Interactive Programs

with Scanner

reading: 3.3 - 3.4

Interactive programs

interactive program: Reads input from the console.

• While the program runs, it asks the user to type input.
• The input typed by the user is stored in variables in the code.
• Can be tricky; users are unpredictable and misbehave.
• But interactive programs have more interesting behavior.

Scanner

• Scanner: An object that can read input from many sources.
• Communicates with System.in
• Can also read from files (Ch. 6), web sites, databases, ...
• The Scanner class is found in the java.util package.

import java.util.*;

// so you can use Scanner

Scanner console = new Scanner(System.in);

Scanner example

import java.util.*; // so that I can use Scanner

public class UserInputExample {
    public static void main(String[] args) {
        Scanner console = new Scanner(System.in);
        System.out.print("How old are you? ");
        int age = console.nextInt();
        System.out.println("You typed "+ age);
        int years = 65 - age;
        System.out.println(years + " years until retirement!");
    }
}

Scanner methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nextInt()</td>
<td>reads an int from the user and returns it</td>
</tr>
<tr>
<td>nextDouble()</td>
<td>reads a double from the user</td>
</tr>
<tr>
<td>next()</td>
<td>reads a one-word String from the user</td>
</tr>
<tr>
<td>nextLine()</td>
<td>reads a one-line String from the user</td>
</tr>
</tbody>
</table>

• Each method waits until the user presses Enter.
• The value typed by the user is returned.
• prompt: A message telling the user what input to type.
import java.util.*; // so that I can use Scanner

public class ScannerMultiply {
    public static void main(String[] args) {
        Scanner console = new Scanner(System.in);
        System.out.print("Please type two numbers: ");
        int num1 = console.nextInt();
        int num2 = console.nextInt();
        int product = num1 * num2;
        System.out.println("The product is "+ product);
    }
}

Output (user input underlined):
Please type two numbers: 8 6
The product is 48

The Scanner can read multiple values from one line.

If many methods need to read input, declare a Scanner in main and pass it to the other methods as a parameter.

public static int readSum3(Scanner console) {
    System.out.print("Type 3 numbers: ");
    int num1 = console.nextInt();
    int num2 = console.nextInt();
    int num3 = console.nextInt();
    return num1 + num2 + num3;
}

Example:
double gpa = console.nextDouble();
if (gpa > 2.0) {
    System.out.println("Application accepted.");
    } else {
    System.out.println("Application denied.");
}
Relational expressions

- If statements and for loops both use logical tests.
  
  ```java
  for (int i = 1; i <= 10; i++) {...
  if (i <= 10) {...
  } // These are boolean expressions, seen in Ch. 5.
  ```

- Tests use relational 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>1.2 == 2.5</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>

Misuse of if

- What’s wrong with the following code?

```java
Scanner console = new Scanner(System.in);
System.out.print("What percentage did you earn? ");
int percent = console.nextInt();
if (percent >= 90) {
  System.out.println("You got an A! ");
} else if (percent >= 80) {
  System.out.println("You got a B! ");
} else if (percent >= 70) {
  System.out.println("You got a C! ");
} else if (percent >= 60) {
  System.out.println("You got a D! ");
} else {
  System.out.println("You got an F! ");
}
...```

Nested if/else

- Choose between outcomes using many tests

```java
if (test) {
  statement(s);
} else if (test) {
  statement(s);
} else {
  statement(s);
}
```

- Example:

```java
if (x > 0) {
  System.out.println("Positive");
} else if (x < 0) {
  System.out.println("Negative");
} else {
  System.out.println("Zero");
}
```

Nested if/else/if

- If it ends with else, exactly one path must be taken.

```java
if (test) {
  statement(s);
} else if (test) {
  statement(s);
} else if (test) {
  statement(s);
}
```

- Example:

```java
if (place == 1) {
  System.out.println("Gold medal!");
} else if (place == 2) {
  System.out.println("Silver medal!");
} else if (place == 3) {
  System.out.println("Bronze medal!");
}
```

Nested if structures

- Exactly 1 path (mutually exclusive)

```java
if (test) {
  statement(s);
} else if (test) {
  statement(s);
} else {
  statement(s);
}
```

- 0 or 1 path (mutually exclusive)

```java
if (test) {
  statement(s);
} else if (test) {
  statement(s);
} else if (test) {
  statement(s);
}
```

- 0, 1, or many paths (independent tests; not exclusive)

```java
if (test) {
  statement(s);
}
if (test) {
  statement(s);
}
if (test) {
  statement(s);
}
...```

Which nested if/else?

- (1) if/else if (2) nested if/else if (3) nested if/else if

- Whether a user is lower, middle, or upper-class based on income.

- (2) nested if/else if

  - Whether you made the dean’s list (GPA ≥ 3.8) or honor roll (3.5-3.8).

  - (3) nested if/else if

- Whether a number is divisible by 2, 3, and/or 5.

  - (1) sequential if/else if

- Computing a grade of A, B, C, D, or F based on a percentage.

  - (2) nested if/else if/else if/else if/else
Nested if/else question

Formula for body mass index (BMI):

\[
BMI = \frac{\text{weight} \times 703}{\text{height}^2}
\]

• Write a program that produces output like the following:

This program reads data for two people and computes their body mass index (BMI).

Enter next person’s information:
height (in inches)?
70.0
weight (in pounds)?
194.25
Person 1 BMI = 27.868928571428572
overweight

Enter next person’s information:
height (in inches)?
62.5
weight (in pounds)?
130.5
Person 2 BMI = 23.485824
normal

Difference = 4.3831045714285715

### BMI

| Weight class | 
|-----|---
| below 18.5 | underweight |
| 18.5 - 24.9 | normal |
| 25.0 - 29.9 | overweight |
| 30.0 and up | obese |

### Nested if/else answer

// This program computes two people’s body mass index (BMI) and compares them. The code uses Scanner for input, and parameters/returns.
import java.util.*;

// This program computes two people’s body mass index (BMI) and compares them. The code uses Scanner for input, and parameters/returns.
public class BMI {
    public static void main(String[] args) {
        introduction();
        Scanner console = new Scanner(System.in);
        double bmi1 = person(console);
        double bmi2 = person(console);
        // report overall results
        report(1, bmi1);
        report(2, bmi2);
        System.out.println("Difference = " + Math.abs(bmi1 - bmi2));
    }

    // prints a welcome message explaining the program
    public static void introduction() {
        System.out.println("This program reads data for two people and computes their body mass index (BMI)." + 
        System.out.println("Please enter the next person’s information:
        System.out.println("Enter next person’s information:
        height (in inches)?
        System.out.println("weight (in pounds)?
        System.out.println("Person 1 BMI = " + bmi1 + 
        System.out.println("Person 2 BMI = " + bmi2 + 
        System.out.println("Difference = " + Math.abs(bmi1 - bmi2));
    }

    // reads information for one person, computes their BMI, and returns it
    public static double person(Scanner console) {
        System.out.println("Enter next person’s information:
        System.out.print("height (in inches)? 
        double height = console.nextDouble();
        System.out.print("weight (in pounds)? 
        double weight = console.nextDouble();
        System.out.println();
        double bodyMass = bmi(height, weight);
        return bodyMass;
    }

    // Computes/returns a person’s BMI based on their height and weight.
    public static double bmi(double height, double weight) {
        return (weight * 703 / height / height);
    }

    // Outputs information about a person’s BMI and weight status.
    public static void report(int number, double bmi) {
        System.out.println("Person " + number + " BMI = " + bmi);
        if (bmi < 18.5) {
            System.out.println("underweight");
        } else if (bmi < 25) {
            System.out.println("normal");
        } else if (bmi < 30) {
            System.out.println("overweight");
        } else {
            System.out.println("obese");
        }
    }
}

Copyright 2010 by Pearson Education