

# Loops

November 11, 2008

---

---

---

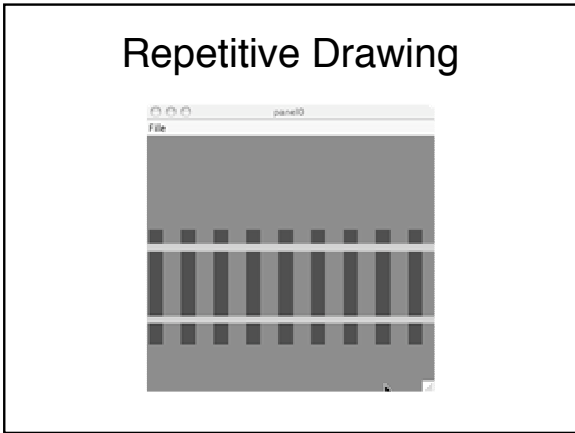
---

---

---

---

---



---

---

---

---

---

---

---

---

## Drawing Train Tracks

```
public void begin () {  
    new FilledRect(0, 0, SCREEN_WIDTH, SCREEN_HEIGHT,  
        canvas).setColor(GROUND_COLOR);  
    double tiePosition = 5;  
    while(tiePosition < SCREEN_WIDTH){  
        new FilledRect(tiePosition, TIE_TOP, TIE_WIDTH,  
            TIE_LENGTH, canvas).setColor(TIE_COLOR);  
        tiePosition = tiePosition + TIE_WIDTH + TIE_SPACING;  
    }  
    new FilledRect(0, TRACK_TOP, SCREEN_WIDTH, RAIL_WIDTH,  
        canvas).setColor(RAIL_COLOR);  
    new FilledRect(0, TRACK_TOP + GAUGE, SCREEN_WIDTH,  
        RAIL_WIDTH, canvas).setColor(RAIL_COLOR);  
}
```

Condition

Statements to repeat

---

---

---

---

---

---

---

---

## While Loop

- initialization
- while ( condition ) {
- statements to be repeated
- update (something that is part of the condition)
- }
- Note:
  - initialization is before the loop
  - update happens in each iteration

---

---

---

---

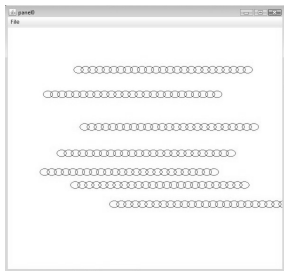
---

---

---

---

## Simple Chain



---

---

---

---

---

---

---

---

## While Loop

- Questions:
  - What is repeating?
  - What is the condition?
    - for continuing the loop?
    - for terminating the loop?
  - What changes inside the loop that is part of the condition?
  - What initialization needs to happen before the loop begins?

---

---

---

---

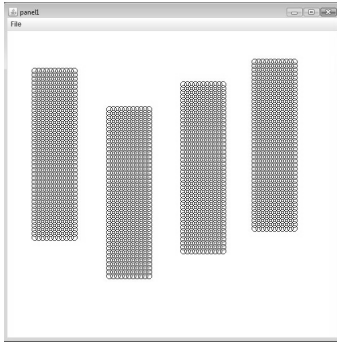
---

---

---

---

## Now Let's draw a scarf



---

---

---

---

---

---

---

---

## Nested Loops

```
public void onMouseClick (Location point) {  
    double x = point.getX();  
    double y = point.getY();  
  
    // while there are more rows to knit  
    while (y < point.getY() + SCARF_HEIGHT) {  
  
        // knits one row  
        while (x < point.getX() + SCARF_WIDTH) {  
  
            // knits one stitch  
            new FramedOval(x, y, DIAMETER, DIAMETER, canvas);  
            x = x + X_DISP;  
        }  
  
        x = point.getX();  
        y = y + Y_DISP;  
    }  
}
```

---

---

---

---

---

---

---

---

## What is a Digital Image?



---

---

---

---

---

---

---

---

## Modifying a Digital Image

```
public void brighten (Picture pic) {
    int row = 0; int col = 0;
    while ( row < imgHeight ) {
        while ( col < imgWidth ) {
            Color originalColor = pic.getPixel(row, col);
            Color brighterColor = originalColor.brighter();
            pic.setPixel (row, col, brighterColor);
            col = col + 1;
        } // end of inner while loop
        row = row + 1;
    } // end of outer while loop
}
```

There is Problem Here !

---

---

---

---

---

---

---

---

## Counting Clicks

```
public class Firework {
    // Initialize the counter
    private int numFireworks = 0;

    public void onMouseClick (Location point) {
        // remove last firework

        // Test the counter
        if (numFireworks < MAX_FIREWORKS) {
            new Firework (...);

            // Increase the counter
            numFireworks++;
        }
    }
}
```

---

---

---

---

---

---

---

---

## Counting Loops

```
private void finale() {
    // Initialize counter
    int finaleCount = 0;

    // Test counter
    while (finaleCount < NUM_IN_FINALE) {
        new Firework (...);

        // Update counter
        finaleCount++;
    }
}
```

---

---

---

---

---

---

---

---

## Summary

Keys to implementing a loop

What should the loop do?

What should happen each time through the loop?

Under what condition should the loop continue?

What update happens inside the loop to ensure the loop eventually ends?

What initialization must happen before reaching the loop?

---

---

---

---

---

---

---

---

## Counting

```
Initialize counter
/
-int fireworkCount = 0;
-while (fireworkCount < NUM_IN_FINALE) {
  -new Firework (...);
  -fireworkCount++; Test counter against limit
-}
  Increment counter
```

---

---

---

---

---

---

---

---

## Counting with For Loops

```
Initialize counter
|
Test counter against limit
/
-for (int fireworkCount = 0;
  fireworkCount < NUM_IN_FINALE;
  fireworkCount++) {
  -new Firework (...);
-}
  Increment counter
```

---

---

---

---

---

---

---

---