Chapter 3: Selections and Conditionals

CS1: Java Programming
Colorado State University

Original slides by Daniel Liang
Modified by Kristina Brown, Ben Say and Wim Bohm
Exam Results

- 78% Average
- 2 Perfect scores
- Best results from the past two semesters on Midterm 1
Motivation

- Raise your hand if you are wearing a red sweater.
- How do you get from the stadium to the CS Building?
Conditionals

Conditionals allow you to **Make Decisions** using your programs.

```java
if (condition)
    statement;
```
The `boolean` Type and Operators

- Often in a program you need to compare two values, such as whether `i` is greater than `j`.
- Java provides six comparison operators (also known as relational operators) that can be used to compare two values.
- The result of the comparison is a Boolean value: true or false.

```java
boolean b = (1 > 2);
```
# Relational Operators

<table>
<thead>
<tr>
<th>Java Operator</th>
<th>Mathematics Symbol</th>
<th>Name</th>
<th>Example (radius is 5)</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;</td>
<td>&lt;</td>
<td>less than</td>
<td>radius &lt; 0</td>
<td>false</td>
</tr>
<tr>
<td>&lt;=</td>
<td>≤</td>
<td>less than or equal to</td>
<td>radius &lt;= 0</td>
<td>false</td>
</tr>
<tr>
<td>&gt;</td>
<td>&gt;</td>
<td>greater than</td>
<td>radius &gt; 0</td>
<td>true</td>
</tr>
<tr>
<td>&gt;=</td>
<td>≥</td>
<td>greater than or equal to</td>
<td>radius &gt;= 0</td>
<td>true</td>
</tr>
<tr>
<td>==</td>
<td>=</td>
<td>equal to</td>
<td>radius == 0</td>
<td>false</td>
</tr>
<tr>
<td>!=</td>
<td>≠</td>
<td>not equal to</td>
<td>radius != 0</td>
<td>true</td>
</tr>
</tbody>
</table>
One-way if Statements

```java
if (radius >= 0) {
    area = radius * radius * PI;
    System.out.println("The area " + " for the circle of radius " + radius + " is " + area);
}
```

Diagram:
- If the boolean-expression is true, execute the statement(s).
- If the boolean-expression is false, do nothing.

Diagram:
- If (radius >= 0), calculate the area as `area = radius * radius * PI`.
- Print the area with `System.out.println("The area for the circle of" + " radius " + radius + " is " + area);`.

Note – Basic Syntax

```java
if i > 0 {
    System.out.println("i is positive");
}
```
(a) Wrong

```java
if (i > 0) {
    System.out.println("i is positive");
}
```
(b) Correct

```java
if (i > 0) {
    System.out.println("i is positive");
}
```
(a)

```java
if (i > 0)
    System.out.println("i is positive");
```
Equivalent

```java
if (i > 0)
    System.out.println("i is positive");
```
(b)
The Two-way if Statement

if (boolean-expression) {
    statement(s) - for-the-true-case;
}
else {
    statement(s) - for-the-false-case;
}
if-else Example

```java
if (radius >= 0) {
    area = radius * radius * PI;
    System.out.println("The area for the " + "circle of radius " + radius + " is " + area);
}
else {
    System.out.println("Negative input");
}
```
In use...

- How could we use conditionals to create a program to calculate the letter grade from a number?
Multiple Alternative if Statements

```java
if (score >= 90.0)
    System.out.print("A");
else
    if (score >= 80.0)
        System.out.print("B");
    else
        if (score >= 70.0)
            System.out.print("C");
        else
            if (score >= 60.0)
                System.out.print("D");
            else
                System.out.print("F");
```

Equivalent

```java
if (score >= 90.0)
    System.out.print("A");
else if (score >= 80.0)
    System.out.print("B");
else if (score >= 70.0)
    System.out.print("C");
else if (score >= 60.0)
    System.out.print("D");
else
    System.out.print("F");
```

This is better
Multi-Way if-else Statements

- score >= 90
  - grade is A
- score >= 80
  - grade is B
- score >= 70
  - grade is C
- score >= 60
  - grade is D
- score < 60
  - grade is F
Trace if-else statement

Suppose score is 70.0

```java
if (score >= 90.0)
    System.out.print("A");
else if (score >= 80.0)
    System.out.print("B");
else if (score >= 70.0)
    System.out.print("C");
else if (score >= 60.0)
    System.out.print("D");
else
    System.out.print("F");
```

The condition is false
Trace if-else statement

Suppose score is 70.0

if (score >= 90.0)
    System.out.print("A");
else if (score >= 80.0)
    System.out.print("B");
else if (score >= 70.0)
    System.out.print("C");
else if (score >= 60.0)
    System.out.print("D");
else
    System.out.print("F");

The condition is false
Trace if-else statement

Suppose score is 70.0

if (score >= 90.0)
   System.out.print("A");
else if (score >= 80.0)
   System.out.print("B");
else if (score >= 70.0)
   System.out.print("C");
else if (score >= 60.0)
   System.out.print("D");
else
   System.out.print("F");

The condition is true
Trace if-else statement

Suppose score is 70.0

if (score >= 90.0)
    System.out.print("A");
else if (score >= 80.0)
    System.out.print("B");
else if (score >= 70.0)
    System.out.print("C");
else if (score >= 60.0)
    System.out.print("D");
else
    System.out.print("F");

grade is C
if (score >= 90.0)  
    System.out.print("A");
else if (score >= 80.0)  
    System.out.print("B");
else if (score >= 70.0)  
    System.out.print("C");
else if (score >= 60.0)  
    System.out.print("D");
else  
    System.out.print("F");
Dangling else

The `else` clause matches the most recent `if` clause in the same block.

```
int i = 1, j = 2, k = 3;
if (i > j)
  if (i > k)
    System.out.println("A");
else
  System.out.println("B");
```

Equivalent
```
int i = 1, j = 2, k = 3;
if (i > j)
  if (i > k)
    System.out.println("A");
else
  System.out.println("B");
```

This is better with correct indentation.
Dangling else example

```java
int i = 1;
int j = 2;
int k = 3;
if (i > j)
    if (i > k)
        System.out.println("A");
    else
        System.out.println("B");
```

The else matches the closest unmatched if, even if the indentation is bad.

So what is printed here?
Dangling else cont’d

To force the `else` clause to match the first `if` clause, you must add a pair of braces (to finish the deepest `if` statement):

```java
int i = 1;
int j = 2;
int k = 3;
if (i > j) {
    if (i > k)
        System.out.println("A");
}
else
    System.out.println("B");
```

So what does this statement print?
Common Errors

Adding a semicolon at the end of an if clause is a common mistake.

```
if (radius >= 0); // Wrong
{
    area = radius*radius*PI;
    System.out.println("The area for the circle of radius " +
                     radius + " is " + area);
}
```

This mistake is hard to find, because it is not a compilation error or a runtime error, it is a logic error.
TIP

There’s always a couple of ways
To do things in programming

if (number % 2 == 0)
   even = true;
else
   even = false;

Equivalent

boolean even
   = number % 2 == 0;

(a)  (b)
Problem: Computing Taxes

The US federal personal income tax is calculated based on the filing status and taxable income. There are four filing statuses: single filers, married filing jointly, married filing separately, and head of household. The tax rates for 2009 are shown below.

<table>
<thead>
<tr>
<th>Marginal Tax Rate</th>
<th>Single</th>
<th>Married Filing Jointly or Qualifying Widow(er)</th>
<th>Married Filing Separately</th>
<th>Head of Household</th>
</tr>
</thead>
<tbody>
<tr>
<td>10%</td>
<td>$0 – $8,350</td>
<td>$0 – $16,700</td>
<td>$0 – $8,350</td>
<td>$0 – $11,950</td>
</tr>
<tr>
<td>15%</td>
<td>$8,351 – $33,950</td>
<td>$16,701 – $67,900</td>
<td>$8,351 – $33,950</td>
<td>$11,951 – $45,500</td>
</tr>
<tr>
<td>35%</td>
<td>$372,951+</td>
<td>$372,951+</td>
<td>$186,476+</td>
<td>$372,951+</td>
</tr>
</tbody>
</table>
Problem: Computing Taxes, cont.

```java
if (status == 0) {
    // Compute tax for single filers
}
else if (status == 1) {
    // Compute tax for married file jointly
    // or qualifying widow(er)
}
else if (status == 2) {
    // Compute tax for married file separately
}
else if (status == 3) {
    // Compute tax for head of household
}
else {
    // Display wrong status
}
```
Your Turn!

Given an int number, e.g.:
int number = 10;

Write code that, if the number is a multiple of 5, it prints HiFive, and if the number is divisible by 2, it prints HiEven.