Snake Game (Java Swing) - Full Code Explanation

This document explains each part of the Snake Game code in clear and simple language.

✅1. Main Class: SnakeGame

This class creates the game window.

What it does:

- Creates a JFrame (window)
- Adds the GamePanel (where the game runs)
- Sets title, size, close operation
- Shows the window

Why needed:

Every Java GUI program starts with a window. SnakeGame is that window.

✅2. Class GamePanel

This is the **heart of the game**. It handles: ✅Drawing snake & food ✅Movement ✅Game logic (collision, eating food) ✅Score ✅Game Over ✅Restart button

Key Variables:

- `SCREEN_WIDTH`, `SCREEN_HEIGHT` → game area size
- `UNIT_SIZE` → size of each grid block
- `x[]`, `y[]` → store snake body positions
- `bodyParts` → length of the snake
- `foodX`, `foodY` → food position
- `direction` → U, D, L, R
- `running` → true/false
- `timer` → moves snake every 75 ms
- `restartButton` → button after game ends

✅3. Constructor GamePanel()

This runs once at the start.

What it does:

- Sets panel size and background
- Enables keyboard controls

- Adds Restart button
- Starts the game

✅4. startGame()

Resets and starts the game.

Actions:

- Places new food
- Sets snake starting length & direction
- Hides restart button
- Starts timer for movement

✅5. restartGame()

Runs when user clicks the Restart button.

What it does:

- Clears old snake position
- Calls `startGame()`

✅6. paintComponent()

This method draws everything on screen.

It calls:

- `draw(g)` → where actual drawing happens

✅7. draw(Graphics g)

Handles all drawing.

Draws:

- Food (red circle)
- Snake (green blocks)
- Score
- OR Game Over screen if game ended

✅8. newFood()

Places new food randomly on the grid.

✅9. move()

Moves the snake one block.

How it works:

- Moves each body part to the spot of the one before it
- Moves head based on direction

Example: - If direction == 'U' → y[0] -= UNIT_SIZE

✅10. checkFood()

Detects when snake eats food.

When head touches food: - Snake grows (`bodyParts++`) - Score increases - New food appears

---

✅11. checkCollision()

Detects game over.

Checks:

- Snake hits its own body
- Snake hits wall

If collision: - Stop timer - Show restart button

✅12. gameOver(Graphics g)

This draws: - GAME OVER text - Final score - Shows restart button

---

✅13. actionPerformed()

Called every 75 ms by the timer.

What it does:

- Moves snake

- Checks food collision
- Checks wall/body collision
- Repaints screen

✅14. MyKeyAdapter (keyboard controls)

Controls snake direction.

Rules:

- Cannot move opposite direction (e.g., can't go right if currently going left)
- Arrow keys move snake

✅Summary

This Snake Game uses: - Java Swing for graphics - Timer for movement - Arrays to store snake body - Keyboard input for direction - Collision detection - Restart button for user-friendly replay

Perfect for Java beginners learning: ✅OOP ✅GUI ✅Game Loop ✅Event Handling