Практическая работа №3.

1. Основы компьютерной графики.

1.1. **Задача:** Изучение теоретического материала. Закрепление полученных знаний путем разработки приложения на языке программирования Java в среде разработки eclipse.

1.2.**Тип занятий:** исследовательский.

1.3.**Содержание:** теоретический материал, задания для самостоятельного выполнения.

2.Теоретический материал к уроку

2.1. **Подключаемые библиотеки**

javax.swing - библиотека для создания графического интерфейса для программ на языке Java.

java.awt - содержит все классы для того, чтобы создать пользовательские интерфейсы и для рисования графики и изображений.

2.2. **Создание конструктора класса**:

пример:

public class Window{

public Window(){ //модификатор доступа public и имя такое же как у касса + круглые скобочки (внутри указываются параметры при необходимости или пусто, если параметров нет)

}

}

2.3. **Вызов конструктора класса** (создание объекта класса):

пример:

Window w = new Window();

2.4. **extends** - ключевое слово для задания наследования.

пример:

public class ourclass extends JFrame {

}

2.5. **Команды.**

**Как начертить прямую линию?**

Метод drawLine начертит прямую линию:

drawLine(20, 30, 360, 30);

здесь 20, 30 — это координаты x, y начала линии,  
360, 30 — координаты конца линии.

**Как задать цвет?**

Метод setColor сделает текущим новый цвет:  
// Устанавливаем синий цвет;  
setColor(Color.BLUE);

**Как нарисовать прямоугольник?**

Методом drawRect класса Graphics:

drawRect(20, 40, 340, 20);

20, 40 — это координаты верхнего левого угла прямоугольника;  
 340 — длина;  
 20 — высота прямоугольника.

**Как залить прямоугольник цветом?**

Методом fillRect:

fillRect(21, 41, 339, 19);

**Как нарисовать овал?**

Методом drawOval:

drawOval(20, 110, 150, 60);

Аргументы определяют прямоугольник, в который вписан овал.

**Как нарисовать окружность?**

Методом drawOval:

drawOval(200, 110, 60, 60);

Аргументы определяют прямоугольник, в который вписана окружность. Здесь рисуем овал, но длина и высота описанного прямоугольника равны, что и даёт окружность.

**Как залить окружность цветом?**

Методом fillOval:

fillOval(200, 110, 60, 60);

**Как вывести строку?**

Методом drawString:

drawString("Yes", 50, 190);

строка "Yes" будет выведена от точки с координатами 50, 190.

**Как задать шрифт?**

Для этого используем класс Font:

Font font = new Font("Tahoma", Font.BOLD|Font.ITALIC, 40);

где "Tahoma" — название шрифта,   
Font.BOLD|Font.ITALIC — жирный шрифт с наклоном,   
40 — высота шрифта.

После задания шрифта мы делаем его текущим и выводим строку этим шрифтом:

g2d.setFont(font);

**Как увеличить толщину линии?**

Для этого используем, где 5 ширина линии:

BasicStroke bs = new BasicStroke(5);

g2d.setStroke(bs);

3. Задания

3.1. уровень 1

1. Создайте новый проект с названием GameИмя, в нем создайте пакет game и в пакете главный класс Main.
2. Выведите на экран сообщение: привет из класса Main!
3. Создайте в вашем пакете второй класс Window.
4. Создайте в нем конструктор класса.
5. Выведите на экран сообщение: привет из класса Window!
6. Запустите программу на выполнение, в консоли должно выйти только первое сообщение, потому что мы еще не реализовали вызов конструктора второго класса.
7. вызовите конструктора класса Window из класса Main (создание объекта класса Window.
8. Запустите программу, теперь должны выйти оба сообщения.
9. В класс Window импортируйте библиотеку javax.swing.\*;
10. Сделайте класс Window наследуемым от класса JFrame (представляющий собой окно с рамкой и строкой заголовка (с кнопками «Свернуть», «Во весь экран» и «Закрыть»)).
11. Добавьте в конструктор класса Window следующие операторы:

//фокус на окно

setFocusable(**true**);

// Задание размеров и положения окна, первый параметр стартовая позиция //по горизонтали, вторая по вертикали, третий параметр размер в пикселях окна //по горизонтали, четвертый размер по вертикали

setBounds(0,0,800,600);

// Задание заголовка окна

setTitle("Игры");

// Запрет изменения размеров окна

setResizable(**false**);

// Операция завершения работы приложения при закрытии окна

setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

// Сделать окно видимым

setVisible(**true);**

1. Запустите программу на выполнение у вас должно выйти пустое окно.
2. Добавьте в проект класс DrawSquare без метода main и импортируйте в него библиотеки javax.swing.\* и java.awt.\* Сделайте его наследуемым от класса JPanel (служит для рисования).
3. Вставьте следующий метод в класс, импортируя все предлагаемые библиотеки:

/\*Метод, перерисовывающий элемент внутри окна

\*при обновлении\*/

**public** **void** paintComponent(Graphics g){

**super**.paintComponents(g);

Graphics2D g2d=(Graphics2D)g;

/\* Устанавливает цвет рисования в зелёный\*/

g2d.setColor(Color.***GREEN***);

/\* Рисует текущим цветом прямоугольник \*/

g2d.drawRect(100, 20, 400, 120);

/\* Устанавливает цвет рисования в красный\*/

g2d.setColor(Color.***RED***);

/\* Рисует текущим цветом строку "привет мир"\*/

g2d.drawString("Привет мир", 100, 230);

g2d.setColor(Color.***BLUE***);

/\* Рисует текущим цветом овал в координатах\*/

g2d.fillOval(200, 150, 250, 100);

/\* Вызывает обновление себя после завершения рисования \*/

**super**.repaint();

}

8) Теперь в конструктор класса Window добавьте вызов методов из класса DrawSquare, сделать это нужно до того, как устанавливаете видимость окна

DrawSquare d= **new** DrawSquare();

add(d);

9) Запустите программу на выполнение, посмотрите что у вас получилось.

10) Выведите на форме свое имя.

11) сделайте оранжевый закрашенный квадрат.

12) сделайте серый овал в правом верхнем углу формы, чтобы он не выходил за границы.

10) Сделайте эмблему олимпийских игр в вашем окне, команда, чтобы отрисовать линию толще см. в теории 