

# **LAPORAN RESMI GRAFIKA KOMPUTER**

## **PERTEMUAN IV PENGENALAN VARIABEL, FUNGSI DAN ANIMASI P5.JS**



**NIM : 21104410019**  
**NAMA : Kharismatul Yogi Nafi'ah**  
**JURUSAN : Teknik Informatika**  
**KELAS : TI 4A**

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS ISLAM BALITAR  
2023**

## **LEMBAR PERSETUJUAN**

# **PERTEMUAN IV PENGENALAN VARIABEL, FUNGSI DAN ANIMASI P5.JS**

<b>NIM</b>	<b>: 21104410019</b>
<b>NAMA</b>	<b>: Kharismatul Yogi Nafi'ah</b>
<b>JURUSAN</b>	<b>: Teknik Informatika</b>
<b>KELAS</b>	<b>: TI 4A</b>

**Disetujui,  
Blitar, 24 Mei 2023**

**Dosen**

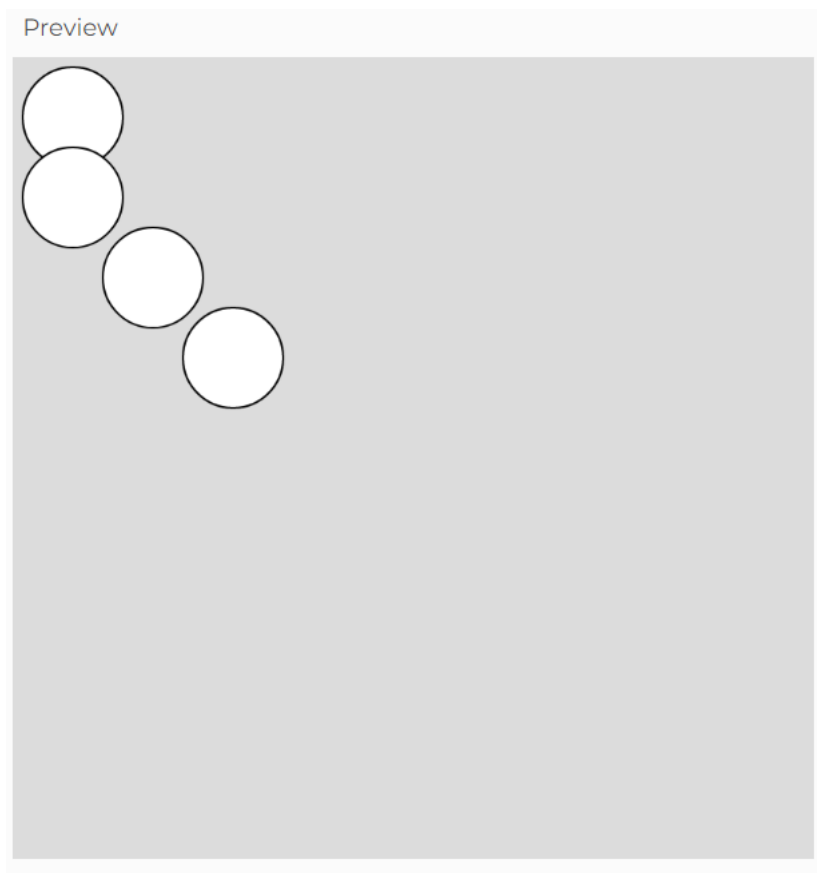
**Mohammad Faried Rahmat, S.ST., M.Tr.T**

## 1. Praktik 1 Membuat Variabel

### Source code:

```
//membuat variable  
let lebarLingkaran = 50;  
  
function setup() {  
  createCanvas(400, 400);  
}  
function draw() {  
  background(220);  
  circle(30, 30, lebarLingkaran);  
  circle(30, 70, lebarLingkaran);  
  circle(70, 110, lebarLingkaran);  
  circle(110, 150, lebarLingkaran);  
}
```

### Output:



### Penjelasan code:

1. Variabel lebarLingkaran digunakan untuk menyimpan ukuran lebar lingkaran.

2. Fungsi `setup()` adalah fungsi yang wajib ada dalam `p5.js` dan akan dipanggil satu kali pada awal program. Dalam fungsi ini, kita membuat kanvas dengan lebar dan tinggi 400 piksel menggunakan `createCanvas(400, 400)`. Ini akan membuat kanvas berukuran 400 piksel x 400 piksel.
3. Fungsi `draw()` dipanggil secara terus-menerus setelah fungsi `setup()`. Fungsi ini menggambar ulang kanvas pada setiap frame. Pada fungsi ini:
4. `background(220)` mengatur warna latar belakang kanvas menjadi abu-abu muda (RGB 220) pada setiap frame, sehingga memberikan efek pemutaran ulang animasi.
5. `circle(30, 30, lebarLingkaran)`, `circle(30, 70, lebarLingkaran)`, `circle(70, 110, lebarLingkaran)`, dan `circle(110, 150, lebarLingkaran)` adalah pemanggilan fungsi `circle()` yang digunakan untuk menggambar lingkaran dengan ukuran `lebarLingkaran` pada posisi yang berbeda di kanvas. Setiap pemanggilan fungsi `circle()` menghasilkan lingkaran pada posisi (x, y) dengan ukuran yang sama.
6. Sebagai hasilnya, kode tersebut menghasilkan animasi dengan latar belakang yang berputar-putar dan menampilkan empat lingkaran dengan ukuran yang sama tetapi berbeda posisi di kanvas. Lingkaran-lingkaran tersebut akan tetap ada pada posisi yang ditentukan karena ukuran dan posisi lingkaran diatur dengan menggunakan variabel `lebarLingkaran`.

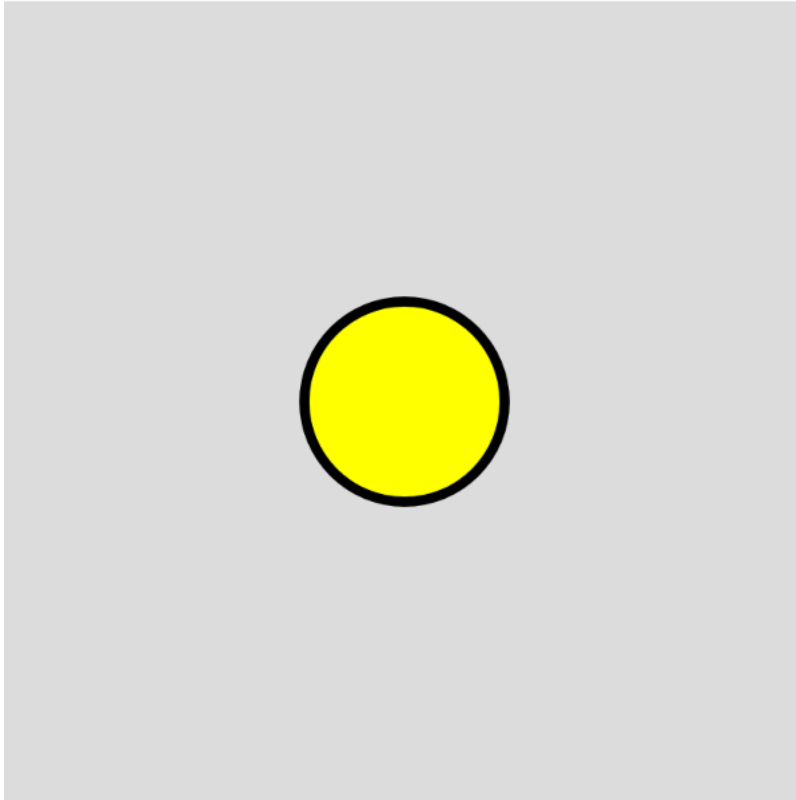
## 2. Praktik 2 Fungsi Pada P5.JS

### Source code:

```
function setup() {  
  createCanvas(400, 400);  
}  
  
function draw() {  
  background(220);  
  lingkaran();  
}  
  
function lingkaran() {  
  fill(255, 255, 0);  
  strokeWeight(5);  
  circle(200, 200, 100);  
}
```

### Output:

Preview



**Penjelasan code:**

1. Fungsi `setup()` adalah fungsi yang wajib ada dalam p5.js dan akan dipanggil satu kali pada awal program. Dalam fungsi ini, kita membuat kanvas dengan lebar dan tinggi 400 piksel menggunakan fungsi `createCanvas(400, 400)`. Ini akan membuat kanvas berukuran 400 piksel x 400 piksel.
2. Fungsi `draw()` dipanggil secara terus-menerus setelah fungsi `setup()`. Fungsi ini menggambar ulang kanvas pada setiap frame. Pada fungsi ini:
3. `background(220)` mengatur warna latar belakang kanvas menjadi abu-abu muda (RGB 220) pada setiap frame, sehingga memberikan efek pemutaran ulang animasi.
4. `lingkaran()` adalah pemanggilan fungsi `lingkaran()`. Ini memanggil fungsi `lingkaran()` yang telah didefinisikan di bawahnya.
5. Fungsi `lingkaran()` adalah fungsi yang didefinisikan oleh pengguna. Fungsi ini digunakan untuk menggambar lingkaran kuning di tengah kanvas. Pada fungsi ini:
6. `fill(255, 255, 0)` mengatur warna isi lingkaran menjadi kuning (RGB 255, 255, 0).
7. `strokeWeight(5)` mengatur ketebalan garis lingkaran menjadi 5 piksel.
8. `circle(200, 200, 100)` menggambar lingkaran di posisi (200, 200) dengan radius 100 piksel. Ini menghasilkan lingkaran yang terletak di tengah kanvas.
9. Sebagai hasilnya, kode tersebut menghasilkan animasi dengan latar belakang yang berputar-putar dan menampilkan sebuah lingkaran kuning di tengah kanvas. Lingkaran tersebut akan

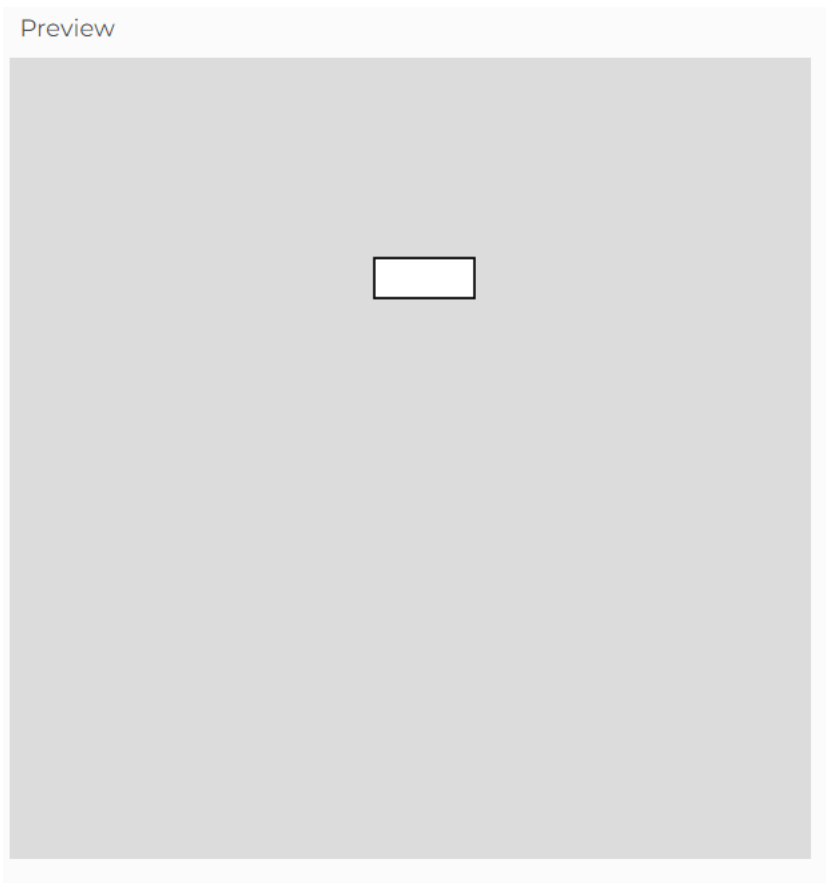
tetap ada di tengah karena fungsi `lingkaran()` dipanggil pada setiap frame dalam fungsi `draw()`.

### 3. Praktik 3 Animasi Pada P5.JS

#### Source code:

```
let sumbuX = 0;
function setup() {
  createCanvas(400, 400);
}
function draw() {
  background(220);
  sumbuX++;
  rect(sumbuX, 100, 50, 20);
}
```

#### Output:



#### Penjelasan code:

1. Variabel `sumbuX` diinisialisasi dengan nilai 0. Variabel ini merepresentasikan posisi horizontal persegi panjang.

2. Fungsi `setup()` adalah fungsi yang wajib ada dalam `p5.js` dan akan dipanggil satu kali pada awal program. Dalam fungsi ini, kita membuat kanvas dengan lebar dan tinggi 400 piksel menggunakan fungsi `createCanvas()`.
3. Fungsi `draw()` dipanggil secara terus-menerus setelah fungsi `setup()`. Fungsi ini mengupdate dan menggambar ulang kanvas pada setiap frame. Pada fungsi ini:
4. `background(220)` mengatur warna latar belakang kanvas menjadi abu-abu muda (RGB 220).
5. `sumbuX++` menambahkan 1 pada variabel `sumbuX` setiap frame, sehingga persegi panjang bergerak secara horizontal.
6. `rect(sumbuX, 100, 50, 20)` menggambar persegi panjang pada posisi `sumbuX` saat ini (yang terus bertambah karena penambahan). Persegi panjang memiliki lebar 50 piksel dan tinggi 20 piksel. Fungsi `rect()` digunakan untuk menggambar persegi panjang pada kanvas.
7. Sebagai hasilnya, kode tersebut menghasilkan animasi dimana persegi panjang bergerak secara horizontal di atas kanvas dari kiri ke kanan. Kecepatan pergerakan ditentukan oleh penambahan yang diterapkan pada variabel `sumbuX`.

#### 4. Praktik 4 Bouncing Ball

##### Source code:

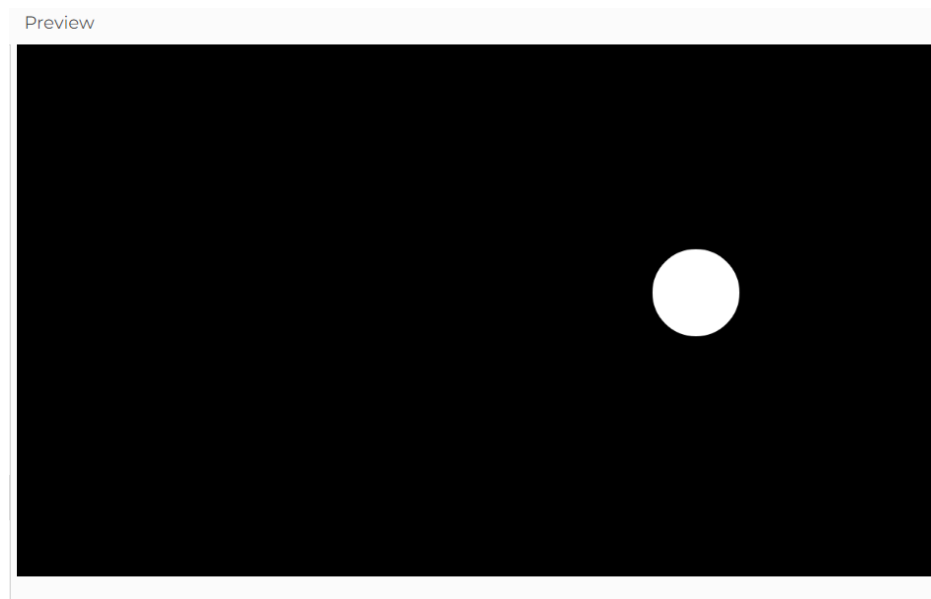
```
//variabel ukuran area bola memantul
let x = 320;
let y = 180;
//atur speed
let xspeed = 20;
let yspeed = 12;
//ukuran bola
let r = 30;
function setup() {
  createCanvas(640, 360);
}
function draw() {
  background(0);
  ellipse(x, y, r*2, r*2);
  x += xspeed;
  y += yspeed;
  if(x > width - r || x < r) {
```

```

        xspeed = -xspeed;
    }
    if(y > height - r || y < r) {
        yspeed = -yspeed;
    }
}

```

### Output :



### Penjelasan code:

1. Variabel x dan y digunakan untuk menyimpan posisi bola di koordinat horizontal dan vertikal.
2. Variabel xspeed dan yspeed menentukan kecepatan bola dalam pergerakan horizontal dan vertikal.
3. Variabel r menyimpan radius bola.
4. Fungsi setup() adalah fungsi yang dijalankan hanya sekali pada awal program. Di sini, fungsi createCanvas() digunakan untuk membuat area gambar dengan lebar 640 piksel dan tinggi 360 piksel.
5. Fungsi draw() adalah fungsi yang dijalankan secara berulang kali setelah fungsi setup() selesai.
6. background(0); digunakan untuk menggambar latar belakang dengan warna hitam (RGB 0).
7. ellipse(x, y, r\*2, r\*2); digunakan untuk menggambar bola dengan menggunakan fungsi ellipse() pada posisi (x, y). Karena r adalah radius, maka diameter bola adalah r\*2.
8. x += xspeed; dan y += yspeed; digunakan untuk menggerakkan bola dengan menambahkan kecepatan horizontal (xspeed) ke variabel x dan kecepatan vertikal (yspeed) ke variabel y.
9. if (x > width - r || x < r) { xspeed = -xspeed; } dan if (y > height - r || y < r) { yspeed = -yspeed; } digunakan untuk memantulkan bola ketika bola mencapai batas area gambar. Jika posisi



bola melebihi batas kanan atau kiri (lebih besar dari  $\text{width} - r$  atau lebih kecil dari  $r$ ), maka kecepatan horizontal ( $x\text{speed}$ ) dikalikan dengan  $-1$  untuk membalikkan arah bola. Hal yang sama juga diterapkan untuk batas atas dan bawah.