Loops

- Often you need to repeat an action in a program

Start

Need More Sandwiches?

- Yes
  - Make Sandwich
  - Distribute Sandwiches
- No
Loops Components

- **Loop**: Part of a program that repeats
- **Body**: The statements being repeated
- **Iteration**: A repetition of the body
- **Controlling Condition**: A Boolean expression that indicates the loop should repeat

Diagram:
- Start
- Need More Sandwiches?
- Yes
  - Make Sandwich
  - Distribute Sandwiches
- No
Types of Loops

- While
- Do...While
- For
- For Each
While Loop

**Flowchart**

1. Start
2. Need More Sandwiches?
   - Yes: Make Sandwich
   - No: Distribute Sandwiches

**Pseudocode**

1. Start
2. While Need More Sandwiches
   1. Make Sandwich
3. Distribute Sandwiches
while Loop

- Syntax
  - while (Boolean_Expression)
    Body_Statement;

- Execution
  - If the Boolean_Expression evaluates to true, the Body_Statement is executed
  - As long as Boolean_Expression remains true, the Body_Statement is executed again
  - If the Boolean_Expression evaluates to false, flow of control continues to the next statement
while Loop

- Multiple statements can be executed in sequence using a block

Syntax

- while (Boolean_Expression)
  {
    Body_Statement_1;
    Body_Statement_2;
    // ...
  }
Loop Termination

- The loop terminates when the controlling expression becomes false
  - Requires the body of the loop to cause this change

- Infinite Loop: a loop with a controlling expression that never becomes false
  - Often caused by a body that does not cause the controlling expression to change to false
Using a while Loop

Example: Print the numbers from 1 to 10

```java
int n = 1;
System.out.println(n);
while (n <= 10)
{
    System.out.println(n);
    n = n + 1;
}
```

```java
int n = 1;
System.out.println(n);
while (n <= 10)
{
    System.out.println(n);
    n = n + 1;
}
```
Questions?
Logistics

Next:
- Open Lab Time

Tomorrow:
- More Loops
- Lab 3

Reminders
- Lab 1 is Due Tomorrow
- Program 1 is Due Friday
- Lab 2 is Due Friday