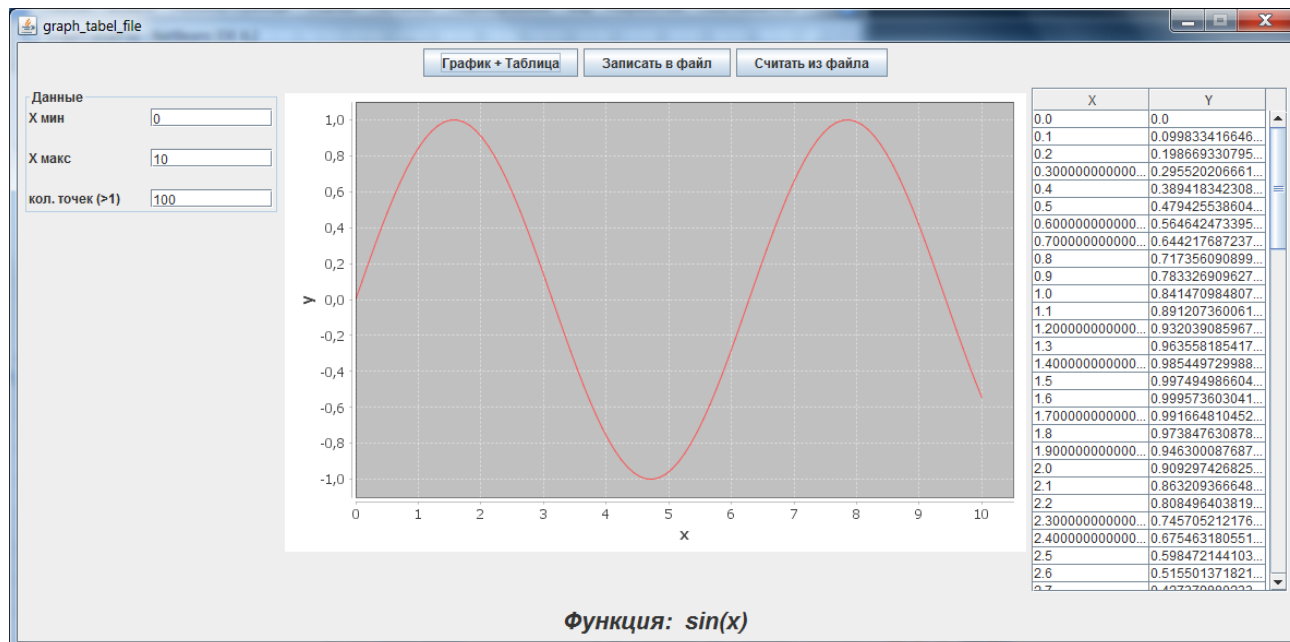


Рисунок 1

## Обязанности классов проекта

№	Имя класса	Обязанность
1.	<b>Main</b>	запуск приложения и создание объекта - формы
2.	<b>MainFrame</b>	создание объектов - панелей
3.	<b>WestPanel</b>	создание панели с элементами управления для ввода данных
4.	<b>CenterPanel</b>	создание панели отображения графика функции
5.	<b>SouthPanel</b>	создание панели отображения вида функции
6.	<b>EastPanel</b>	создание панели отображения таблицы
7.	<b>NorthPanel</b>	создание панели с кнопками
8.	<b>MakeGraph</b>	построение графика функции
9.	<b>MakeTable</b>	построение таблицы
10.	<b>RWRandomAccessFile</b>	методы работы с файлом (запись и считывание данных )
11.	<b>Verify</b>	проверка корректности введенных данных
12.	<b>MyFunc</b>	задание вида функции
13.	<b>ActionBtn1, ActionBtn2, ActionBtn3</b>	обработка событий нажатия кнопок

# Разработка классов

## Задание № 1

1. Запустите *NetBeans* и создайте новый проект.
2. На первом шаге выберите Категорию *Java*, Проекты – *Приложение Java*.
3. На втором шаге укажите имя проекта *GraphTabelFile*, расположение *D:\Appl*, и имя главного класса *graphtabelfile.Main*.
4. В файле *Main.java* наберите код классов *Main* и *MyFunc*, Рисунок 2.

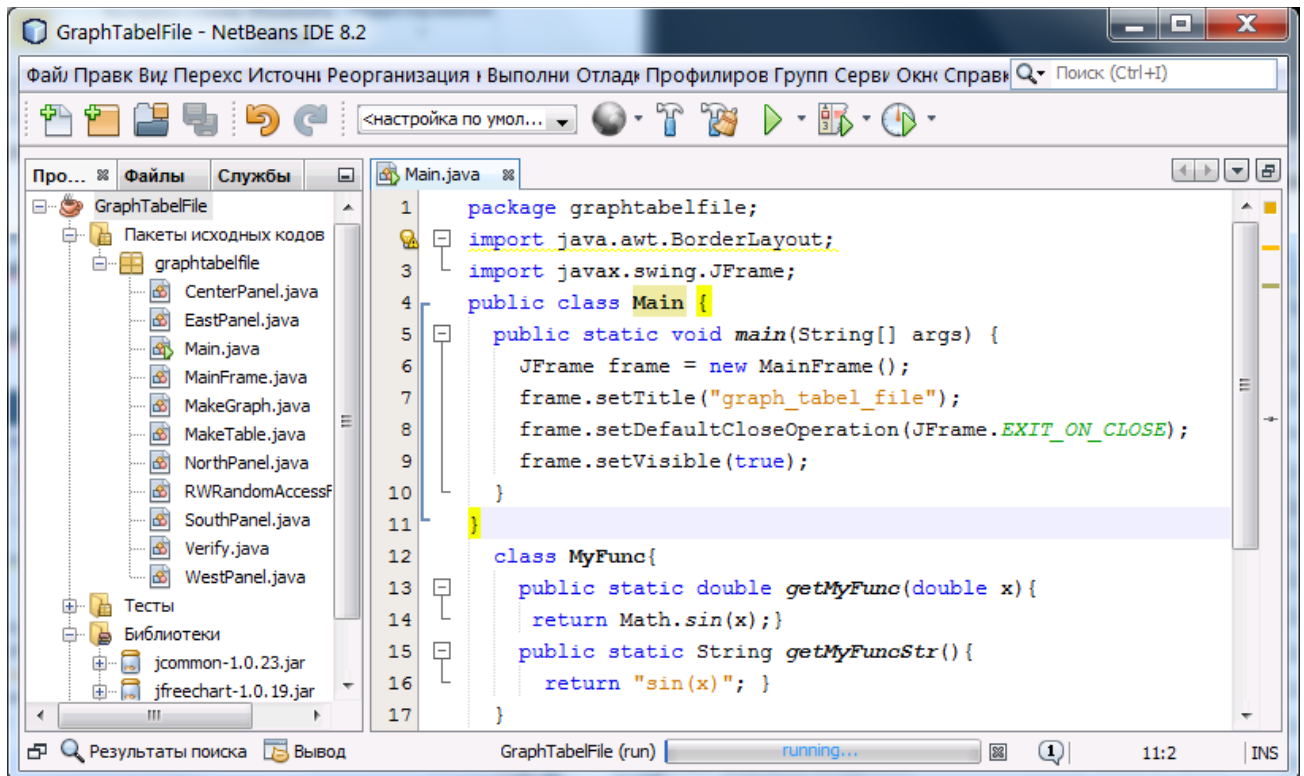


Рисунок 2

5. Загрузите архивный файл *jfreechart* с сайта *jfree.org*: <http://www.jfree.org/jfreechart/download.html> и распакуйте его в папку проекта.
6. Для подключения классов, обеспечивающих построения графика, в контекстном меню раздела проекта *Библиотеки* выберите *Добавить файл JAR/папку* и укажите файлы *jcommon-1.0.23.jar* и *jfreechart-1.0.19.jar* из распакованной папки *\jfreechart-1.0.19\lib\*.

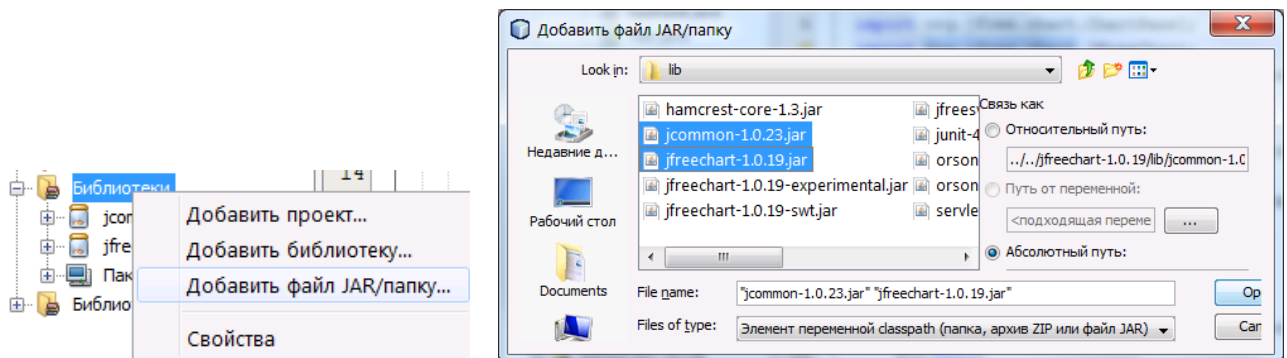


Рисунок 3

7. Создайте в пакете *graphtabelfile* новый класс с именем *MainFrame* и наберите код, Рисунок 4.

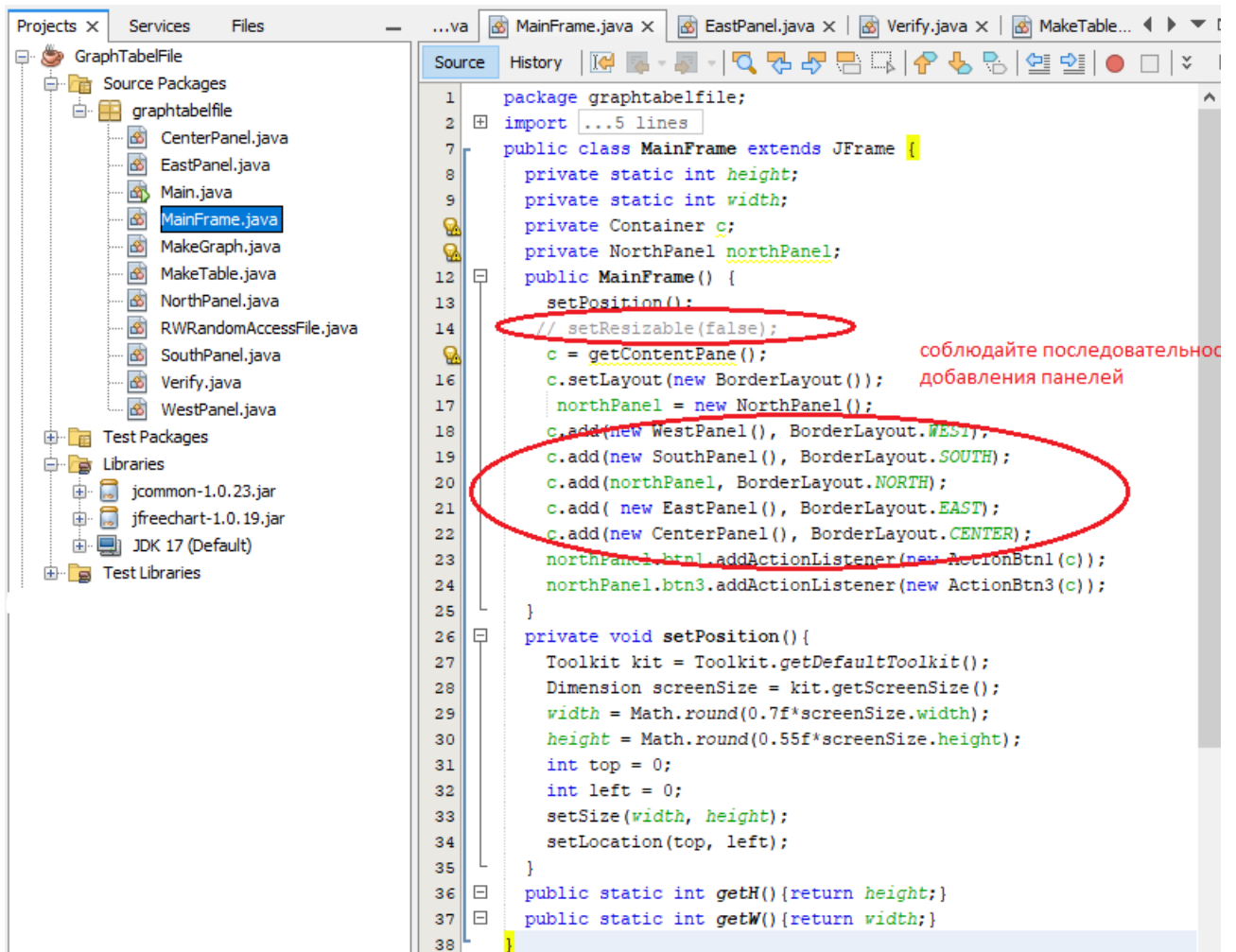


Рисунок 4

8. Создайте в пакете *graphtablefile* новый класс с именем *SouthPanel* и наберите код, Рисунок 5.

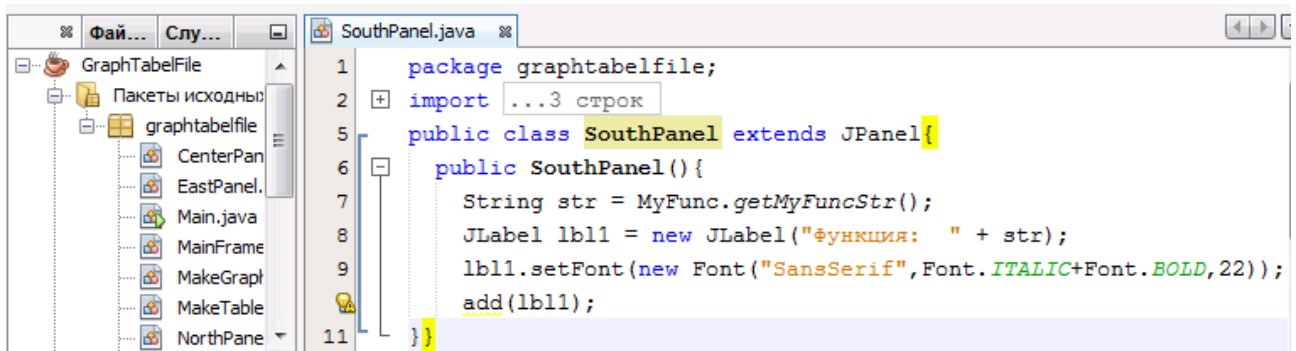


Рисунок 5

9. Создайте в пакете *graphtablefile* новый класс с именем *MakeGraph* и наберите код, Рисунок 6.

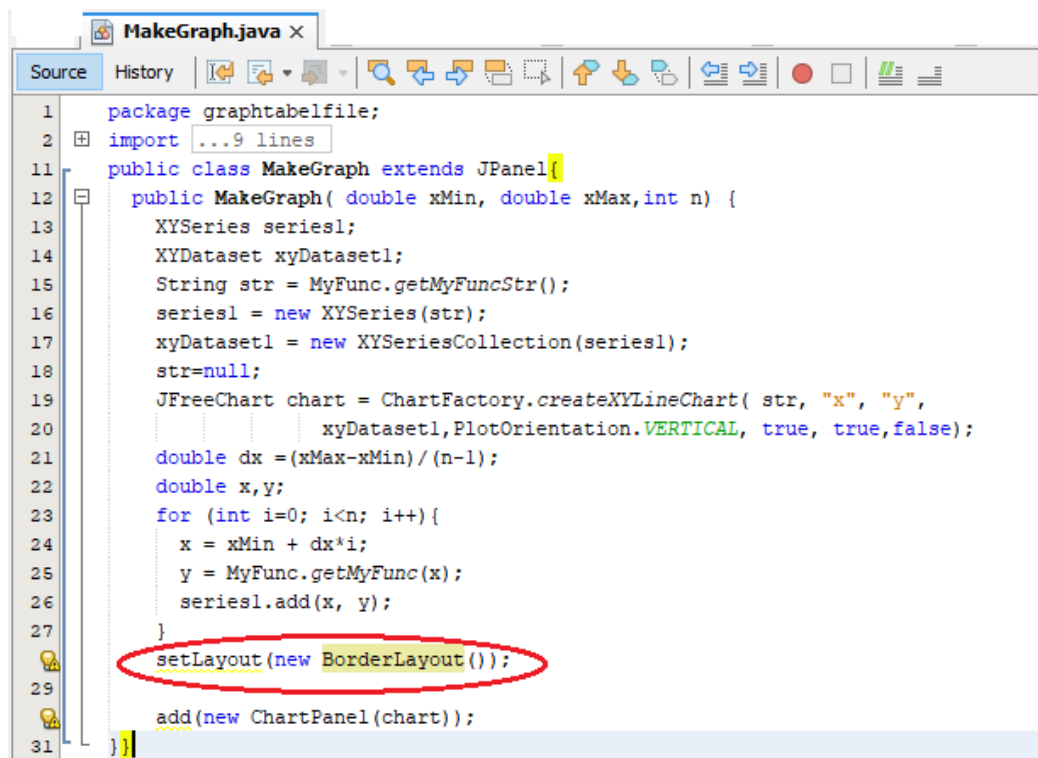


Рисунок 6

10. Создайте в пакете *graphtabelfile* новый класс с именем *NorthPanel* и наберите код, Рисунок 7, 8.

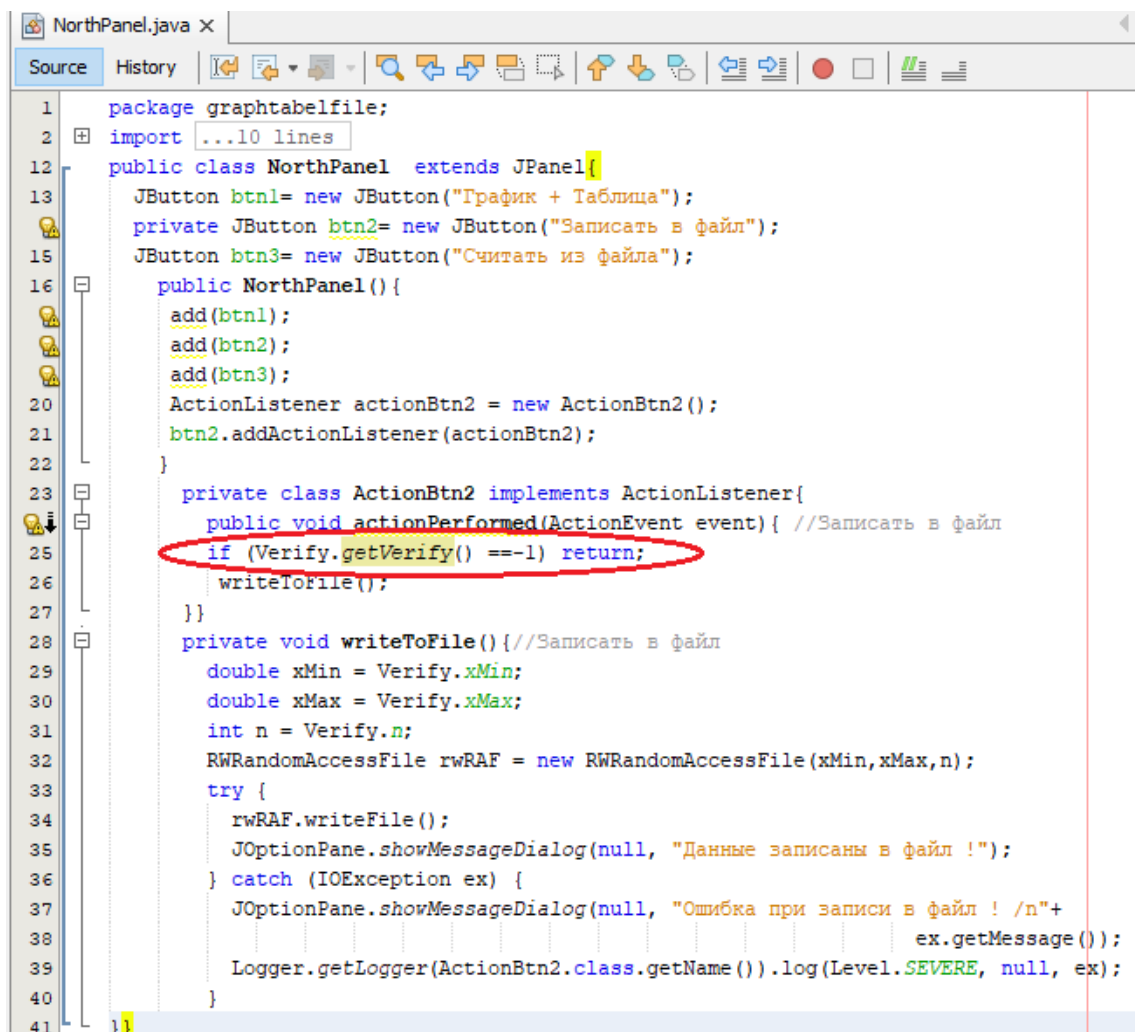


Рисунок 7

```

42 class ActionBtn1 implements ActionListener{
43     private Container c; // все панели на фрейме
44     ActionBtn1(Container c){ this.c = c; }
45     @Override
46     public void actionPerformed(ActionEvent event){
47         if (Verify.getVerify() == -1) return;
48         c.remove(4);
49         c.remove(3);
50         c.add( new EastPanel(), BorderLayout.EAST);
51         c.add(new CenterPanel(), BorderLayout.CENTER);
52         c.validate();
53     }
54 class ActionBtn3 implements ActionListener{
55     private Container c; // все панели на фрейме
56     ActionBtn3(Container c){ this.c = c; }
57     @Override
58     public void actionPerformed(ActionEvent event){
59         readFromFile(); // считать из файла и отобразить график и таблицу
60         if (Verify.getVerify() == -1) return;
61         c.remove(4);
62         c.remove(3);
63         c.add( new EastPanel(), BorderLayout.EAST);
64         c.add(new CenterPanel(), BorderLayout.CENTER);
65         c.validate();
66     }
67     private void readFromFile(){
68         String xMin, xMax, n;
69         double[] y;
70         RWRandomAccessFile rwRAF=new RWRandomAccessFile();
71         try {
72             y = rwRAF.readFile();
73             JOptionPane.showMessageDialog(null, "Данные считаны из файла !");
74         } catch (IOException ex) {
75             JOptionPane.showMessageDialog(null, "Ошибка при считывании файла ! /n"+
76                                     ex.getMessage());
77             Logger.getLogger(ActionBtn3.class.getName()).log(Level.SEVERE, null, ex);
78         }
79         n = Integer.toString(rwRAF.getN());
80         xMin =Double.toString(rwRAF.getXmin());
81         xMax =Double.toString(rwRAF.getXmax());
82         WestPanel.setText1(xMin);
83         WestPanel.setText2(xMax);
84         WestPanel.setText3(n);
85     }

```

Рисунок 8

11. Создайте в пакете *graphtabelfile* новый класс с именем *CenterPanel* и наберите код, Рисунок 9.

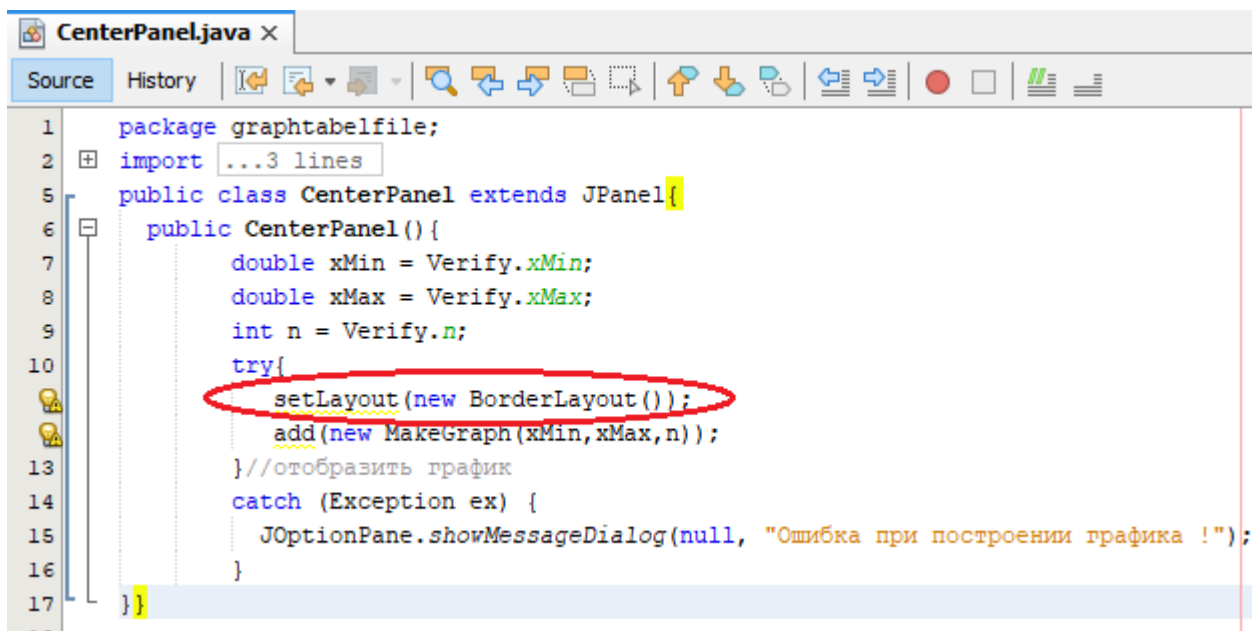


Рисунок 9

12. Создайте в пакете *graphtabelfile* новый класс с именем *MakeTable* и наберите код, Рисунок 10.

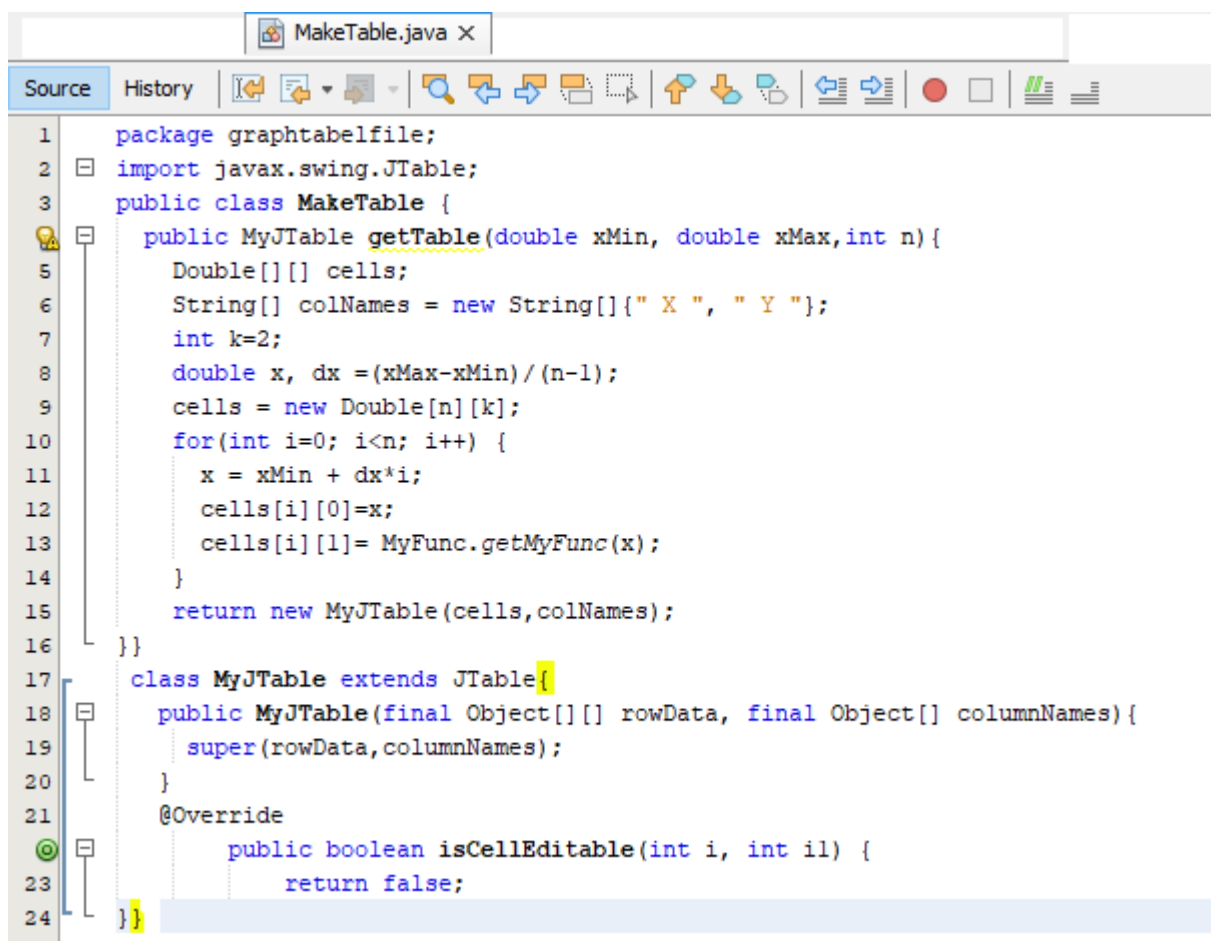
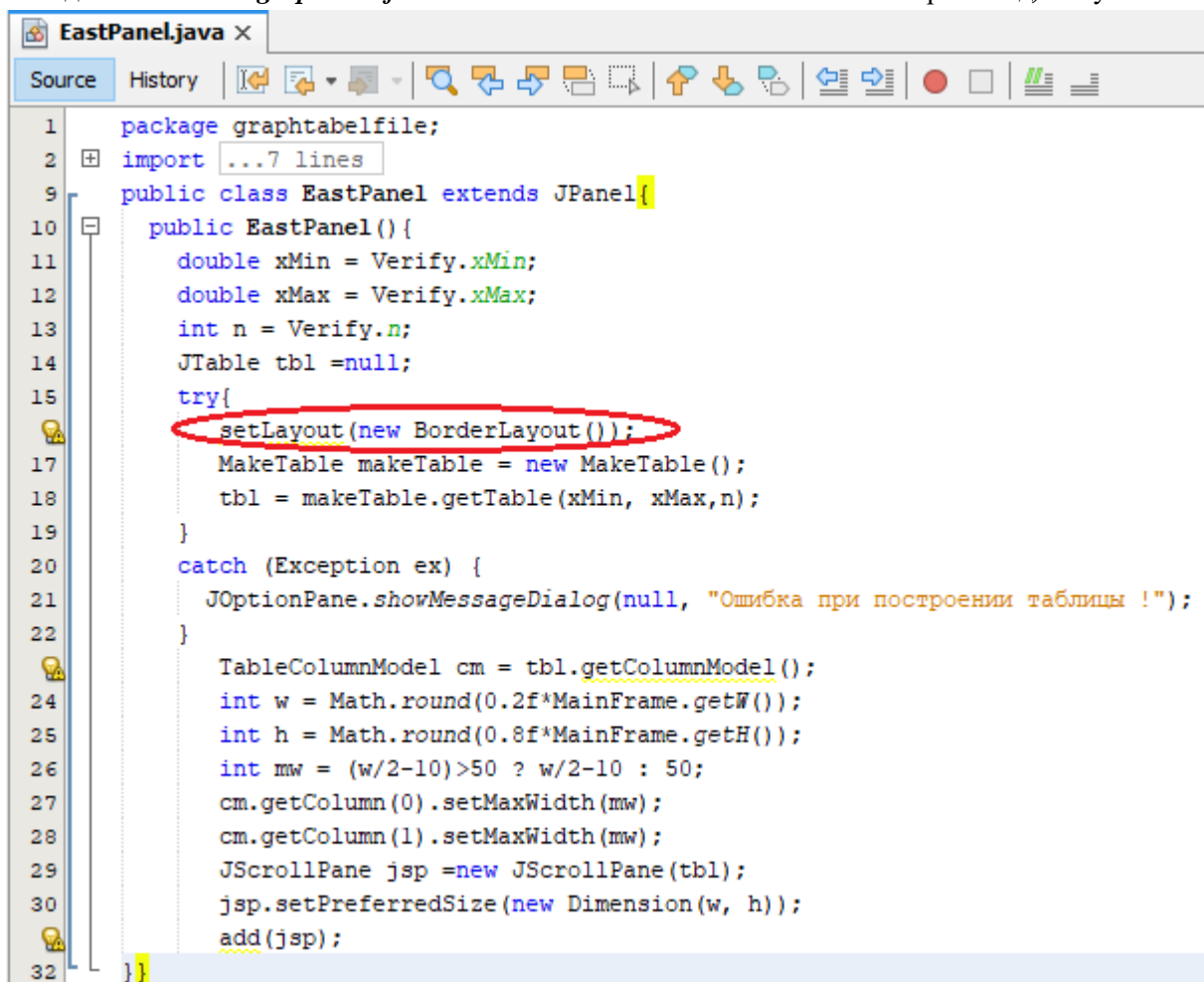


Рисунок 10



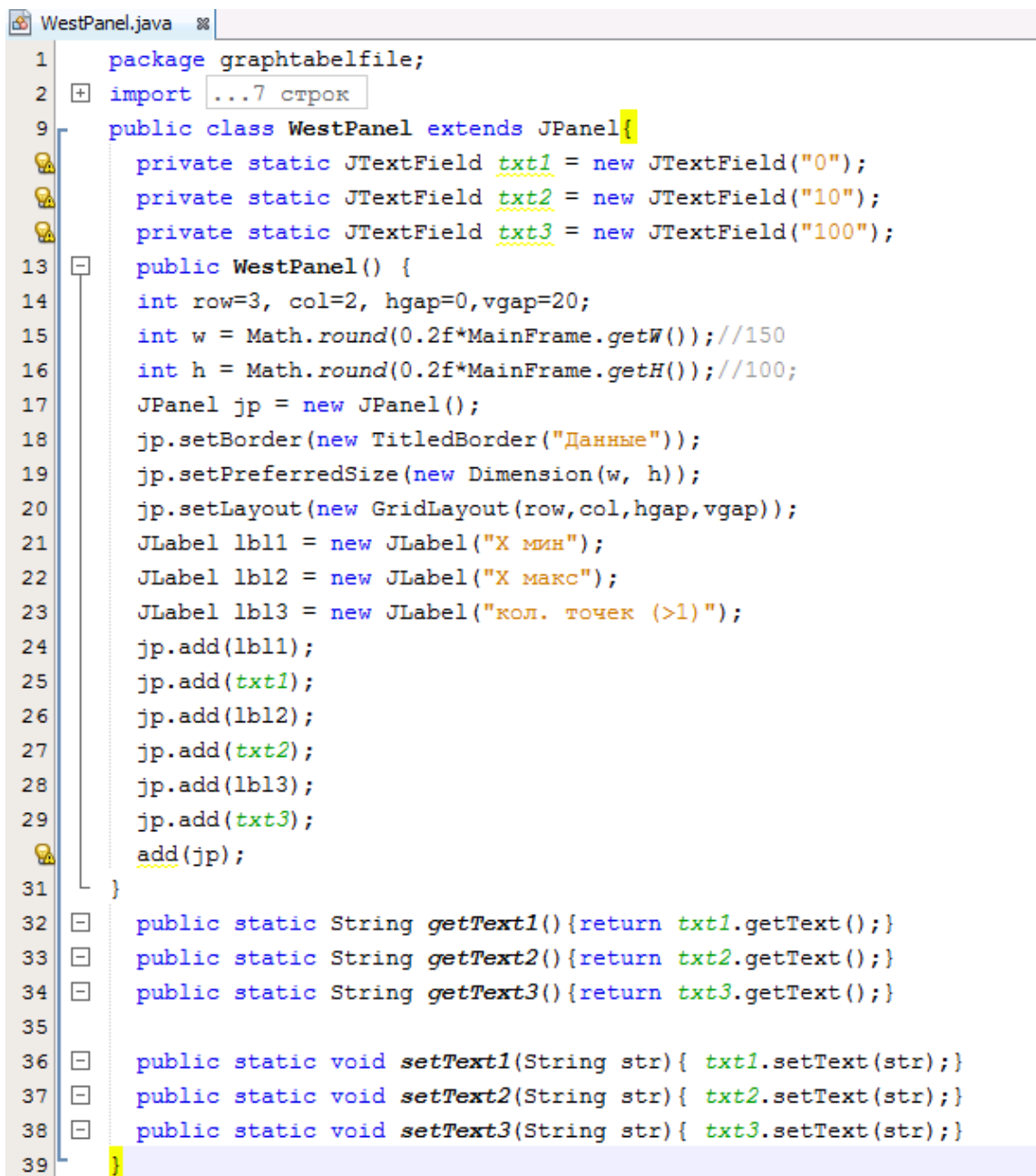
13. Создайте в пакете *graphtabelfile* новый класс с именем *EastPanel* и наберите код, Рисунок 11.



```
1 package graphtabelfile;
2 import ...7 lines
9 public class EastPanel extends JPanel{
10     public EastPanel(){
11         double xMin = Verify.xMin;
12         double xMax = Verify.xMax;
13         int n = Verify.n;
14         JTable tbl =null;
15         try{
16             setLayout(new BorderLayout());
17             MakeTable makeTable = new MakeTable();
18             tbl = makeTable.getTable(xMin, xMax,n);
19         }
20         catch (Exception ex) {
21             JOptionPane.showMessageDialog(null, "Ошибка при построении таблицы !");
22         }
23         TableColumnModel cm = tbl.getColumnModel();
24         int w = Math.round(0.2f*MainFrame.getW());
25         int h = Math.round(0.8f*MainFrame.getH());
26         int mw = (w/2-10)>50 ? w/2-10 : 50;
27         cm.getColumn(0).setMaxWidth(mw);
28         cm.getColumn(1).setMaxWidth(mw);
29         JScrollPane jsp =new JScrollPane(tbl);
30         jsp.setPreferredSize(new Dimension(w, h));
31         add(jsp);
32     }
}
```

Рисунок 11

14. Создайте в пакете *graphtabelfile* новый класс с именем *WestPanel* и наберите код, Рисунок 12.



```
1 package graphtabelfile;
2 import ...7 строк
9 public class WestPanel extends JPanel{
10     private static JTextField txt1 = new JTextField("0");
11     private static JTextField txt2 = new JTextField("10");
12     private static JTextField txt3 = new JTextField("100");
13     public WestPanel() {
14         int row=3, col=2, hgap=0,vgap=20;
15         int w = Math.round(0.2f*MainFrame.getW()); //150
16         int h = Math.round(0.2f*MainFrame.getH()); //100;
17         JPanel jp = new JPanel();
18         jp.setBorder(new TitledBorder("Данные"));
19         jp.setPreferredSize(new Dimension(w, h));
20         jp.setLayout(new GridLayout(row,col,hgap,vgap));
21         JLabel lbl1 = new JLabel("X мин");
22         JLabel lbl2 = new JLabel("X макс");
23         JLabel lbl3 = new JLabel("кол. точек (>1)");
24         jp.add(lbl1);
25         jp.add(txt1);
26         jp.add(lbl2);
27         jp.add(txt2);
28         jp.add(lbl3);
29         jp.add(txt3);
30         add(jp);
31     }
32     public static String getText1(){return txt1.getText();}
33     public static String getText2(){return txt2.getText();}
34     public static String getText3(){return txt3.getText();}
35
36     public static void setText1(String str){ txt1.setText(str);}
37     public static void setText2(String str){ txt2.setText(str);}
38     public static void setText3(String str){ txt3.setText(str);}
39 }
```

Рисунок 12



15. Создайте в пакете *graphtabelfile* новый класс с именем *RWRandomAccessFile* и наберите код, Рисунок 13.

```
RWRandomAccessFile.java
1  package graphtabelfile;
2  import ...3 строк
5  public class RWRandomAccessFile {
6      private double xMin, xMax;
7      private int n;
8      private String file = "rWtest.dat";
9      RWRandomAccessFile(double xMin, double xMax, int n) {
10         this.xMin = xMin;
11         this.xMax = xMax;
12         this.n = n;
13     }
14     RWRandomAccessFile() {};
15     void writeFile() throws FileNotFoundException, IOException {
16         RandomAccessFile rf = new RandomAccessFile(file, "rw");
17         rf.writeInt(n);
18         rf.writeDouble(xMin);
19         rf.writeDouble(xMax);
20         for(int i = 0; i < n; i++) {
21             double x = xMin + (xMax - xMin) * i / (n - 1);
22             rf.writeDouble(MyFunc.getMyFunc(x));
23         }
24         rf.close();
25     }
26     double[] readFile() throws IOException {
27         RandomAccessFile rf = new RandomAccessFile(file, "r");
28         n = rf.readInt();
29         xMin = rf.readDouble();
30         xMax = rf.readDouble();
31         double[] y = new double[n];
32         for(int i = 0; i < n; i++) y[i] = rf.readDouble();
33         rf.close();
34         return y;
35     }
36     int getN() { return n; };
37     double getXmin() { return xMin; };
38     double getXmax() { return xMax; };
39 }
```

Рисунок 13

16. Создайте в пакете *graphtabelfile* новый класс с именем *Verify* и наберите код, Рисунок 14.

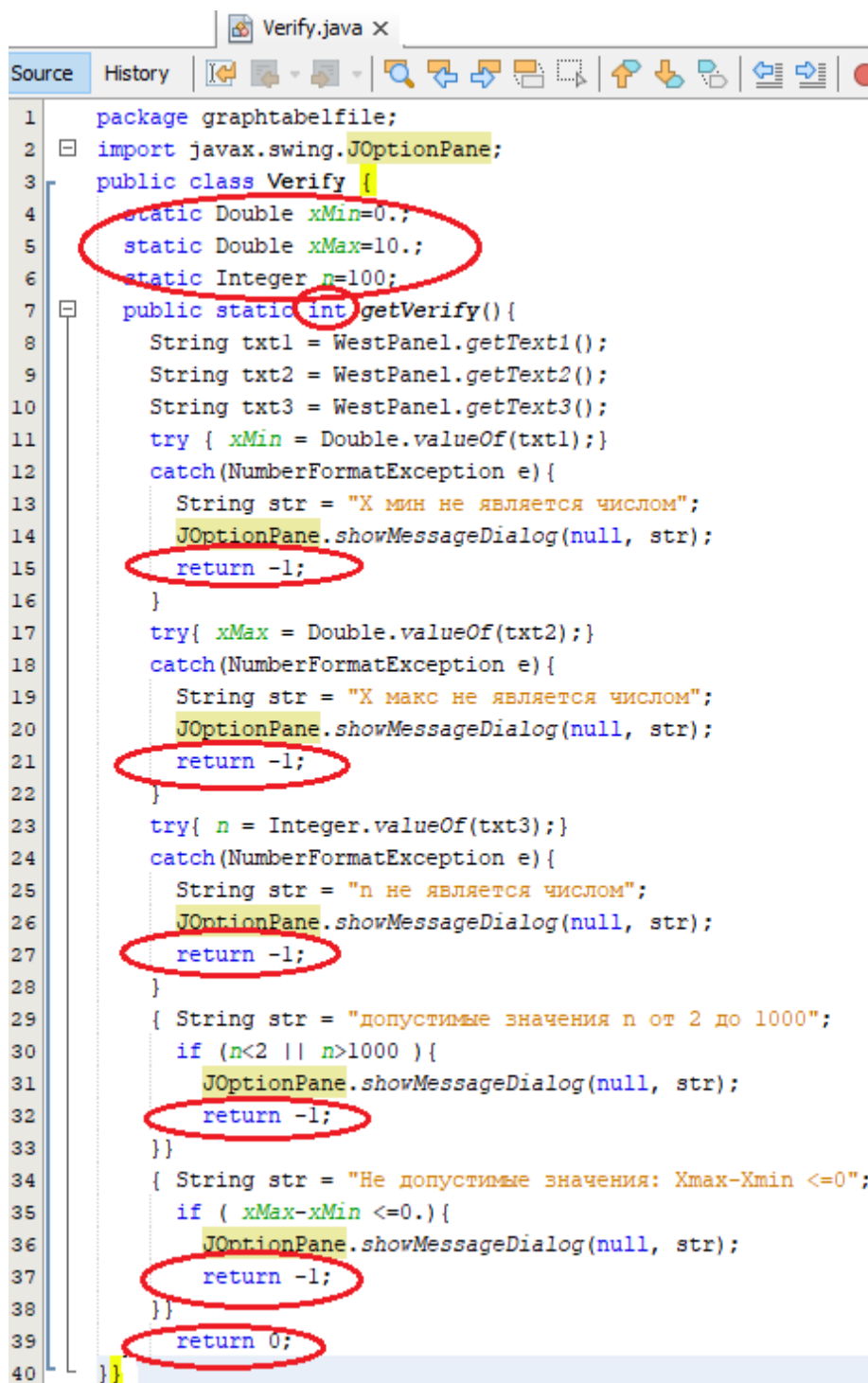


Рисунок 14

17. Запустите проект на выполнение (*Выполнить, Запустить проект*) и проверьте работу элементов управления.
18. Создайте исполнимый файл проекта – *Выполнить, Очистить и собрать проект*. Откройте папку проекта *GraphTabelFile\dist* и запустите на выполнение файл *GraphTabelFile.jar*.
19. Протестируйте проект при различных данных и работу кнопок (вывод графика и таблицы, запись в файл и чтение из файла).