Control Flow Revisited
Chris Piech
CS106A, Stanford University
Java

Piech, CS106A, Stanford University
Making a New Variable

```
int age = 29;
```

- **type**: `int`
- **name**: `age`  
- **Initial value**: `29`
My computer has space for about 2 billion boxes
// Create a variable, of type int
// called age with the value 29.
int age = 29;

// Modify age to be one greater.
age = age + 1;

// Use the value in age (output it)
println("age is: " + age);
// integer values
int num = 5;

// real values
double fraction = 0.2;

// letters
char letter = 'c';

// true or false
boolean isLove = true;

* Why is it called a double? *
All *binary operators* result in a value (like a temporary variable) which *has a type*. The general rule is: operations always return the *most expressive* type:

Expressive hierarchy:

```
boolean < char < int < double
```

Example:

```
int / double  results in a double
```
Write a program that can turn a measurement of C14 into an estimate of age.

Radioactive molecule = C14
Halflife = 5730 years
C14 in living organisms = 13.6 dpm

What is the amount of C14 remaining in your sample: 10.2
Your sample is 2378.0 years old.
Example: Carbon Dating

C14 = 1.2 dpm

C14 = 13.6 dpm
Carbon Dating Equation

\[
\text{age} = \frac{\log\left(\frac{c}{13.6}\right)}{\log\left(\frac{1}{2}\right)} \times 5730
\]

* Some of these values are constants
** Use the function: Math.log( num )
End Review
• **constant**: A variable that cannot be changed after it is initialized. Declared at the top of your class, *outside of the run() method*. Can be used anywhere in that class.

• Better style – can easily change their values in your code, and they are easier to read in your code.

• Syntax:

```
private static final type name = value;
```

  – name is usually in *ALL_UPPER_CASE*

  – Examples:

```
private static final int DAYS_IN_WEEK = 7;
private static final double TAX_PERCENT = 0.08;
private static final int SSN = 658234569;
```
public class Receipt extends ConsoleProgram {
    public void run() {
        double subtotal = readDouble("Meal cost? $");
        double tax = subtotal * 0.08;
        double tip = subtotal * 0.20;
        double total = subtotal + tax + tip;

        println("Tax : "+ tax);
        println("Tip: "+ tip);
        println("Total: "+ total);
    }
}
public class Receipt extends ConsoleProgram {
    private static final double TAX_RATE = 0.08;
    private static final double TIP_RATE = 0.2;

    public void run() {
        double subtotal = readDouble("Meal cost? $");
        double tax = subtotal * TAX_RATE;
        double tip = subtotal * TIP_RATE;
        double total = subtotal + tax + tip;

        println("Tax : $" + tax);
        println("Tip: $" + tip);
        println("Total: $" + total);
    }
}
Binary Operators

+ Addition
- Subtraction
* Multiplication
/ Division
% Remainder

Today is your day, tio
// an example of the % operator
println(17 % 4);

// reads a number from the user
int num = readInt("?: ");

// stores the ones digit
int onesDigit = num % 10;

// equal to 1 if num is odd,
// 0 if num is even.
int isOdd = num % 2;
Today’s Goal

1. Be able to use For / While / If in Java
Today’s Route

You are here

Simple Java

The River of Java

Review

Conditions

Game Show

For Loops

Piech, CS106A, Stanford University
Today’s Route

Simple Java

You are here

Review

Conditions

For Loops

Game Show

The River of Java

Piech, CS106A, Stanford University
While Loop in Karel

```plaintext
while(frontIsClear()) {
    body
}

if(beepersPresent()) {
    body
}
```
The condition should be a “boolean” which is either **true** or **false**
What Does This Do?

// read the amount of C14 from the user
double amount = readDouble("Amount of C14 in your sample: ");

// use the half life formula to calculate the age
double fractionLeft = amountLeft / LIVING_C14;
double age = Math.log(fractionLeft) / Math.log(0.5) * HALF_LIFE;
println("Your sample is " + age + " years old.");

* It calculates the age of a C14 sample
What Does This Do?

Before repeating the body, check if this statement evaluates to true.

```java
while(true) {
    // read the amount of C14 from the user
    double amount = readDouble("Amount of C14 in your sample: ");

    // use the half life formula to calculate the age
    double fractionLeft = amountLeft / LIVING_C14;
    double age = Math.log(fractionLeft) / Math.log(0.5) * HALF_LIFE;
    println("Your sample is " + age + " years old.");

    // add an extra line between queries
    println("\n");
}
```

* It repeatedly calculates the age of a C14 sample.
Booleans

1 < 2
Booleans

1 < 2

true
## Comparison Operators

<table>
<thead>
<tr>
<th>Operator</th>
<th>Meaning</th>
<th>Example</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>==</td>
<td>equals</td>
<td>$1 + 1 == 2$</td>
<td>true</td>
</tr>
<tr>
<td>!=</td>
<td>does not equal</td>
<td>$3.2 != 2.5$</td>
<td>true</td>
</tr>
<tr>
<td>&lt;</td>
<td>less than</td>
<td>$10 &lt; 5$</td>
<td>false</td>
</tr>
<tr>
<td>&gt;</td>
<td>greater than</td>
<td>$10 &gt; 5$</td>
<td>true</td>
</tr>
<tr>
<td>&lt;=</td>
<td>less than or equal to</td>
<td>$126 &lt;= 100$</td>
<td>false</td>
</tr>
<tr>
<td>&gt;=</td>
<td>greater than or equal to</td>
<td>$5.0 &gt;= 5.0$</td>
<td>true</td>
</tr>
</tbody>
</table>

* All have equal precedence
## Comparison Operators

<table>
<thead>
<tr>
<th>Operator</th>
<th>Meaning</th>
<th>Example</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>==</td>
<td>equals</td>
<td>$1 + 1 == 2$</td>
<td>true</td>
</tr>
<tr>
<td>!=</td>
<td>does not equal</td>
<td>$3.2 \ne 2.5$</td>
<td>true</td>
</tr>
<tr>
<td>&lt;</td>
<td>less than</td>
<td>$10 &lt; 5$</td>
<td>false</td>
</tr>
<tr>
<td>&gt;</td>
<td>greater than</td>
<td>$10 &gt; 5$</td>
<td>true</td>
</tr>
<tr>
<td>&lt;=</td>
<td>less than or equal to</td>
<td>$126 \le 100$</td>
<td>false</td>
</tr>
<tr>
<td>&gt;=</td>
<td>greater than or equal to</td>
<td>$5.0 \ge 5.0$</td>
<td>true</td>
</tr>
</tbody>
</table>

* All have equal precedence
Comparison Operators

```java
if (1 < 2) {
    println("1 is less than 2!");
}

int num = readInt("Enter a number: ");
if (num == 0) {
    println("That number is 0!");
} else {
    println("That number is not 0.");
}
```
int num = readInt("Enter a number: ");
if (num == 0) {
    println("Your number is 0 ");
} else {
    if (num > 0) {
        println("Your number is positive");
    } else {
        println("Your number is negative");
    }
}
int num = readInt("Enter a number: ");
if (num == 0) {
    println("Your number is 0 ");
} else if (num > 0) {
    println("Your number is positive");
} else {
    println("Your number is negative");
}
Example: Sentinel Loops

- **sentinel**: A value that signals the end of user input.
  - **sentinel loop**: Repeats until a sentinel value is seen.

- Example: Write a program that prompts the user for numbers until the user types -1, then output the sum of the numbers.
  - In this case, -1 is the sentinel value.

  ```
  Type a number: 10
  Type a number: 20
  Type a number: 30
  Type a number: -1
  Sum is 60
  ```
Example: Sentinel Loops

// fencepost problem!
// ask for number - post
// add number to sum - fence

```java
int sum = 0;
int num = readInt("Enter a number: ");
while (num != -1) {
    sum += num;
    num = readInt("Enter a number: ");
}
println("Sum is " + sum);
```
Example: Sentinel Loops

// Solution #2
// harder to see loop end condition here

int sum = 0;
while (true) {
    int num = readInt("Enter a number: ");
    if (num == -1) {
        break;  // immediately exits loop
    }
    sum += num;
}
println("Sum is " + sum);
Guess My Number

I am thinking of a number between 0 and 99...
Enter a guess: 50
Your guess is too high

Enter a new number: 25
Your guess is too low

Enter a new number: 40
Your guess is too low

Enter a new number: 45
Your guess is too low

Enter a new number: 48
Congrats! The number was: 48
```java
int secretNumber = SECRET;
println("I am thinking of a number between 0 and 99...");
int guess = readInt("Enter a guess: ");
// true if guess is not equal to secret number
while(guess != secretNumber) {
    // true if guess is less than secret number
    if(guess < secretNumber) {
        println("Your guess is too low");
    } else {
        println("Your guess is too high");
    }
    println(" "); // an empty line
    guess = readInt("Enter a new number: ");
}
println("Congrats! The number was: " + secretNumber);
```
Logical Operators

In order of precedence:

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
<th>Example</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>!</td>
<td>not</td>
<td>!(2 == 3)</td>
<td>true</td>
</tr>
<tr>
<td>&amp;&amp;</td>
<td>and</td>
<td>(2 == 3) &amp;&amp; (-1 &lt; 5)</td>
<td>false</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>or</td>
</tr>
</tbody>
</table>

Cannot "chain" tests as in algebra; use && or || instead

// assume x is 15
2 <= x <= 10
true  <= 10
Error!  

// correct version
2 <= x && x <= 10
ture  && false
false
Precedence:

! > arithmetic > comparison > logical

\[ 5 \times 7 \geq 3 + 5 \times (7 - 1) \land \lnot \text{false} \]
Precedence Madness

Precedence:

! > arithmetic > comparison > logical

5 * 7 >= 3 + 5 * (7 - 1) && !false
5 * 7 >= 3 + 5 * 6 && !false
Precedence Madness

Precedence:

! > arithmetic > comparison > logical

5 * 7 >= 3 + 5 * (7 - 1) && !false
5 * 7 >= 3 + 5 * 6 && !false
5 * 7 >= 3 + 5 * 6 && true
Precedence Madness

Precedence:

! > arithmetic > comparison > logical

5 * 7 >= 3 + 5 * (7 - 1) && !false
5 * 7 >= 3 + 5 * 6 && !false
5 * 7 >= 3 + 5 * 6 && true
35 >= 3 + 30 && true
Precedence:

! > arithmetic > comparison > logical

\[5 \times 7 \geq 3 + 5 \times (7 - 1) \land \lnot false\]

\[5 \times 7 \geq 3 + 5 \times 6 \land \lnot false\]

\[5 \times 7 \geq 3 + 5 \times 6 \land true\]

\[35 \geq 3 + 30 \land true\]

\[35 \geq 33 \land true\]
Precedence:

! > arithmetic > comparison > logical

\[
5 \times 7 \geq 3 + 5 \times (7 - 1) \land \land \neg \text{false}
\]

\[
5 \times 7 \geq 3 + 5 \times 6 \land \land \neg \text{false}
\]

\[
5 \times 7 \geq 3 + 5 \times 6 \land \land \text{true}
\]

\[
35 \geq 3 + 30 \land \land \text{true}
\]

\[
35 \geq 33 \land \land \text{true}
\]

\[
\text{true} \land \land \text{true}
\]
Precedence:

! > arithmetic > comparison > logical

5 * 7 >= 3 + 5 * (7 - 1) && !false
5 * 7 >= 3 + 5 * 6 && !false
5 * 7 >= 3 + 5 * 6 && true
35 >= 3 + 30 && true
35 >= 33 && true
true && true
true

Never write code like this 😊
George Boole

English Mathematician 1815 – 1864
Boole died of being too cool

Piech, CS106A, Stanford University
// Store expressions that evaluate to true/false

boolean x = 1 < 2;    // true

boolean y = 5.0 == 4.0; // false
// Store expressions that evaluate to true/false
boolean x = 1 < 2;       // true
boolean y = 5.0 == 4.0;   // false

// Directly set to true/false
boolean isFamilyVisiting = true;
boolean isRaining = false;
// Store expressions that evaluate to true/false
boolean x = 1 < 2; // true
boolean y = 5.0 == 4.0; // false

// Directly set to true/false
boolean isFamilyVisiting = true;
boolean isRaining = false;

// Ask the user a true/false (yes/no) question
boolean playAgain = readBoolean("Play again?", "y", "n");
if (playAgain) {
...
Please...

NO FOOD OR DRINKS

*know your logical precedence

Piech, CS106A, Stanford University
Today’s Route

Simple Java

You are here

Game Show

For Loops

Conditions

Review

The River of Java

Piech, CS106A, Stanford University
Today’s Route

You are here

Simple Java

Review

Conditions

For Loops

Game Show

The River of Java

Piech, CS106A, Stanford University
How would you println “Stanford rocks socks” 100 times
public void run() {
    for(int i = 0; i < 100; i++) {
        println(“Stanford rocks socks!”);
    }
}
for(int i = 0; i < 100; i++) {
    println("Stanford rocks socks!");
}
```java
for(int i = 0; i < 3; i++) {
    println("Stanford rocks socks!");
}
```
for(int i = 0; i < 3; i++) {
    println("Stanford rocks socks!");
}
For Loop Redux

\[ i \quad 0 \]

\[
\text{for}(\text{int} \ i = 0; \ i < 3; \ i++) \ { \\
\quad \text{println(“Stanford rocks socks!”);} \\
}\]

Piech, CS106A, Stanford University
for (int i = 0; i < 3; i++) {
    println("Stanford rocks socks!");
}
```java
for (int i = 0; i < 3; i++) {
    println("Stanford rocks socks!");
}
```
For Loop Redux

```java
for (int i = 0; i < 3; i++) {
    println("Stanford rocks socks!");
}
```

Output:

```
Stanford rocks socks
```

Piech, CS106A, Stanford University
For Loop Redux

```
for(int i = 0; i < 3; i++) {
    println("Stanford rocks socks!");
}
```

Stanford rocks socks
For Loop Redux

```
for(int i = 0; i < 3; i++) {
    println("Stanford rocks socks!");
}
```

Output:
```
Stanford rocks socks
Stanford rocks socks
```
```
for(int i = 0; i < 3; i++) {
    println("Stanford rocks socks!");
}
```
For Loop Redux

\[ i \quad 2 \]

\begin{verbatim}
for(int i = 0; i < 3; i++) {
    println("Stanford rocks socks!");
}
\end{verbatim}

Stanford rocks socks
Stanford rocks socks
For Loop Redux

```java
for (int i = 0; i < 3; i++) {
    println("Stanford rocks socks!");
}
```

Output:

```
Stanford rocks socks
Stanford rocks socks
Stanford rocks socks
```
For Loop Redux

for (int i = 0; i < 3; i++) {
    println("Stanford rocks socks!");
}

Stanford rocks socks
Stanford rocks socks
Stanford rocks socks
For Loop Redux

```
for (int i = 0; i < 3; i++) {
    println("Stanford rocks socks!");
}
```

Output:
```
Stanford rocks socks
Stanford rocks socks
Stanford rocks socks
```
for (int i = 0; i < 3; i++) {
    println("Stanford rocks socks!");
}
for(int i = 0; i < 3; i++) {
    println("Stanford rocks socks!");
}
You can use the for loop variable
How would you println the first 100 even numbers?
Printing Even Numbers

0
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38

Piech, CS106A, Stanford University
for(int i = 0; i < NUM_NUMS; i++) {
    println(i * 2);
}
for(int i = 0; i < 3; i++) {
    println(i * 2);
}
Printing Even Numbers

```java
for(int i = 0; i < 3; i++) {
    println(i * 2);
}
```
Printing Even Numbers

\[
\begin{array}{c|c}
  i & 0 \\
\end{array}
\]

```java
for (int i = 0; i < 3; i++) {
    println(i * 2);
}
```

Piech, CS106A, Stanford University
Printing Even Numbers

\[
\text{i} \quad 0
\]

\[
\text{for(int } i = 0; i < 3; i++) {
    \text{println}(i \times 2);
}\]

Piech, CS106A, Stanford University
Printing Even Numbers

```
for (int i = 0; i < 3; i++) {
    println(i * 2);
}
```
Printing Even Numbers

```
for(int i = 0; i < 3; i++) {
    println(i * 2);
}
```

- i 1
Printing Even Numbers

```java
for (int i = 0; i < 3; i++) {
    println(i * 2);
}
```

| i | 1 |

For Loop Redux

0
Printing Even Numbers

\begin{center}
\begin{tabular}{|c|c|}
\hline
\textit{i} & 1 \\
\hline
\end{tabular}
\end{center}

```java
for(int i = 0; i < 3; i++) {
    println(i * 2);
}
```

Piech, CS106A, Stanford University
Printing Even Numbers

```
for(int i = 0; i < 3; i++) {
    println(i * 2);
}
```

```
0
2
```
Printing Even Numbers

```
for(int i = 0; i < 3; i++) {
    println(i * 2);
}
```

Piech, CS106A, Stanford University
Printing Even Numbers

```
for(int i = 0; i < 3; i++) {
    println(i * 2);
}
```

```
0
2
4
```
Printing Even Numbers

```java
for(int i = 0; i < 3; i++) {
    println(i * 2);
}
```

```
0
2
4
```
Printing Even Numbers

```
for(int i = 0; i < 3; i++) {
    println(i * 2);
}
```

0
2
4
Printing Even Numbers

for (int i = 0; i < 3; i++) {
    println(i * 2);
}

0
2
4
Printing Even Numbers

```java
for(int i = 0; i < 3; i++) {
    println(i * 2);
}
```

Output:

```
0
2
4
```
Today’s Route

You are here

Simple Java

The River of Java

Review

Conditions

For Loops

Game Show

You are here
Today’s Route

Simple Java

Game Show

For Loops

Conditions

Review

You are here

The River of Java
Game Show

Welcome to the CS106A game show!
Choose a door and win a prize
Door: 2
You chose door 2
You win $
Choose a Door

```java
int door = readInt("Door: ");
// while the input is invalid
while (door < 1 || door > 3) {
    // tell the user the input was invalid
    println("Invalid door!");
    // ask for a new input
    door = readInt("Door: ");
}
```

|| or

&& and
The Door Logic

```java
int prize = 4;
if (door == 1) {
    prize = 2 + 9 / 10 * 100;
} else if (door == 2) {
    boolean locked = prize % 2 != 0;
    if (!locked) {
        prize += 6;
    }
} else if (door == 3) {
    prize++;
}
```
```java
int prize = 4;
if(door == 1) {
    prize = 2 + 9 / 10 * 100;
} else if(door == 2) {
    boolean locked = prize % 2 != 0;
    if(!locked) {
        prize += 6;
    }
} else if(door == 3) {
    prize++;  
}
```

The Door Logic
The Door Logic

```java
int prize = 4;
if(door == 1) {
    prize = 2 + 9 / 10 * 100;
} else if(door == 2) {
    boolean locked = prize % 2 != 0;
    if(!locked) {
        prize += 6;
    }
} else if(door == 3) {
    prize++;
}
```
int prize = 4;
if(door == 1) {
    prize = 2 + 9 / 10 * 100;
} else if(door == 2) {
    boolean locked = prize % 2 != 0;
    if(!locked) {
        prize += 6;
    }
} else if(door == 3) {
    prize++;
**The Door Logic**

```java
int prize = 4;
if (door == 1) {
    prize = 2 + 9 / 10 * 100;
} else if (door == 2) {
    boolean locked = prize % 2 != 0;
    if (!locked) {
        prize += 6;
    }
} else if (door == 3) {
    prize++;
}
```
Today’s Goal

1. Be able to use For / While / If in Java