

МИНИСТЕРСТВО ОБЩЕГО И ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
РОСТОВСКОЙ ОБЛАСТИ
Государственное бюджетное профессиональное образовательное учреждение
Ростовской области
«Гуковский строительный техникум»

МЕТОДИЧЕСКАЯ РАЗРАБОТКА

открытого урока

по дисциплине «Основы алгоритмизации и программирования» для
студентов специальности

09.02.07

«Информационные системы и программирование»

«Создание элемента визуального интерфейса в Java»

Преподаватель: Голубниченко А.А.



Гуково 2022 г.

Тема урока: «Создание элемента визуального интерфейса в Java»

Цель урока:

Образовательные:

- Повторить основные понятия;
- Научиться создавать элемент визуального интерфейса;
- Научиться изменять интерфейс Кнопки JButton.

Развивающие:

1. развитие логического мышления, познавательных интересов, умения излагать мысли;
2. развитие практических навыков работы с Java,
3. развивать алгоритмическое мышление.

Воспитательные:

1. воспитание организованности, аккуратности;
2. способствовать повышению мотивации освоения программирования;
3. воспитывать у обучающихся умение самостоятельно решать вопросы, а также работать в группе;

Планируемые образовательные результаты:

предметные –создавать элементы управления Java;

метапредметные – организовать применение студентами полученных знаний в процессе самостоятельной работы; развивать умение самостоятельно ставить цели.

Общие компетенции: ОК.01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам, ОК.04 Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами, ОК.09 Использовать информационные технологии в профессиональной деятельности;

Профессиональные компетенции:

ПК 1.1 Реализовать построенные алгоритмы в виде программ на конкретном языке программирования;

ПК 1.5 Оформлять код программы в соответствии со стандартом кодирования;

Знать: объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения.

личностные – понимание роли информационных систем и баз данных в жизни современного человека, в освоении специальности

Тип урока: комбинированный

Методы обучения: Решение конкретных задач программирования, работа в группах, работа в онлайн компиляторе Java.

Материальное оснащение:

Ноутбуки планшетные Dell, интерактивная панель SmartVisio; ПО Microsoft Windows 10, PowerPoint, IntelliJ IDEA, раздаточный материал – Технологическая карта для самостоятельной работы на закрепление.

Основные понятия, рассматриваемые на уроке:

«При нажатии» onClick()

Выполнять Execute

Метод Method

Вызов Calling

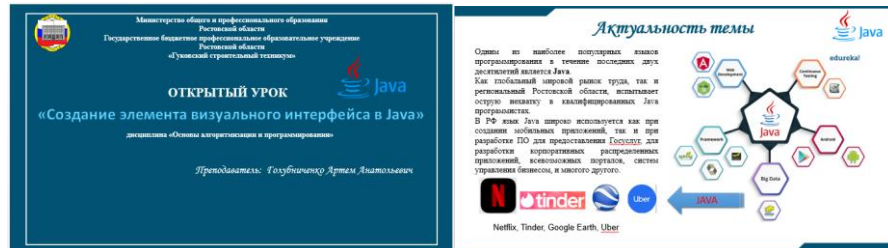
Выражения Expressions

ПЛАН УРОКА

1. Актуализация опорных знаний «Введение в программирование»
<https://joyteka.com/ru> Joyteka
2. Объяснение материала, работа в IntelliJ IDEA
3. Выполнение проекта IntelliJ IDEA
4. Проверка знаний, онлайн тест TestPad
5. Формирование профессионального облака тегов
6. Рефлексия, виртуальная доска Lino
7. Подведение итога урока

ХОД УРОКА

I Организационный момент: Проверка готовности аудитории и обучающихся к занятию.



II Актуализация знаний:

Одним из наиболее популярных языков программирования в течение последних двух десятилетий является Java. Как глобальный мировой рынок труда, так и региональный Ростовской области, испытывает острую нехватку в квалифицированных Java программистах. В РФ язык Java широко используется как при создании мобильных приложений, так и при разработке ПО для предоставления госуслуг, для разработки корпоративных распределенных приложений, всевозможных порталов, систем управления бизнесом, и многого другого.

Сегодня будет рассмотрено создание класса на языке java и последующее обращение к этому классу, то есть мы научимся **основам работы с классами**.

Студенты самостоятельно определяют цели и задачи урока

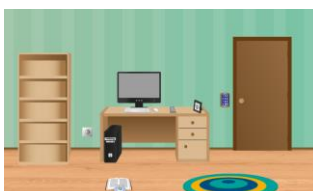


Веб- квест «Введение в программирование» <https://joyteka.com/100028565>

- Небольшая программа, которая распространяется с одного компьютера на другой и мешает работе компьютера (Вирус);
- Система условных знаков для представления информации (Код)
- Какой это язык программирования? (Java)
- Человек, занимающийся разными видами компьютерного мошенничества: цифровым взломом, выводом из строя оборудования, кражей, подменой, удалением данных. (Хакер)
- Набор инструкций, описывающих порядок действий исполнителя для достижения результата решения задачи за конечное число действий (Алгоритм)



Начало Квеста



2. Убрать книги со шкафа, отодвинуть шкаф



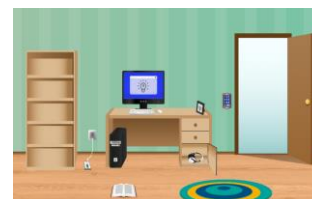
3. Взять диск из выпавшей книги, вставить его в системный блок



4. Переместить коврик, достать зарядку на телефон



5 Открыть стол, Взять наушники



6 Ответить на вопросы Квеста Дверь открыта, Квест выполнили

III Объяснение нового материала

Работать мы будем в IntelliJ IDEA- Интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java, JavaScript, Python, разработанная компанией JetBrains.

«Создание элемента визуального интерфейса в Java»

Java — не просто язык программирования, а целая программная платформа с широкими возможностями. Основными составляющими блоками этой платформы являются: Базовые инструменты для написания и запуска программ на Java. Библиотеки и классы — ядро языка.

Каждое приложение, которое имеет графический интерфейс пользователя не может обходиться без кнопок. В Java Swing кнопка представлена классом `JButton`. У кнопки имеются различные методы для ее конфигурирования — установка надписи на `JButton`, установка иконки, выравнивание текста, установка размеров.

«Создание элемента визуального интерфейса в Java»

Кнопка JButton

```
JButton button = new JButton("Кнопка");
button.addActionListener(new ButtonAction());
```

Внешний вид кнопок `JButton` можно легко изменить, не меняя менеджера внешнего вида и поведения. С интерфейсом кнопок можно делать практически все — сопоставлять каждому действию пользователя своё изображение, убирать рамку, закрашивать в любой цвет, перемещать содержимое по разным углам.

```
import javax.swing.*;
import java.awt.*;

public class JButtonExample {
    public static void main(String[] args) {
        JFrame frame = new JFrame("JButton Example");
        JButton button = new JButton("Click Me");
        button.setBackground(Color.RED);
        button.setForeground(Color.WHITE);
        button.setMargin(new Insets(10, 20, 10, 20));
        button.setHorizontalAlignment(SwingConstants.LEFT);
        button.setVerticalAlignment(SwingConstants.TOP);
        frame.add(button);
        frame.setSize(200, 100);
        frame.setVisible(true);
    }
}
```

Кнопка создается одним из пяти конструкторов, в частности:

- `JButton()`
- `JButton(String text)`
- `JButton(Icon icon)`
- `JButton(String text, Icon icon)`

Кнопка — это прямоугольник с текстом (или иконкой), по которому пользователь щелкает, чтобы вызвать какое-то действие (или о чем-то оповестить).

Каждое приложение, которое имеет графический интерфейс пользователя не может обходиться без кнопок. В Java Swing кнопка представлена классом `JButton`. У кнопки имеются различные методы для ее конфигурирования — установка надписи на `JButton`, установка иконки, выравнивание текста, установка размеров и так далее. Как это сделать? Поговорим об этом.

Библиотека Swing включает абсолютно все придуманные на сегодняшний день элементы управления. К ним относятся кнопки, флажки, переключатели, меню и его элементы, и многое другое. Все эти элементы в библиотеке связаны, поскольку они унаследованы от абстрактного класса `AbstractButton`, определяющего поведение любого компонента, претендующего на звание элемента управления.

Кнопки JButton

Кнопки `JButton` кроме собственного внешнего вида не включают практически ничего уникального. Поэтому всё, что верно для кнопок, будет верно и для остальных элементов управления. Пример кода создания обычной кнопки :

```
JButton button = new JButton("Кнопка");
button.addActionListener(new ButtonAction());
```

Интерфейс кнопок

Внешний вид кнопок `JButton` можно легко изменить, не меняя менеджера внешнего вида и поведения. С интерфейсом кнопок можно делать практически все — сопоставлять каждому действию пользователя своё изображение, убирать рамку, закрашивать в любой цвет, перемещать содержимое по разным углам, не рисовать фокус.

Рассмотрим пример, в котором будут созданы кнопки `JButton`;

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class GoodLesson extends JFrame implements ActionListener {
    JPanel panel = new JPanel();
```

```

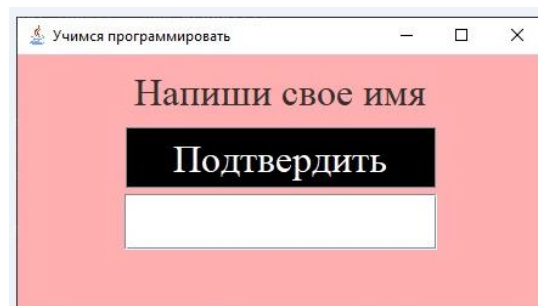
JLabel label1 = new JLabel();
JButton btn1 = new JButton("Кнопка 1");
JTextArea txtArea = new JTextArea(5, 20);
JTextField textField1 = new JTextField(11);
Font customFont = new Font("Serif", Font.PLAIN, 32);
public GoodLesson() {
    setTitle("Учимся программировать");
    this.setIconImage(new ImageIcon(getClass().getResource("java.png")).getImage());
    Dimension screenSize = Toolkit.getDefaultToolkit().getScreenSize();
    int sizeWidth = 450;
    int sizeHeight = 350;
    int locationX = (screenSize.width - sizeWidth) / 2;
    int locationY = (screenSize.height - sizeHeight) / 2;
    setBounds(locationX, locationY, sizeWidth, sizeHeight);
    setSize(450, 250);
    setDefaultCloseOperation(EXIT_ON_CLOSE);
    add(panel);
    setVisible(true);
    panel.setEnabled(false);
    btn1.setText("Подтвердить"); // МЕНЯТЬ НАДПИСИ
    label1.setText("Напиши свое имя");//
    btn1.setBackground(Color.BLACK); // МЕНЯТЬ ЦВЕТА
    btn1.setForeground(Color.WHITE);
    panel.setBackground(Color.PINK);
    label1.setFont(customFont);
    panel.add(label1);
    panel.add(btn1);
    panel.add(textField1);
    textField1.setFont(customFont);
    textField1.setHorizontalAlignment(JTextField.CENTER);
    btn1.setFont(customFont);
    label1.setHorizontalAlignment(JTextField.CENTER);
    btn1.setPreferredSize(new Dimension(255, 50));
    label1.setPreferredSize(new Dimension(255, 50));
    btn1.addActionListener(this);
}

public void actionPerformed(ActionEvent event)
{
    label1.setText("Привет, "+textField1.getText()+" !");
} // Менять вывод надписи

public static void main(String[] args) {
    GoodLesson gui = new GoodLesson();
}
}

```

Результат;



IV Выполнение самостоятельной работы

Используя технологическую карту

- Измените цвет кнопки;
- Измените цвет фона;
- Изменить текст Приветствия

«Создание элемента визуального интерфейса в Java»

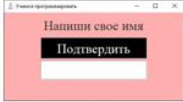
1. Измените цвет кнопки
2. Изменить цвет фона
3. Изменить текст приветствия

Обозначения основных цветов

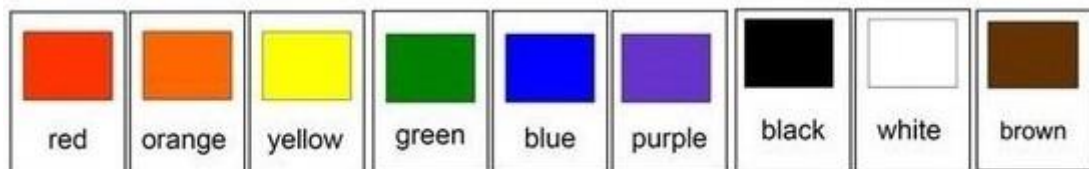
red	orange	yellow	green	blue	purple	black	white	brown
-----	--------	--------	-------	------	--------	-------	-------	-------

```
btn1.setText("Подтвердить"); // МЕНЯТЬ НАДПИСИ
label1.setText("Напишите свое имя"); //
btn1.setBackground(Color.BLACK); // МЕНЯТЬ ЦВЕТА
btn1.setForeground(Color.WHITE);
panel.setBackground(Color.PINK);
```

```
add(panel);
setVisible(true);
panel.setEnabled(false);
btn1.setText("Подтвердить"); // МЕНЯТЬ НАДПИСИ
label1.setText("Напишите свое имя"); //
btn1.setBackground(Color.BLACK); // МЕНЯТЬ ЦВЕТА
btn1.setForeground(Color.WHITE);
panel.setBackground(Color.PINK);
label1.setFont(customFont);
panel.add(label1);
panel.add(btn1);
panel.add(textField1);
textField1.setFont(customFont);
textField1.setHorizontalAlignment(JTextField.CENTER);
btn1.setFont(customFont);
label1.setHorizontalAlignment(JTextField.CENTER);
btn1.setPreferredSize(new Dimension(100, 25, 50));
label1.setPreferredSize(new Dimension(100, 25, 50));
btn1.addActionListener(this);
```



Обозначения основных цветов



Цветовая модель RGB

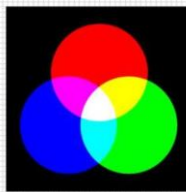
Д. Максвелл, 1860

цвет = (R, G, B)

red green blue

красный зеленый синий

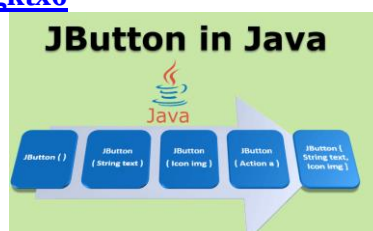
0.255 0.255 0.255



■ (0, 0, 0)	■ (0, 255, 0)
■ (255, 255, 255)	■ (255, 255, 0)
■ (255, 0, 0)	■ (0, 0, 255)
■ (255, 150, 150)	■ (100, 0, 0)

V. Проверка знаний Выполнение онлайн теста на платформе TestPad

<https://onlinetestpad.com/f2hamba7gktx6>



Тест содержит пять вопросов

Мини тест к уроку "Создание элемента интерфейса в Java"

1

Элементы управления: кнопки, флажки, переключатели, меню и его элементы, и многое другое образуют графический

Далее

Завершить

2

С интерфейсом кнопок можно делать

- ☐ внедрить вирус
- ☐ закрашивать в любой цвет
- ☐ добавлять изображение

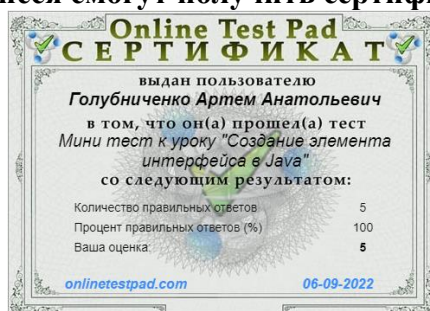
3

btn1.setBackground(Color.BLACK) эта строка кода задает

- ☐ изображение
- ☐ рамку
- ☐ цвет фона

4 В Java класс JButton это ☐ вкладка ☐ скрепка ☐ кнопка	5 btn1.setText("Подтвердить") Строка кода устанавливает ☐ надпись в кнопке ☐ цвет рамки кнопки ☐ цвет заливки кнопки
---	--

По итогам теста обучающиеся смогут получить сертификат, с указанием оценки



VI. Формирование профессиональной терминологии

Создать облако тегов из новых слов изучаемой темы

Облако слов - создать онлайн (wordscound.pythonanywhere.com)

<https://www.xn--80abe5adcqeb2a.xn--p1ai/>

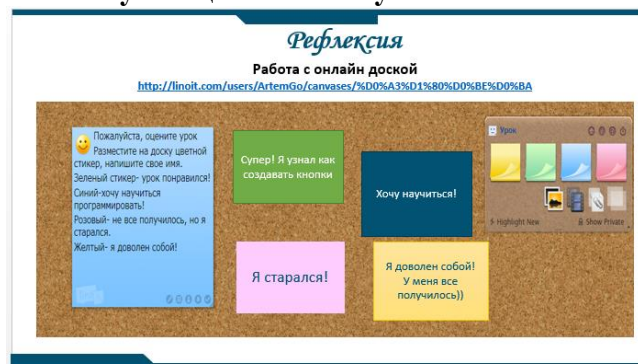


VII Рефлексия

Работа с онлайн доской

<http://linoit.com/users/ArtemGo/canvases/%D0%A3%D1%80%D0%BE%D0%BA>

Обучающиеся самостоятельно оценивают свое состояние, свои эмоции, результаты своей деятельности, знание и незнание. Размещают на онлайн доске разноцветные стикеры, соответствующие своему эмоциональному состоянию.



Подведение итогов. Выводы: Цели урока достигнуты, задачи выполнены.

Источники:

- Интернет учебник <http://study-java.ru>
- Онлайн редактор https://www.onlinegdb.com/online_java_compiler#
- МакГрат, Майк. Программирование на Java для начинающих / Майк МакГрат ; [пер. с англ. М.А. Райтмана]. – Москва: Издательство «Э», 2016. – 192 с.
- Платформа для разработки квеста <https://joyteka.com>
- Платформа для создания виртуальной доски <http://linoit.com>
- Платформа онлайн тестов TestPad <https://onlinetestpad.com/f2hamba7gktx6>

РАЗДАТОЧНЫЙ МАТЕРИАЛ

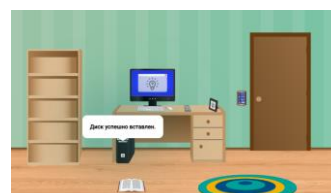
Веб- квест «Введение в программирование» <https://joyteka.com/100028565>
выполняется онлайн



Начало Квеста



*2. Убрать книги со шкафа,
отодвинуть шкаф*



*3. Взять диск из выпавшей книги,
Вставить его в системный блок*



*4. Переместить коврик,
достать зарядку на телефон*



*5 Открыть стол,
Взять наушники*



*6 Ответить на вопросы Квеста
Дверь открыта, Квест выполнили*

Вопрос веб- квеста	Ответ
Некорректная работа программы, вызванная ошибкой в программном коде или дизайне продукта	
Система условных знаков для представления информации	
Какой это язык программирования? Логотип какого языка программирования изображен на рисунке	
Набор команд, инструкций в виде кода, который запакован в исполняемый файл	
Набор инструкций, описывающих порядок действий исполнителя для достижения результата решения задачи за конечное число действий	

Онлайн ресурсы:

Виртуальная доска

<http://linoit.com/users/ArtemGo/canvases/%D0%A3%D1%80%D0%BE%D0%BA>

Облако тегов

<https://wordsccloud.pythonanywhere.com>

Онлайн тест <https://onlinetestpad.com/f2hamba7gktx6>

ТЕХНОЛОГИЧЕСКАЯ КАРТА

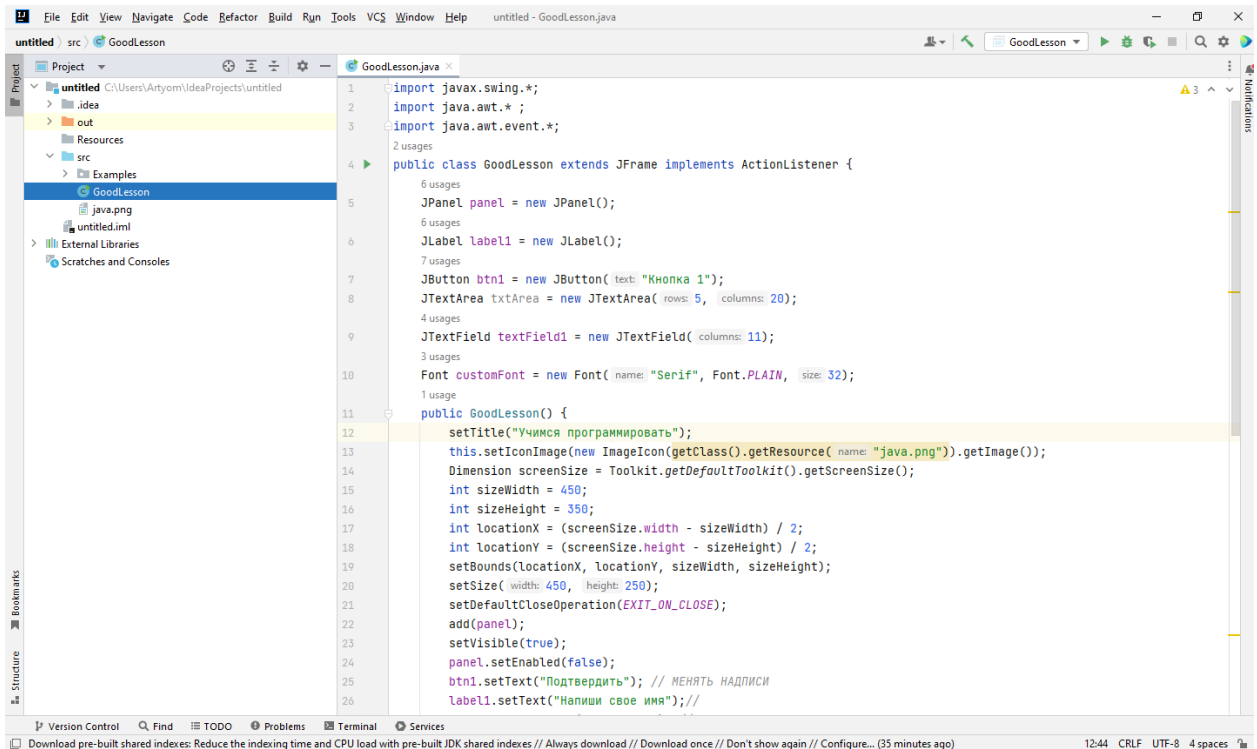
Тема урока: «Создание элемента визуального интерфейса в Java»

Цель урока:

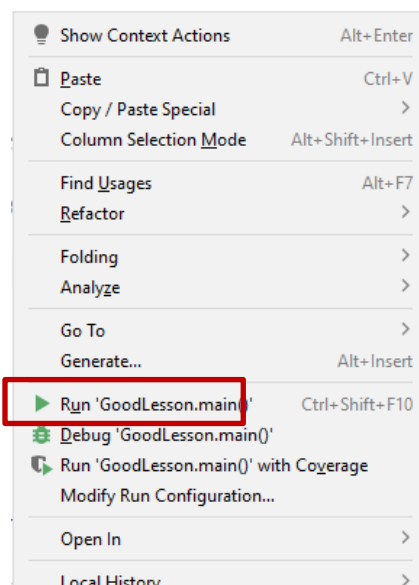
- Научиться создавать элемент визуального интерфейса;
- Научиться изменять интерфейс Кнопки JButton.

Ход работы

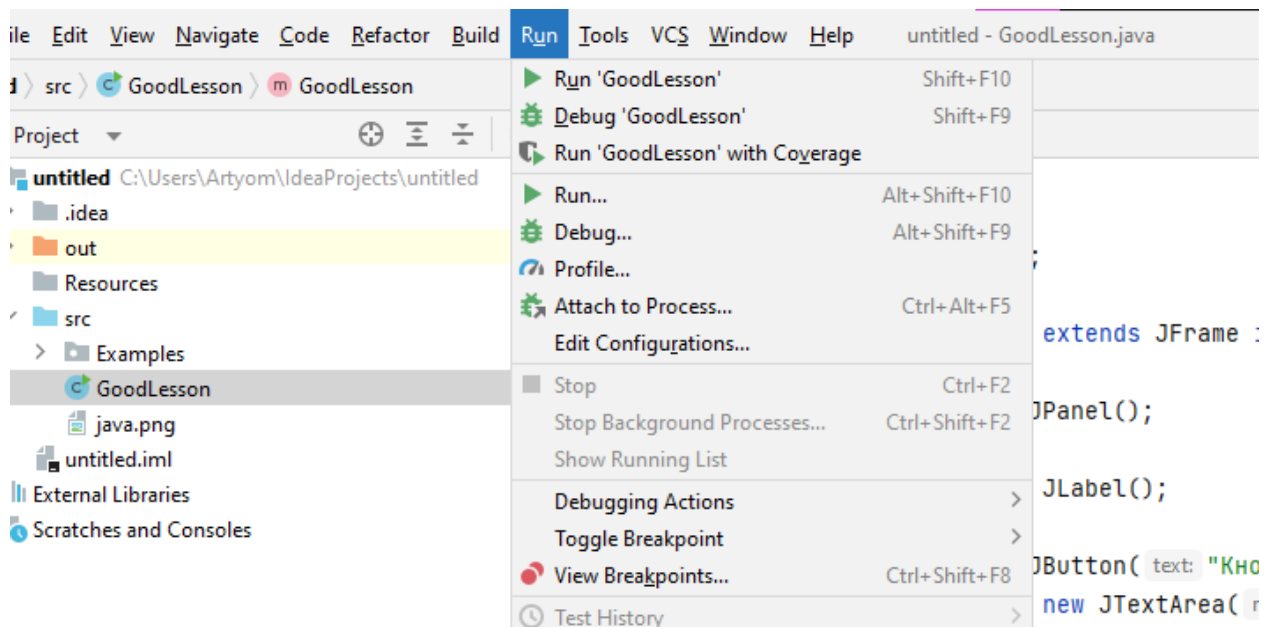
1. На Рабочем столе откройте папку Урок
2. Запустите программу IntelliJ IDEA 2 щелчка по иконке
3. Откроется проект с кодом программы, папка src файл GoodLesson



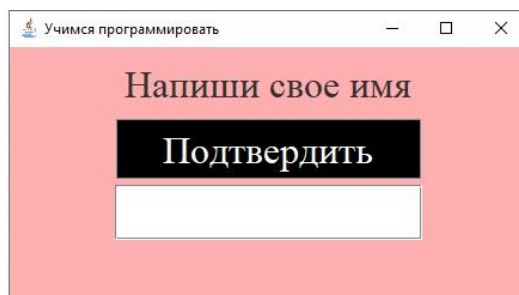
4. Запустите программу нажатием правой кнопки мыши по свободному месту и выбрать Run опцию



Или запустить через Run в меню.

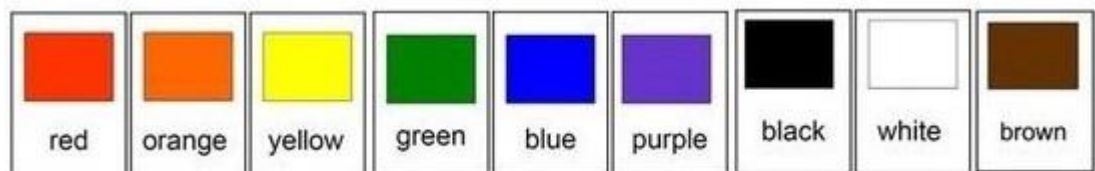


5. Подождите пока программа отобразит кнопку



6. Вернитесь в программу, для этого закройте Кнопку нажав на крестик X

7. В строках 27 измените цвет кнопки Black на цвет который выберете



8. В строках 28, 29 измените цвет заливки WHITE, PINK

```

22      add(panel);
23      setVisible(true);
24      panel.setEnabled(false);
25      btn1.setText("Подтвердить"); // МЕНЯТЬ НАДПИСИ
26      label1.setText("Напиши свое имя");//
27      btn1.setBackground(Color.BLACK); // МЕНЯТЬ ЦВЕТА
28      btn1.setForeground(Color.WHITE);
29      panel.setBackground(Color.PINK);
30      label1.setFont(customFont);
31      panel.add(label1);
32      panel.add(btn1);

```

9. Снова запустите программу



10. Посмотрите, как изменился интерфейс вашей кнопки.