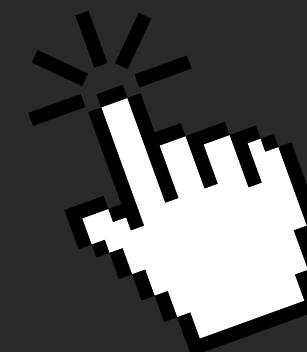


BACK

NEXT

ANGGOTA KELompok



YUSUF RAIS WICAKSANA
5002241147

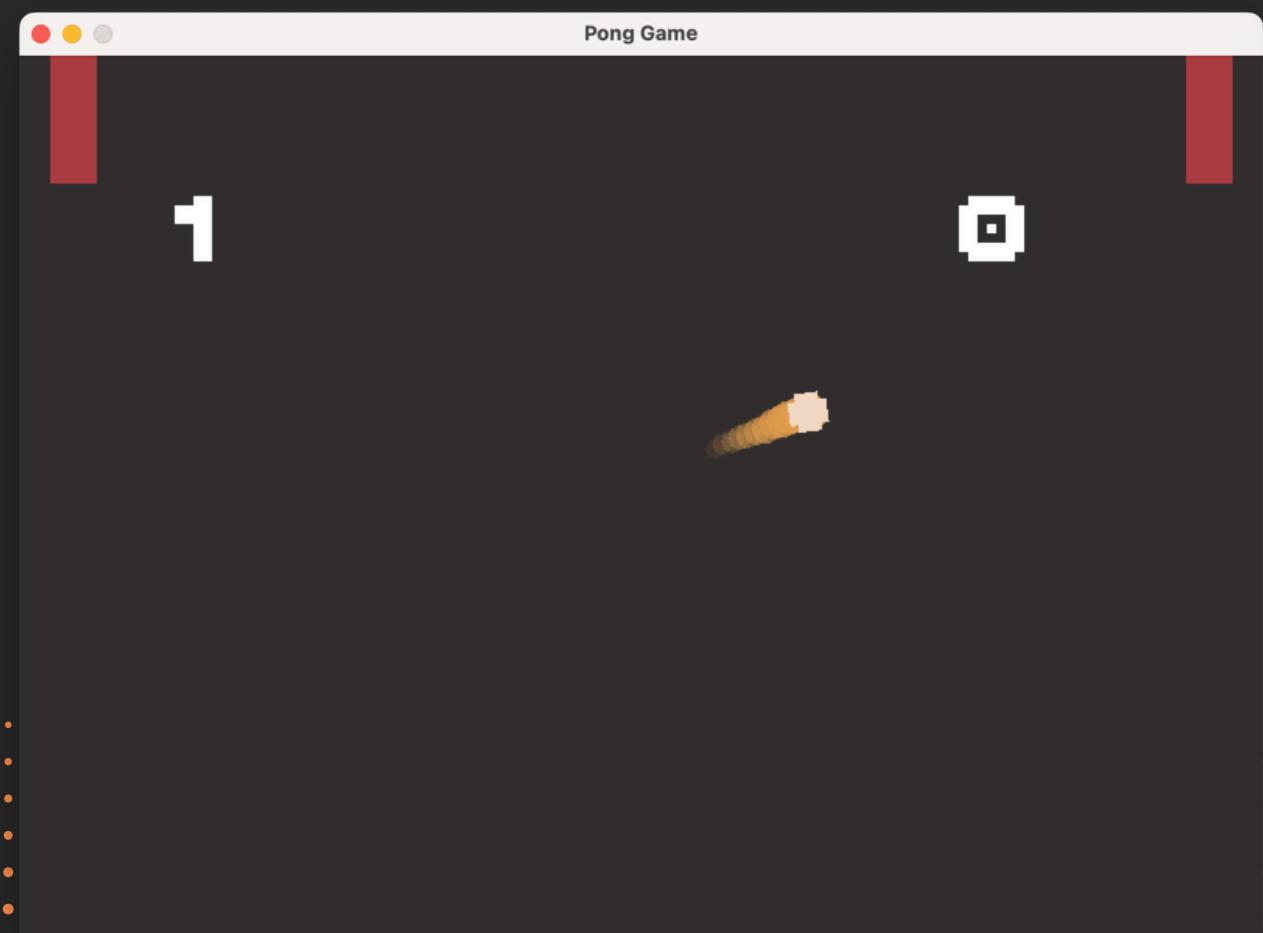
RISKA YENI ZULMIA
5002241055

BACK

NEXT

GAME PONG

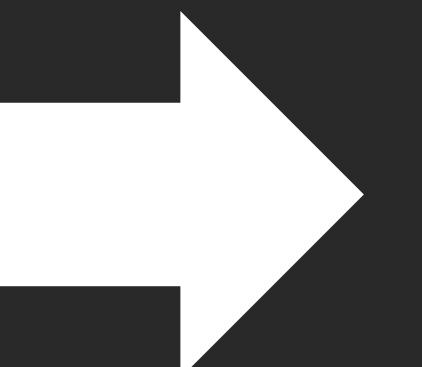
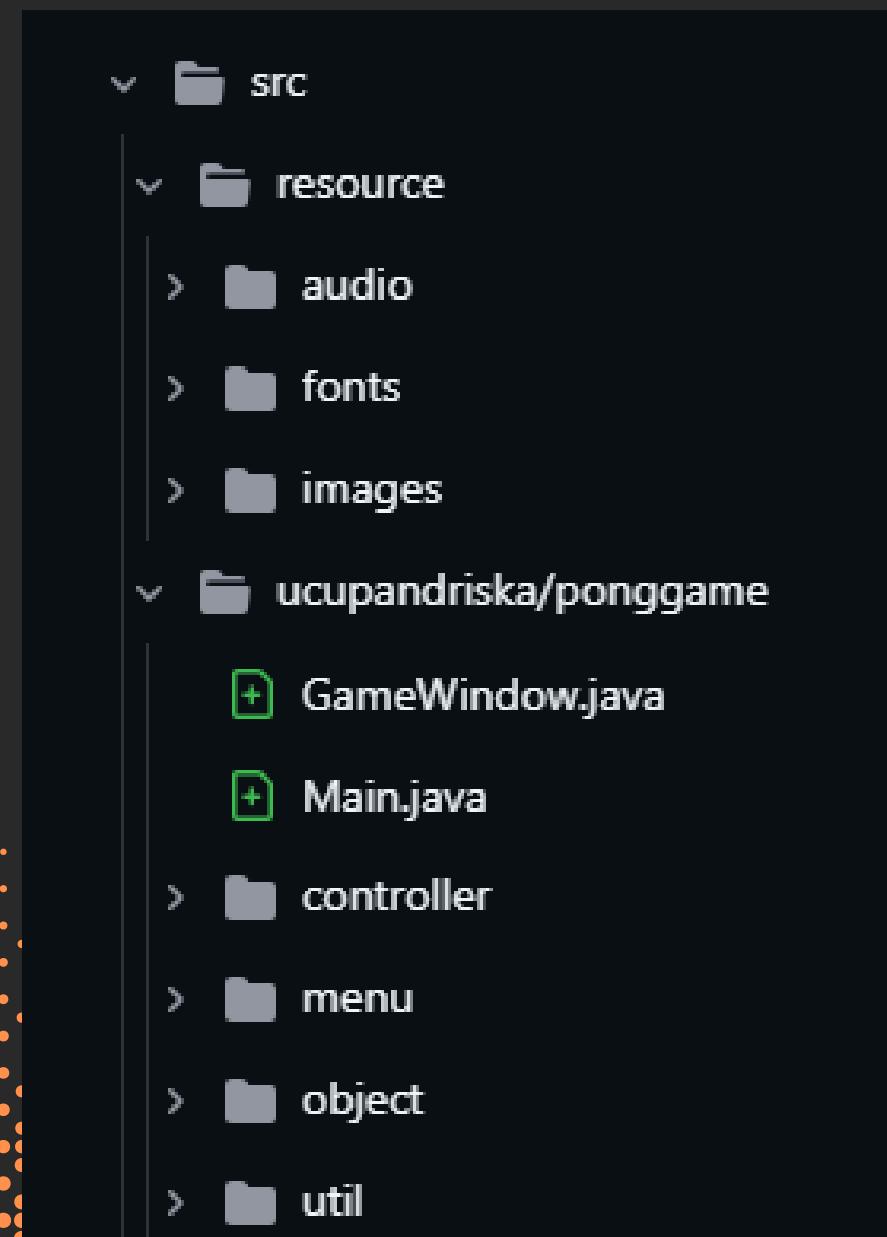
Pong adalah salah satu video game pertama yang pernah dibuat dan menjadi pelopor dalam industri game. Pong merupakan simulasi sederhana dari permainan tenis meja (ping pong), di mana dua pemain mengendalikan balok (paddle) untuk memantulkan bola bolak-balik di layar.



BACK

NEXT

STRUKTUR FILE



```
menu
+ AbstractMenu.java
+ DifficultyMenu.java
+ GameOverMenu.java
+ MainMenu.java
+ MenuItem.java
+ ModeMenu.java
+ PauseMenu.java
```

```
object
+ BallTrail.java
+ Rect.java
+ RotatingImageObject.j...
+ Text.java
```

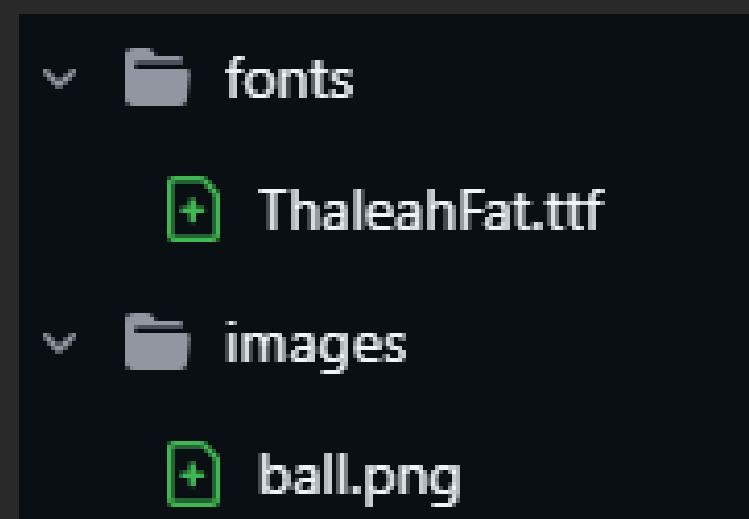
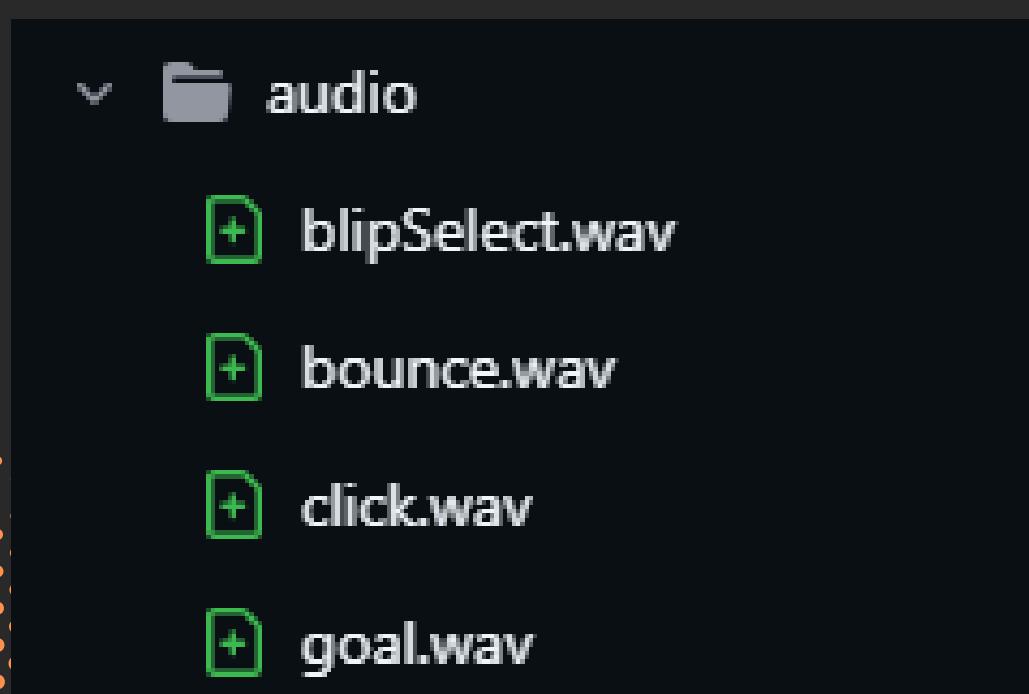
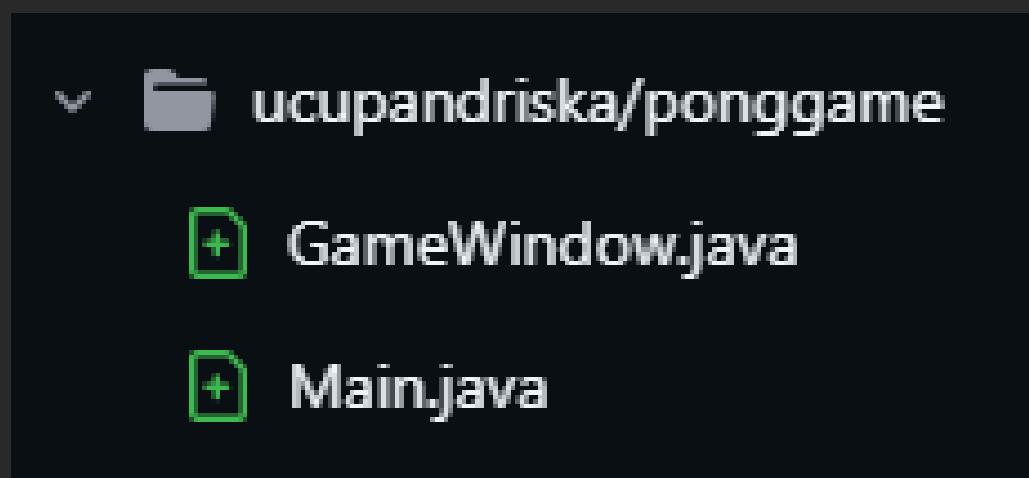
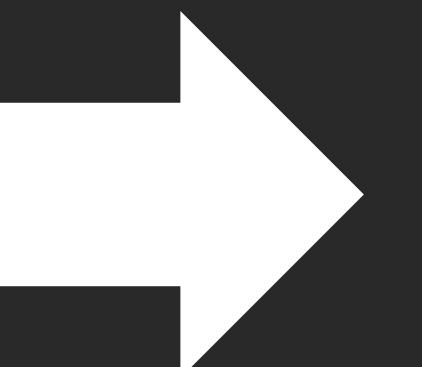
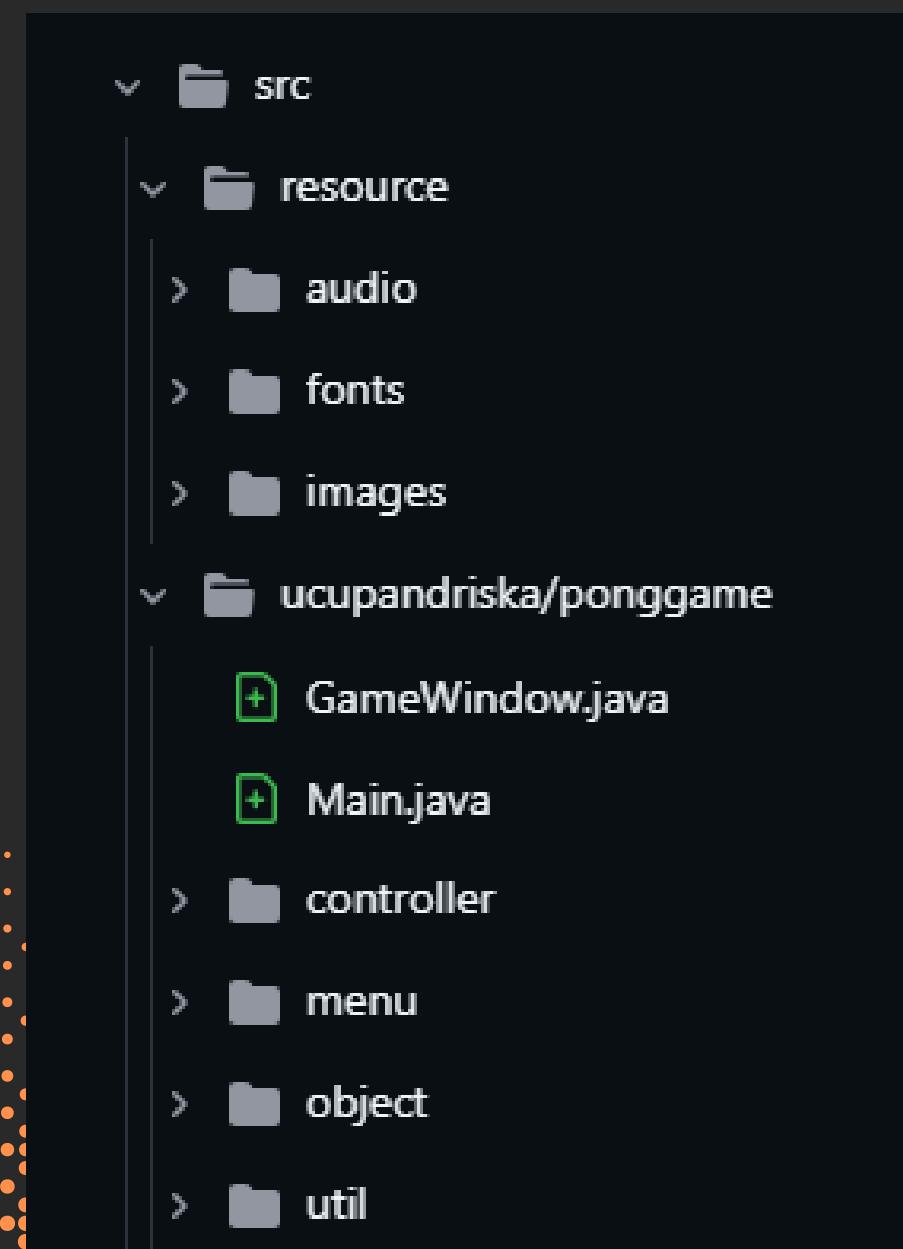
```
controller
+ BallController.java
+ BotController.java
+ PlayerController.java
```

```
util
+ AudioLoader.java
+ ColorScheme.java
+ Const.java
+ Countdown.java
+ Difficulty.java
+ FontLoader.java
+ GameSettings.java
+ ImageLoader.java
+ KL.java
+ MenuInfoHelper.java
+ Time.java
```

BACK

NEXT

STRUKTUR FILE



PENJELASAN PROGRAM



```
1 package ucupandriska.ponggame.util;  
2 import java.awt.event.KeyEvent;  
3  
4 public class Const {  
5     public static final int SCREEN_WIDTH = 800;  
6     public static final int SCREEN_HEIGHT = 600;  
7  
8     public static final String WINDOW_TITLE = "Pong Game";  
9  
10    public static final double FPS = 60.0;  
11    public static final double FRAME_TIME = 1.0 / FPS;
```

Const adalah singkatan dari "Constants".

- Kelas ini menyimpan semua nilai tetap (konstanta) yang digunakan dalam game.
- Menggunakan `public static final` agar:
 - `public` → bisa diakses dari kelas lain.
 - `static` → tidak butuh objek `Const` untuk mengakses.
 - `final` → nilainya tidak dapat diubah setelah ditetapkan (konstanta).

PENGELASAN PROGRAM



```
1 package ucupandriska.ponggame.util;
2 import java.awt.event.KeyEvent;
3
4 public class Const {
5     public static final int SCREEN_WIDTH = 800;
6     public static final int SCREEN_HEIGHT = 600;
7
8     public static final String WINDOW_TITLE = "Pong Game";
9
10    public static final double FPS = 60.0;
11    public static final double FRAME_TIME = 1.0 / FPS;
12
13    public static final double PADDLE_WIDTH = 30.0;
14    public static final double PADDLE_HEIGHT = 90.0;
15    public static final double BALL_WIDTH = 25.0;
16    public static final double H_PADDING = 20.0;
17    public static final double V_PADDING = 20.0;
18
19    public static final double MV_INCREMENT = 300;
20    public static final double BALL_SPEED = 400;
21
22    public static double TOOLBAR_HEIGHT;
23    public static double INSETS_BOTTOM;
24
25    public static final int BIND_UP = KeyEvent.VK_UP;
26    public static final int BIND_DOWN = KeyEvent.VK_DOWN;
27    public static final int BIND_UP_ALT = KeyEvent.VK_W;
28    public static final int BIND_DOWN_ALT = KeyEvent.VK_S;
29
30    public static final String FONT_PATH = "/resource/fonts/ThaleahFat.ttf";
31
32    public static final double SCORE_FONT_SIZE = 96f;
33    public static final double PLAYER_ONE_SCORE_POS_X = SCREEN_WIDTH * (1.0 / 8.0);
34    public static final double PLAYER_TWO_SCORE_POS_X = SCREEN_WIDTH * (7.0 / 8.0) - SCORE_FONT_SIZE;
35    public static final double SCORE_POS_Y = 160;
36
37    public static final double COUNTDOWN_FONT_SIZE = 196f;
38
39    public static final int SCORE_TO_WIN = 3;
40 }
```

PENGELASAN PROGRAM



```
1 package ucupandriska.ponggame.util;
2
3 import java.awt.Color;
4
5 public class ColorScheme {
6     // Background
7     public static final Color BACKGROUND = hex("#343131");           // Dark gray
8     public static final Color MENU_BACKGROUND = hex("#343131");         // Dark gray
9
10    // Text
11    public static final Color TEXT_PRIMARY = hex("#EEDF7A");           // Soft yellow
12    public static final Color TEXT_HIGHLIGHT = hex("#D8A25E");          // Muted orange
13    public static final Color TEXT_GAMEOVER_TITLE = hex("#4fab66");
14    public static final Color TEXT_ACTIVE_DIFFICULTY = hex("#7d706f");
15
16    // Game objects
17    public static final Color PADDLE = hex("#A04747");                  // Warm red
18    public static final Color BALL = hex("#EEDF7A");                      // Soft yellow
19    public static final Color SCORE = hex("#D8A25E");                     // Muted orange
20
21    // UI / FX
22    public static final Color OUTLINE = hex("#A04747");                 // Red tone
23    public static final Color SHADOW = new Color(0, 0, 0, 100);           // Transparent black
24
25    private static Color hex(String hexCode) {
26        return Color.decode(hexCode);
27    }
28 }
```

PENJELASAN PROGRAM



```
1 public class PlayerController {  
2     public Rect rect;  
3     public KL keyListener;  
4     public int playerBindUp, playerBindDown;  
5     private double paddleSpeed;  
6  
7     public PlayerController(Rect rect, KL keyListener, int playerBindUp, int playerBindDown) {  
8         this.rect = rect;  
9         this.keyListener = keyListener;  
10        this.playerBindUp = playerBindUp;  
11        this.playerBindDown = playerBindDown;  
12        this.paddleSpeed = this.paddleSpeed = GameSettings.getDifficulty().getPaddleSpeed();  
13        System.out.println("Paddle Speed: " + paddleSpeed);  
14    }
```

Kelas ini berfungsi sebagai pengontrol pemain (paddle), yang bertugas:
Mengontrol gerakan paddle (atas/bawah) berdasarkan input keyboard.
Menyesuaikan posisi paddle berdasarkan input dari pemain dan waktu.

PENJELASAN PROGRAM



```
1 public class BallController {  
2     private GameWindow window; // Reference to GameWindow  
3     public RotatingImageObject rect;  
4     public Rect leftPaddle, rightPaddle;  
5     public Text playerOneScoreText, playerTwoScoreText;  
6     public Countdown timer;  
7     private double ballSpeed;  
8  
9     private double vx = -1;  
10    private double vy = 0;  
11  
12    private final double maxBounceAngle = Math.toRadians(60);  
13    private boolean ballJustScored = false;  
14    private boolean gameStarted = false;  
15    private boolean gameOver = false;
```

Merupakan pengendali utama perilaku bola dalam game Pong. Fungsinya mencakup pergerakan bola, deteksi tumbukan dengan paddle dan dinding, pencatatan skor, serta reset bola setelah gol.

PENJELASAN PROGRAM



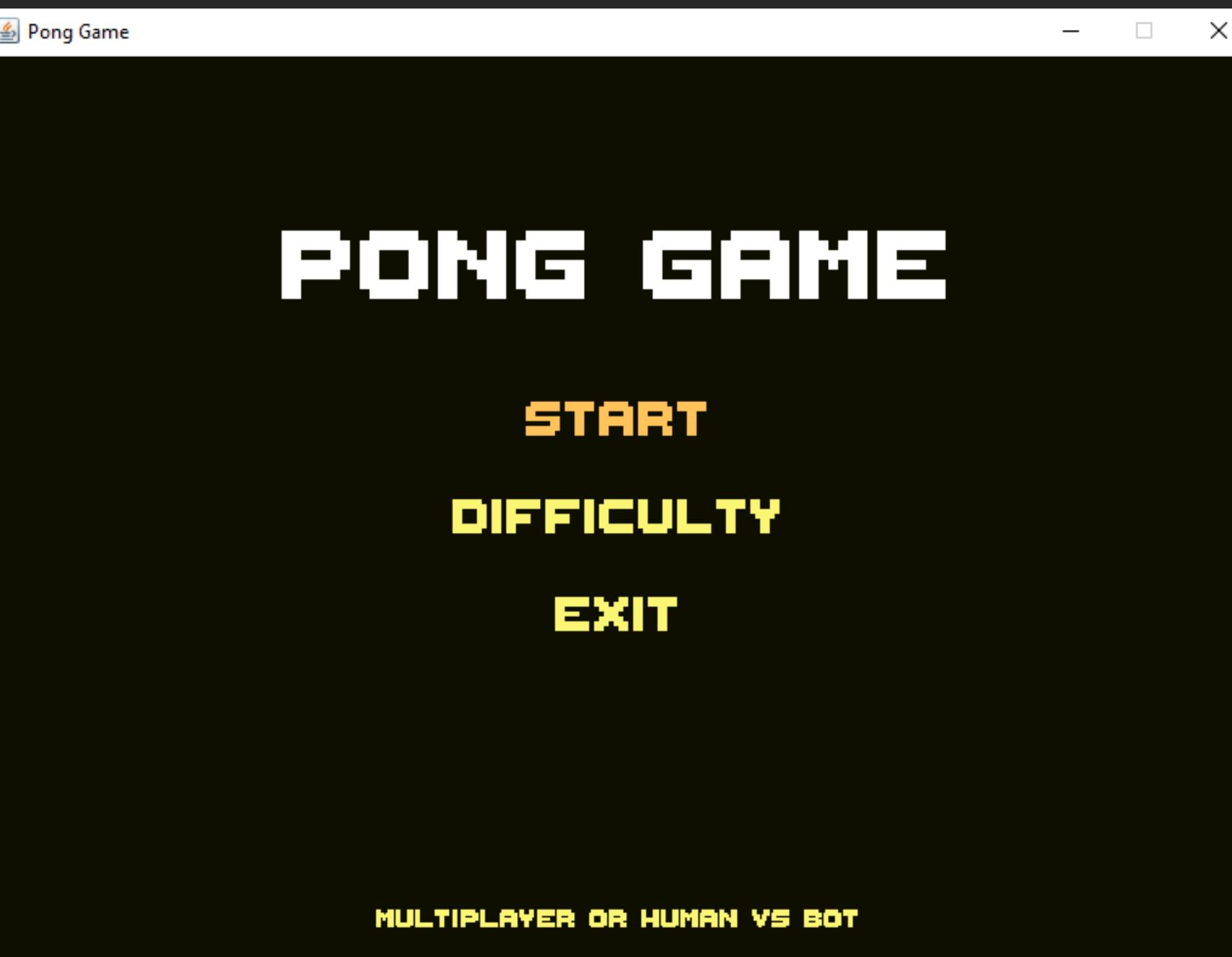
```
1 public class GameWindow extends JFrame implements Runnable {  
2     public Graphics2D g2;  
3     public KL keyListener = new KL(); // Your KL class instance  
4     private boolean isPaused = false;  
5     private PauseMenu pauseMenu;  
6  
7     public Rect playerOne, playerTwo;  
8     public RotatingImageObject ball;  
9     public PlayerController playerOneController, playerTwoController;  
10    public BallController ballController;  
11    public Text playerOneScoreText, playerTwoScoreText;  
12    public int playerOneScore, playerTwoScore;  
13  
14    Font customFontScore = FontLoader.loadFont(Const.FONT_PATH, (float)Const.SCORE_FONT_SIZE);
```

Program GameWindow ini adalah inti dari game Pong yang mengatur window, logika game loop, dan rendering grafis. Kelas ini merupakan turunan dari JFrame dan mengimplementasikan Runnable agar game bisa berjalan dalam game loop (perulangan terus menerus).

BACK

CARA KERJA GAME PONG

NEXT



BACK

CARA KERJA GAME PONG

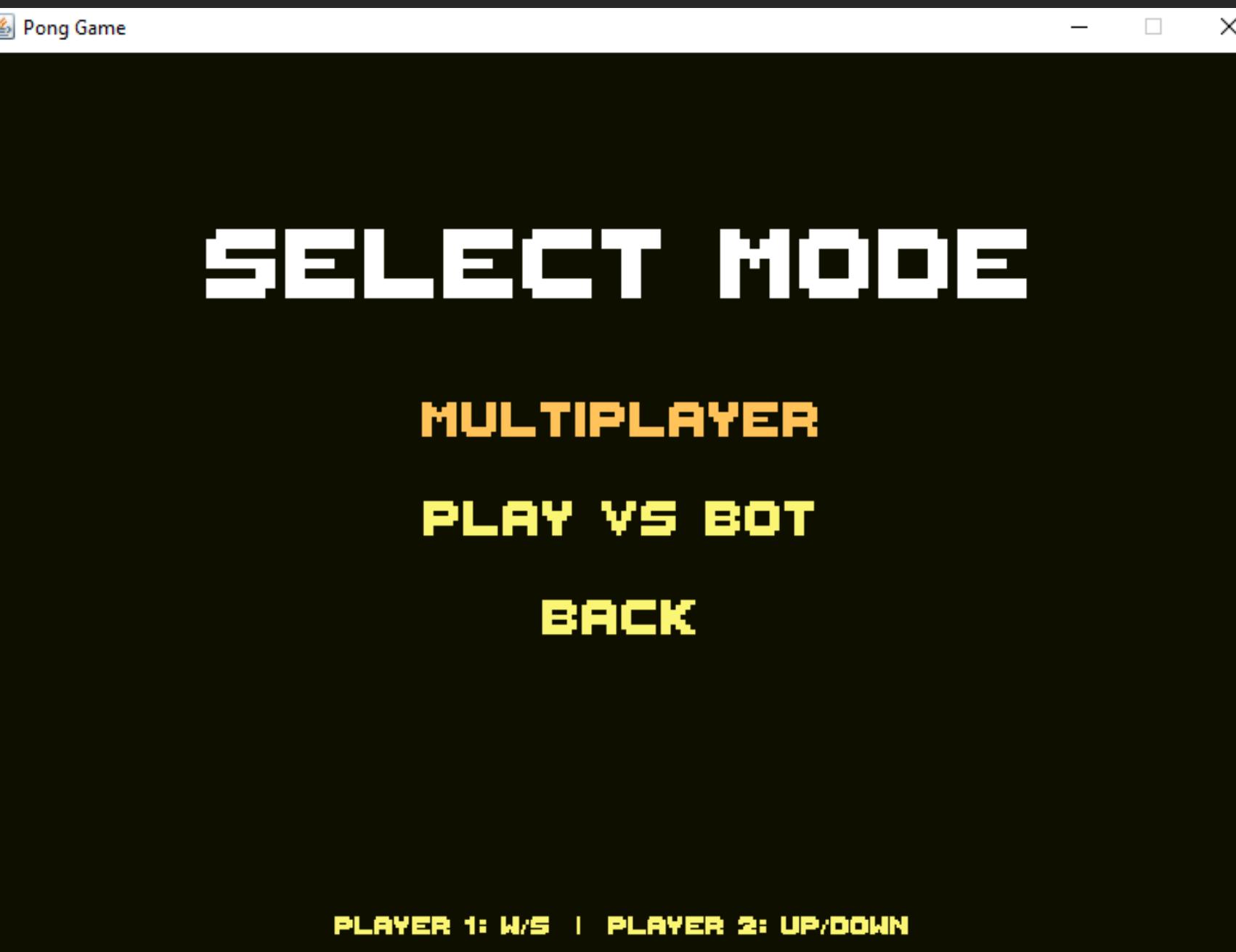
NEXT



BACK

CARA KERJA GAME PONG

NEXT



BACK

**THANK
YOU**

END