

//КЛАС Position

```
public class Position {
    private int x,y;
    public Position(){
        x=0;
        y=0;
    }
    public Position(int x,int y){
        this.x=x;
        this.y=y;
    }
    public int getX(){
        return x;
    }
    public int getY(){
        return y;
    }
    public String toString(){
        return "("+x+", "+y+")";
    }
}
```

//КЛАС Point, НАСЛЕДНИК НА Position

```
import java.awt.Color;//Клас за цвят (RGB)
//Библиотека за рисуване
import org.eclipse.swt.graphics.*;
import org.eclipse.swt.widgets.Display;//Клас Display
public class Point extends Position{
    private Color color;//Допълнително свойство "цвят"
    //Конструктор по позиция и цвят
    public Point(int x,int y,Color c){
        super(x,y);//Повикване на наследения конструктор
        color=c;//Инициализиране на цвета
    }
    //Предефиниране на метода toString
    @Override
    public String toString(){
        return "["+super.toString()+", "+color+"]";
    }
    //Дефиниране на нов метод за извеждане върху обект от тип Canvas
    public void show(org.eclipse.swt.widgets.Canvas canvas){
        GC g=new GC(canvas);//Графичен контекст на обекта canvas
        //Установяване на цвета за запълване
        //-превърщаме цвета от java.awt.Color в org.eclipse.swt.graphics.Color
        g.setBackground(new org.eclipse.swt.graphics.Color(Display.getCurrent(),
            color.getRed(),color.getGreen(),color.getBlue()));
        //Изобразяване на точката
        g.fillOval(getX(), getY(), 4, 4);
        g.dispose();//Освобождаване на ресурса g
    }
}
```

//КЛАС Test, ЗА ТЕСТВАНЕ В КОНЗОЛЕН РЕЖИМ

```
import java.awt.Color;
//import org.eclipse.swt.*;
public class Test {
    public static void main(String[] args) {
        Point p=new Point(1,2,new Color(0,0,0));
        System.out.println(p.toString());
    }
}
```

```

}
}
//КЛАС VisualTest, ЗА ТЕСТВАНЕ В ГРАФИЧЕН РЕЖИМ
//SWT Java Project
import java.awt.Color;//Клас за цвят (RGB)
//Автоматично включени библиотеки
import org.eclipse.swt.*;
import org.eclipse.swt.widgets.*;
public class VisualTest {
//canvas, изнесена като частно свойство
private org.eclipse.swt.widgets.Canvas canvas;
protected Shell shell;
//Автоматично създаден метод main
public static void main(String[] args) {
try {
VisualTest window = new VisualTest();
window.open();
} catch (Exception e) {
e.printStackTrace();
}
}
//Автоматично създаден метод за обслужване на събитието open на фóрмата
public void open() {
Display display = Display.getDefault();
createContents();
shell.open();
shell.layout();
//Тук вече можем да повикаме метода за тестване
test();

//Автоматично генериран основен цикъл на апликацията
while (!shell.isDisposed()) {
if (!display.readAndDispatch()) {
display.sleep();
}
}
}
//Автоматично създаден метод за инициализация на фóрмата
protected void createContents() {
shell = new Shell();
shell.setSize(450, 300);
shell.setText("Test objects");
//canvas е изнесена като частно свойство,
//за да могат други методи да имат достъп до нея.
canvas = new org.eclipse.swt.widgets.Canvas(shell, SWT.NONE);
canvas.setBounds(0, 10, 422, 233);
}
//Метод за тестване на графичното представяне на обекти от класове
private void test(){
Point p=new Point(canvas.getBounds().width/2,
canvas.getBounds().height/2,
new Color(255,0,0));

p.show(canvas);
}
}

```