

# IT ШКОЛА SAMSUNG

Модуль 1. Основы программирования

Урок 13-14. Цикл for

SAMSUNG

## while

## for

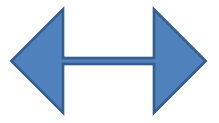
```
i = 0; //инициализация счетчика  
while (i < N){ //условие цикла  
    //... тело цикла  
    i++; //приращение счетчика  
}
```

```
for (i = 0; i < N ; i ++){  
    //... тело цикла  
}
```



Начальную инициализацию и приращение счетчика очень легко забыть написать

```
p = 1;
for (; p <= 1000000; ){
    p *= 3;
}
```



```
for (p = 1; p <= 1000000; p *= 3){
}
```

```
for (int i = 10; i > 0; i--){
    out.println(i);
}
out.println(i);
```

```
for (int i = 0; i < N; i++)
{
    for (int j = 0; j < N; j++)
    {
        out.print("*");
    }
    out.println();
}
```

**!** Программа не скомпилируется:  
после цикла переменная **i**  
**НЕ СУЩЕСТВУЕТ**

**!** Выход из вложенных циклов, в языке Java реализовать можно при помощи оператора **break** с меткой.

**outer:**

```
for (int i = 0; i < 3; i++) {  
    System.out.print("Итерация " + i + ": ");  
    for (int j = 0; j < 100; j++) {  
        if (j == 10) {  
            break outer; // ВЫЙТИ ИЗ ОБОИХ ЦИКЛОВ  
        }  
        out.print(j + " ");  
    }  
    out.println("Эта строка никогда не будет выведена");  
}
```



В случае, если нужно продолжить выполнение цикла, но прекратить обработку остатка кода в его теле для данной итерации поможет **continue**

**Задача:** вывести числа от 0 до 10 в два столбика

```
for (int i = 0; i < 10; i++) {  
    System.out.print(i + " ");  
    if (i % 2 == 0) {  
        continue;  
    }  
    out.println();  
}
```

Оператор **continue** срабатывает на каждом четном  $i$  и перевод строки **не** происходит

## Некоторые методы класса **PAINT**

<b>setColor(int color)</b>	Установка цвета
<b>setStrokeWidth(float width)</b>	Установка толщины строки
<b>setStyle(Paint.Style style)</b>	Установка способа заливки

## Возможные значения **Paint.Style**:

<b>Paint.Style.FILL</b>	рисование с заливкой
<b>Paint.Style.STROKE</b>	рисование только контура
<b>Paint.Style.FILL_AND_STROKE</b>	рисование и контура и заливки

1. Создать объект класса Paint:

```
Paint paint = new Paint();
```

2. Настроить его:

```
paint.setColor(Color.GREEN);  
paint.setStrokeWidth(5);
```

3. Выполнить рисование

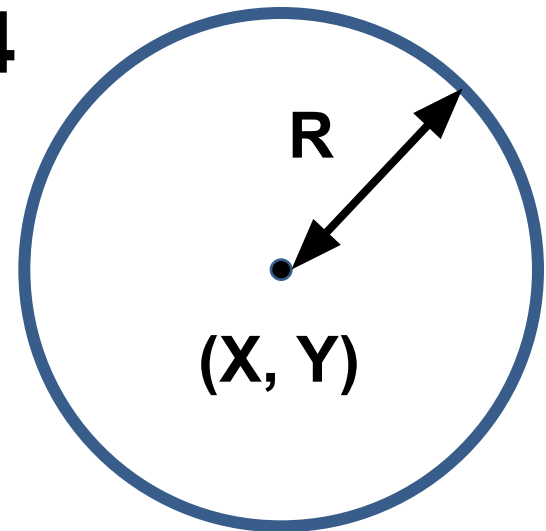
```
canvas.drawLine(25 , 25 , 50 , 50 , paint);
```

В классе *Canvas* реализовано много различных методов рисования кроме рисования линий.

**drawCircle** (float **cx**, float **cy**, float **radius**, Paint **paint**)

Круг с центром в точке (40,40) и радиусом 34

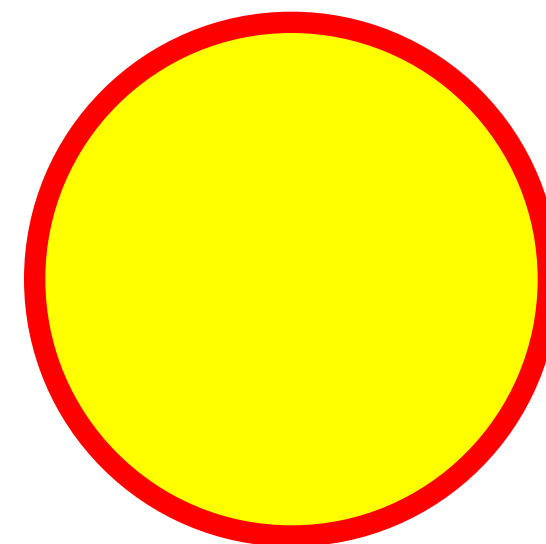
```
drawCircle (40, 40, 34, paint);
```





**Задача:** нарисовать желтый круг с красным утолщенным контуром

```
Paint paint = new Paint();  
paint.setColor(Color.YELLOW);  
paint.setStyle(Paint.Style.FILL);  
canvas.drawCircle(300, 300, 200, paint);  
paint.setColor(Color.RED);  
paint.setStyle(Paint.Style.STROKE);  
paint.setStrokeWidth(20);  
canvas.drawCircle(300, 300, 200, paint);
```

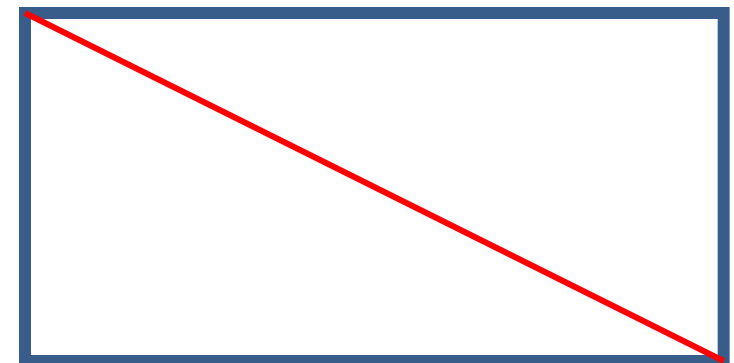


**drawRect (float left, float top, float right, float bottom, Paint paint)**

**Прямоугольник с вершинами диагонали  
в точках (40,40) и (100,100)**

**drawRect (40, 40, 100, 100, paint);**

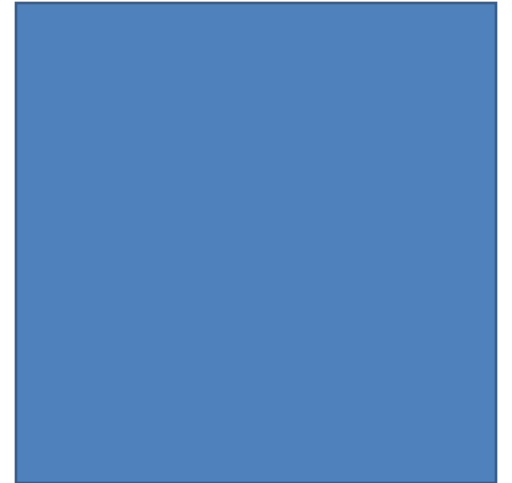
(40, 40)



(100, 100)

**Задача:** нарисовать синий квадрат:

```
Paint paint = new Paint();  
paint.setColor(Color.BLUE);  
paint.setStyle(Paint.Style.FILL);  
canvas.drawRect(100, 100, 300, 300, paint);
```



**Задача:** нарисовать столбик из прямоугольников:

```
Paint paint = new Paint();
paint.setColor(Color.BLUE);
paint.setStyle(Paint.Style.STROKE);
for (int y = 0; y < 500; y += 60)
{
    paint.setStyle(Paint.Style.FILL);
    canvas.drawRect(100, y + 10, 300, y + 50, paint);
}
```



**Paint создается и настраивается один раз до цикла. Если разместить первые три строки кода в теле цикла, это замедлит выполнение программы: каждый раз, на каждой итерации объект `paint` будет создаваться, настраиваться и... сразу разрушаться.**

1. Импортировать проект Android под именем **ProjectCanvas**
2. В классе **MyDraw** в теле метода **onDraw** прописываем код

**Задача 1:** Нарисуйте квадрат, центр которого совпадает с центром экрана.

**Задача 2:** Нарисуйте 4 окружности с вершинами в углах экрана

**Задача 3:** Нарисуйте мишень из 10 колец

**Задача 4:** Разлините экран Android-приложения следующим образом: слева горизонтальные линии, справа вертикальные

**Задача 5:** Придумайте и реализуйте собственный узор

1. Выполните задания урока.
2. Решите задачи 117, 216, 346 и 1437.

# Спасибо!

В презентации использованы материалы Ильина В.В.  
и элементы оформления презентации Деникиной Н.

The Samsung logo, consisting of the word "SAMSUNG" in white capital letters inside a blue oval shape.

SAMSUNG