

Цикл for, графика.

Графические команды и общие принципы рисования на канве я решил не переписывать, так как их довольно много, и при этом они хорошо описаны на различных сайтах, например [здесь](#).

```
procedure TForm1.Button1Click(Sender: TObject);
var a,b,i:integer;
begin
    ...
    a:=...
    b:=...
    ...
    for i:=a to b do
    begin
        //набор команд
    end;
    ...
end.
```

Рис. 1: Синтаксис цикла for.

Цикл **for** пишется так, как показано на рис. 1. Он позволяет многократно выполнить набор команд, находящийся между командами **begin** и **end;**. Этот набор команд называют *телом цикла*, каждое их выполнение - *итерацией цикла*. Такой цикл выполняется $(b - a + 1)$ раз; на первой итерации $i = a$, на второй $i = a + 1$, ..., на последней $i = b$. Переменная i называется *счётчиком цикла* и её нельзя изменять в теле цикла.

P.S. Если нужно многократно выполнить всего одну команду, то **begin** и **end;** можно не писать. Кстати говоря это общее правило для всех циклов, а так же (хотя это и не упоминалось ранее) - и для условного оператора.

Домашнее задание: написать программу, которая рисует на канве решётку из капель (пока что неподвижную). Каждая капля рисуется как маленький эллипс или линия (хотя можно использовать и более сложную форму). Строка какпель рисуется циклом, содержащим рисование капли, вся решётка - циклом, содержащим цикл рисования строки.

P.S. По не помню чьей просьбе - на следующей странице добавлен разбор домашнего задания.

Если добавить на канву **Image1** и **Button1**, то по нажатию кнопки должна выполняться такая процедура:

```
procedure TForm1.Button1Click(Sender: TObject);
var i,j,x,y:integer;
begin
  with image1.Canvas do
    begin

      brush.color:=clWhite;
      rectangle(-5,-5,5000,5000);
      brush.color:=clBlue;
      pen.color:=clBlue;

      for i:=1 to 20 do
        begin
          y:=i*10;
          for j:=1 to 30 do
            begin
              x:=j*10;
              ellipse(x,y,x+5,y+7);
            end;
          end;
        end;

      end;
    end;
```