

## Modul Praktikum Layer Manager

### 1. Tujuan

- Memahami dan menggunakan class LayerManager
- Menggambar beberapa sprite dan di satukan pada LayerManager
- Menentukan posisi titik awal LayerManager pada screen mobile
- Mengatur besar screen yang di tampilkan pada mobile
- Menggambar animasi pada LayerManager

### 2. Latar Belakang

Setelah kita mempelajari class Sprite, kita akan mempelajari bagaimana pembuatan LayerManager dan pada GameCanvas dan menggabungkan dengan sprite hingga dapat dimanipulasi sprite tersebut, dalam layer manager dapat kita temukan untuk menggerakkan background. Kita akan membuat contoh pembuatan manipulasi sprite dan membuat animasi background pada LayerManager.

Disini peserta wajib mengikuti pembuatan class-class dan memodifikasi class yang telah dibuat agar peserta lebih memahami kinerja atau fungsi-fungsi yang telah dibahas di modul 6 (JENI 4).

### 3. Percobaan

#### Percobaan 1: Membuat Class MidletLayerManager

```
import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;

public class MidletLayerManager extends MIDlet implements CommandListener {
    private Display display;
    private Command cmExit;
    private LayerManagerCanvas lmCanvas;

    public MidletLayerManager() {
        display = Display.getDisplay(this);
        cmExit = new Command("Keluar", Command.EXIT, 1);

        try {
            lmCanvas = new LayerManagerCanvas();
        }
    }
}
```

## Modul Praktikum Layer Manager

```
        lmCanvas.addCommand(cmExit);
        lmCanvas.setCommandListener(this);
        lmCanvas.start();
    } catch (Exception ex) {
        System.out.println(ex);
    }
}

public void startApp() {
    display.setCurrent(lmCanvas);
}

public void pauseApp() { }

public void destroyApp(boolean unconditional) { }

public void exit() {
    destroyApp(true);
    notifyDestroyed();
}

public void commandAction(Command c, Displayable s) {
    if(c == cmExit) {
        exit();
    }
}
};
```

## Modul Praktikum Layer Manager

### Percobaan 2 : Membuat Class LayerManagerCanvas

```
import javax.microedition.lcdui.*;
import javax.microedition.lcdui.game.*;

public class LayerManagerCanvas extends GameCanvas implements Runnable {
    // Menggunakan sprite
    private Sprite pesawatSprite;
    private Sprite backgroundSprite;
    // Layer Manager
    private LayerManager layerManager;

    // Konstruktor dan inisialisasi
    public LayerManagerCanvas() throws Exception {
        super(true);
        // Mengakses gambar dari sprite
        Image gambarPesawat = Image.createImage("/pesawat.png");
        pesawatSprite = new Sprite (gambarPesawat);

        Image gambarBackground = Image.createImage("/background.png");
        backgroundSprite = new Sprite(gambarBackground);
        layerManager = new LayerManager();
        //Menampilkan gambar dari Sprite
        layerManager.append(pesawatSprite);
        layerManager.append(backgroundSprite);
    }

    // Method start dijalankan awal penggunaan thread dan di ulang-ulang
    public void start() {
        Thread t = new Thread(this);
        t.start();
    }

    // method run abstrak dari Runnable Mejalankan pertamakali pada graphics
    public void run() {
        Graphics g = getGraphics();
        drawScreen(g);
    }

    // Method drawScreen untuk menggambar Graphics
    private void drawScreen(Graphics g) {
```

## Modul Praktikum Layer Manager

```
g.setColor(255, 255, 255);
g.fillRect(0, 0, getWidth(), getHeight());
g.setColor(0, 0, 255);
    //Menampilkan posisi gambar pesawat
pesawatSprite.setPosition(30,-10);
    //mengatur background
layerManager.setViewWindow(15,5,70,80);
layerManager.paint(g,5,5);
flushGraphics();
}
};
```

## Modul Praktikum Layer Manager

Hasil :



## Modul Praktikum Layer Manager

### Percobaan 3 : Membuat Class MidletAnimasiBackground

```
import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;

public class MidletAnimasiBackground extends MIDlet implements
CommandListener {

    private Display display;
    private Command cmExit;
    private LayerManagerCanvas lmCanvas;

    public MidletAnimasiBackground() {
        display = Display.getDisplay(this);
        cmExit = new Command("Keluar", Command.EXIT, 1);
        try {
            lmCanvas = new LayerManagerCanvas();
            lmCanvas.addCommand(cmExit);
            lmCanvas.setCommandListener(this);
            lmCanvas.start();
        } catch (Exception ex) {
            System.out.println(ex);
        }
    }

    public void startApp() {
        display.setCurrent(lmCanvas);
    }

    public void pauseApp() { }

    public void destroyApp(boolean unconditional) { }

    public void exit() {
        destroyApp(true);
        notifyDestroyed();
    }
}
```

## Modul Praktikum Layer Manager

```
public void commandAction(Command c, Displayable s) {  
    if(c == cmExit) {  
        exit();  
    }  
}  
};
```

### Percobaan 4 : Membuat Class AnimasiBackgroundCanvas

```
import javax.microedition.lcdui.*;  
import javax.microedition.lcdui.game.*;  
  
public class AnimasiBackgroundCanvas extends GameCanvas implements Runnable {  
    private boolean gerak; // background bergerak jika variabel gerak itu bernilai true  
    private long delay = 20; // Pemberian waktu yang konsisten pada thread  
    private int width = getWidth(); // lebar screen pada mobile  
    private int height = getHeight(); // tinggi screen pada mobile  
    private int scnX = 55, scnY = 20; // posisi x dan y pada animasi background  
    private int layer = 125; // Penentuan layer yang ditampilkan  
  
    // Menggunakan sprite  
    private Image gambarBackground;  
    private Sprite backgroundSprite;  
    // Layer Manager  
    private LayerManager layerManager;  
  
    // Konstruktor dan inisialisasi  
    public AnimasiBackgroundCanvas() throws Exception {  
        super(true);  
        // Mengambil gambar pada Sprite  
        gambarBackground = Image.createImage("/background.png");  
        backgroundSprite = new Sprite(gambarBackground);  
        layerManager = new LayerManager();  
        layerManager.append(backgroundSprite);  
    }  
  
    // Method start dijalankan awal penggunaan thread dan di ulang-ulang  
    public void start() {  
        gerak = true;  
        Thread t = new Thread(this);  
        t.start();  
    }  
}
```

## Modul Praktikum Layer Manager

```
// Method stop berfungsi untuk memberhentikan gambar
public void stop() {
    gerak = false;
}

// method run abstrak dari Runnable pada graphics dan berulang ulang
dijalankan karena while
public void run() {
    Graphics g = getGraphics();
    while(gerak == true) {
        if(scnX < gambarBackground.getWidth()-layer) {
            scnX++;
        }
        drawScreen(g);
        try { Thread.sleep(delay); }
        catch (InterruptedException ie) {}
    }
}

// Method drawScreen untuk menggambar Graphics
private void drawScreen(Graphics g) {
    g.setColor(255, 255, 255);
    g.fillRect(0, 0, getWidth(), getHeight());
    g.setColor(0, 0, 255);
    //Menampilkan semua layer
    layerManager.setViewWindow(scnX, (gambarBackground.getHeight()-scnY)-
                                layer, layer, layer);
    layerManager.paint(g, 5, 5);
    flushGraphics();
}
}
```



## Modul Praktikum Layer Manager

Hasil :