

Iseng–Iseng dengan Processing

Penulis: Auzi Asfarian

Iseng-Iseng dengan Processing

Pengantar

Sesi Koding

Bedah Kode

Deklarasi Variabel

Fungsi `Setup()`

Fungsi `Draw()`

Pengantar

“*Programming* itu menyenangkan!” adalah kalimat motivasi para pejuang-pejuang Ilmu Komputer saat kuliah. Yah, tapi kenyataannya pasti ada saat-saat kita merasa bosan berkutat dengan kode program, apalagi saat kode yang kita buat hanya menampilkan program dalam bentuk *command line interface* alias teks putih dengan latar belakang hitam. Agar *programming* tetap menyenangkan, kadang baik untuk kita mempelajari bahasa pemrograman yang lain. Dalam hal ini, kita mempelajari tentang Processing yang keluaran programnya berupa grafis 😊.

Sesi Koding

Untuk percobaan pertama ini, kita akan membuat sebuah program yang akan terus-menerus mengeluarkan pola bunga (*lihat gambar*). Bunganya tidak terlalu rumit, hanya terdiri atas lima buah lingkaran yang ditumpuk sedemikian rupa. Untuk sedikit memperindah hasil akhir program, akan kita buat supaya ukuran dan warna bunga yang dihasilkan berbeda-beda.



Inilah langkah-langkah untuk merealisasikan program yang telah kita bayangkan sebelumnya :

1. Download Processing.
2. Ekstraksi file yang telah anda unduh sebelumnya.
3. Buka Processing.
4. Ketikkan kode berikut di lembar kerja Processing :

```
//Parameter
final int SCREEN_HEIGHT = 320;
final int SCREEN_WIDTH = 640;
final int MIN_DIAMETER = 0;
final int MAX_DIAMETER = 10;
final int MIN_RED = 100;      //0 - 255
final int MAX_RED = 250;      //0 - 255
final int MIN_GREEN = 100;    //0 - 255
final int MAX_GREEN = 250;    //0 - 255
final int MIN_BLUE = 0;       //0 - 255
final int MAX_BLUE = 0;       //0 - 255

//Pengaturan program, dijalankan satu kali
//saat program pertama kali berjalan
void setup()
{
    size(SCREEN_WIDTH,SCREEN_HEIGHT); //Ukuran Layar
    noStroke();      //Menghilangkan garis tepi
    background(0); //Set latar belakang menjadi warna hitam
    smooth();
}

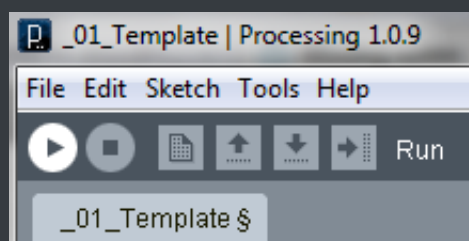
//Fungsi untuk menggambar objek di layar.
//Dipanggil terus-menerus sampai program berhenti.
void draw()
{
    float r,g,b,a; //Warna
    float d,x,y;   //Posisi

    //Bangkitkan satu nilai acak
    r = random(MIN_RED,MAX_RED);
    g = random(MIN_GREEN,MAX_GREEN);
    b = random(MIN_BLUE,MAX_BLUE);
    a = random(0,155);
    d = random(MIN_DIAMETER,MAX_DIAMETER);
    x = random(0,width);
    y = random(0,height);

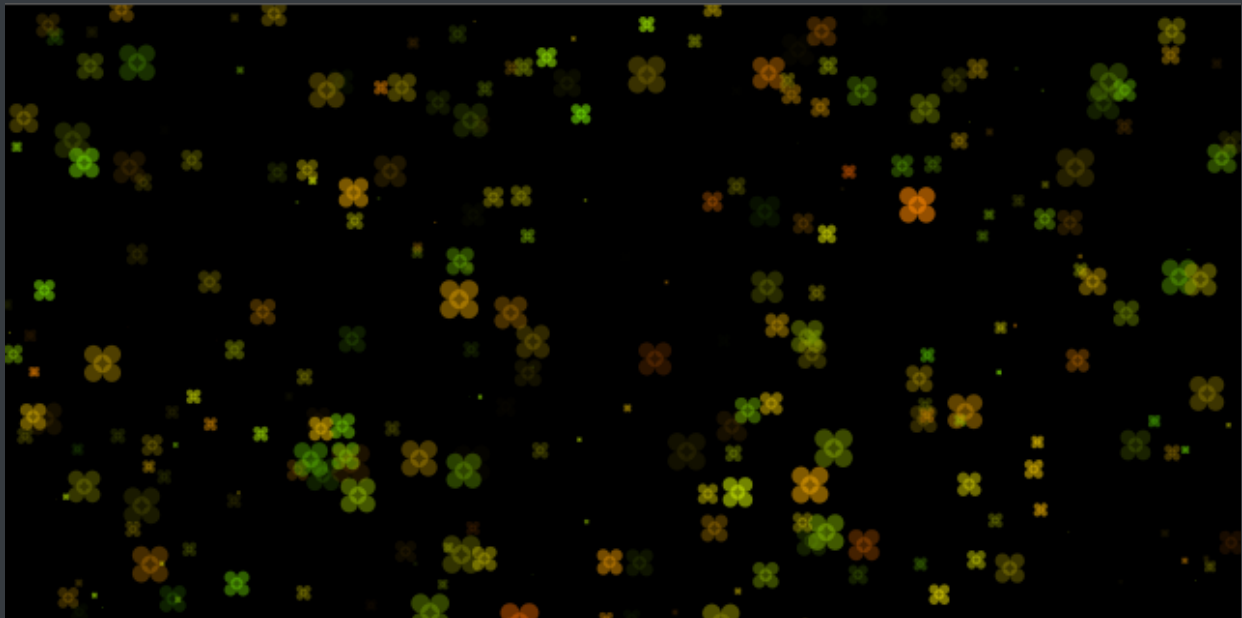
    fill(r,g,b,a); //Set warna bunga
```

```
//Gambar lingkaran
//Lima lingkaran dibentuk mirip dengan bunga
ellipse(x, y, d, d);
ellipse(x+d/2, y-d/2, d, d);
ellipse(x+d/2, y+d/2, d, d);
ellipse(x-d/2, y+d/2, d, d);
ellipse(x-d/2, y-d/2, d, d);
}
```

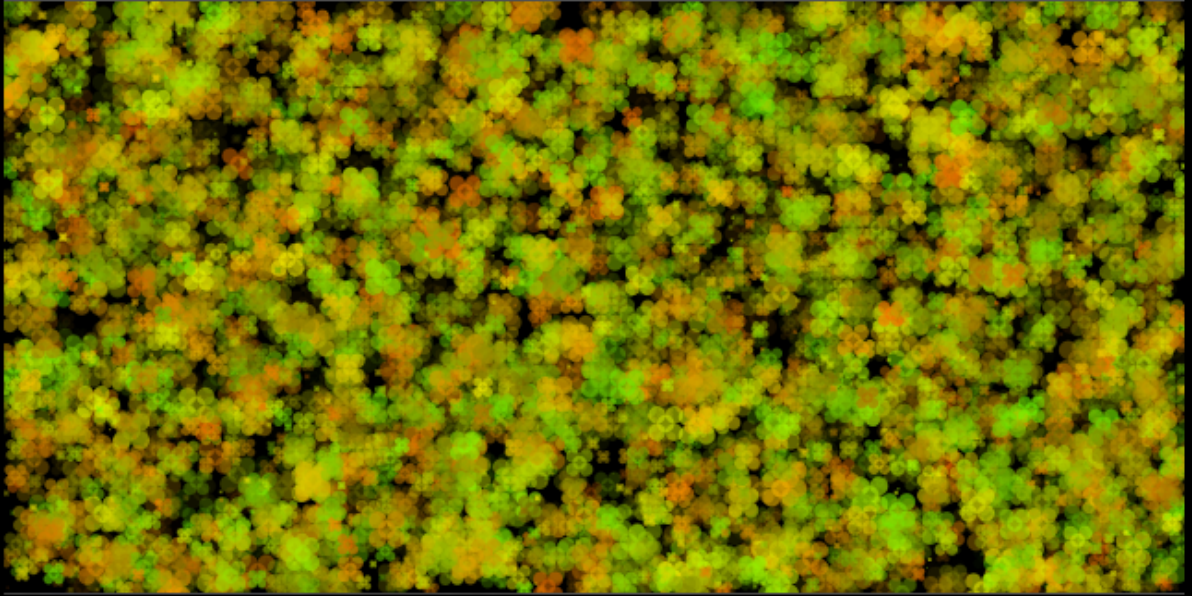
Setelah itu, untuk menjalankan program yang telah dibuat, Anda cukup menekan tombol panah yang ada di pojok kiri atas. Untuk menghentikan program, tekan tombol kotak yang ada di sebelah tombol panah. Mirip lah dengan menjalankan lagu di Windows Media Player 😊.



Hasilnya?



Setelah satu detik:



Bedah Kode

Deklarasi Variabel

Bagian pertama kode di atas berisi nilai-nilai yang digunakan pada program yang kita buat. Nilai-nilai tersebut tentu saja dapat kita ubah, untuk mendapatkan hasil yang lebih menarik. Dua nilai pertama menyatakan ukuran layar dari program yang dihasilkan.

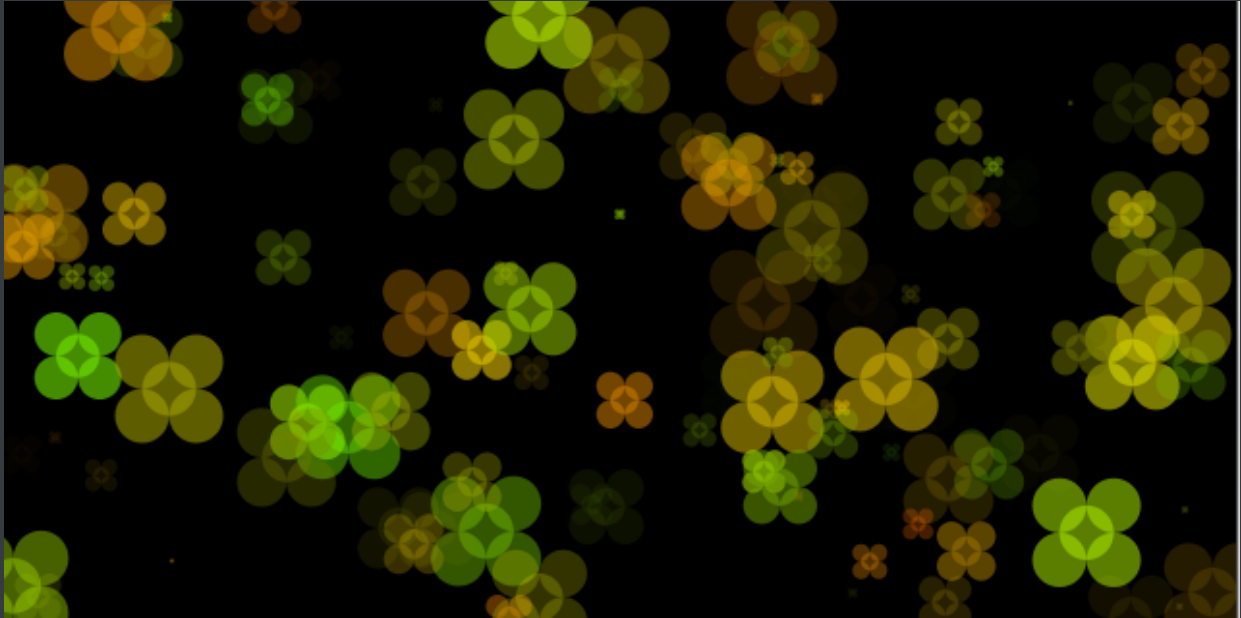
```
final int SCREEN_HEIGHT = 320;  
final int SCREEN_WIDTH = 640;
```

Dua nilai selanjutnya menyatakan ukuran minimum dan maksimum bunga yang akan ditampilkan. Semakin besar nilainya, semakin besar bunga yang digambar.

```
final int MIN_DIAMETER = 0;  
final int MAX_DIAMETER = 10;
```

Nilai sisanya mempengaruhi warna bunga yang akan dihasilkan. Pengaturan seperti di bawah saya pilih agar warna bunga yang dihasilkan sewarna dengan warna daun di musim gugur 🍂.

```
final int MIN_RED = 100;      //0 - 255
final int MAX_RED = 250;      //0 - 255
final int MIN_GREEN = 100;    //0 - 255
final int MAX_GREEN = 250;    //0 - 255
final int MIN_BLUE = 0;       //0 - 255
final int MAX_BLUE = 0;       //0 - 255
```



Fungsi `Setup()`

Bagian kedua program adalah fungsi `setup()`. Fungsi ini akan dijalankan tepat satu kali pada saat program pertama kali berjalan. Fungsi ini biasanya berisi pengaturan umum dari program, misalnya ukuran layar.

```
//Pengaturan program, dijalankan satu kali
//saat program pertama kali berjalan
void setup()
{
    size(SCREEN_WIDTH,SCREEN_HEIGHT); //Ukuran Layar
    noStroke();      //Menghilangkan garis tepi
    background(0); //Set latar belakang menjadi warna hitam
    smooth();
}
```

Fungsi `Draw()`

Bagian terakhir adalah fungsi `draw()` . Sesuai namanya, fungsi ini digunakan untuk menggambar objek di layar. Fungsi `draw()` akan dipanggil secara terus-menerus selama program masih berjalan. Hasil dari fungsi `draw()` adalah sebuah *frame*. Mungkin anda masih ingat istilah *frame per second*. Bila program berjalan dengan 30 *frame per second*, berarti dalam satu detik fungsi `draw()` dipanggil sebanyak tiga puluh kali. Di bagian berikut, kita bangkitkan nilai-nilai acak untuk memberikan variasi warna dan ukuran pada bunga yang digambar.

```
...
//Bangkitkan satu nilai acak
r = random(MIN_RED,MAX_RED);
g = random(MIN_GREEN,MAX_GREEN);
b = random(MIN_BLUE,MAX_BLUE);
a = random(0,155);
d = random(MIN_DIAMETER,MAX_DIAMETER);
x = random(0,width);
y = random(0,height);
...
```

Dengan menggunakan fungsi `fill()` , kita memberitahu program untuk menggambar sesuatu dengan warna yang telah dibangkitkan tadi.

```
...
fill(r,g,b,a); //Set warna bunga
...
```

Terakhir, kita gambar lima buah lingkaran sesuai dengan pola yang kita buat.

```
...
//Gambar lingkaran
//Lima lingkaran dibentuk mirip dengan bunga
ellipse(x, y, d, d);
ellipse(x+d/2, y-d/2, d, d);
ellipse(x+d/2, y+d/2, d, d);
ellipse(x-d/2, y+d/2, d, d);
ellipse(x-d/2, y-d/2, d, d);
...
```

Selesai, selamat mencoba!. Semoga bisa dibuat kelanjutan tulisannya.

Ketika nilai $\int_0^{\infty} e^{-x} dx$ mencapai titik maksimum, nilai x akan....

$$\int_{\Omega} \nabla u \cdot \nabla v \, dx = \int_{\Omega} f v \, dx$$

Also check out this [LaTeX introduction](#).

```
public static void main(){  
    writeln("hello world");  
}
```

```
public static void main(){  
    a  
}
```