Design Simple Programs with Input and Output

Hui Chen

Department of Computer & Information Science

Brooklyn College

Objectives

- To design Java programs to perform simple computations (§2.2).
- To obtain input from the console using the **Scanner** class (§2.3).
- To use identifiers to name variables, constants, methods, and classes (§2.4).
- To use variables to store data (§§2.5–2.6).
- To program with assignment statements and assignment expressions (§2.6).

Outline

- Motivation
- From “problem”, to “algorithm”, and to “implementation”
- Design a program with input and output
  - Hardcode input
  - Read from users’ input (from console)
- Dissecting the program
- Exercises/labs

Motivations

- Why do we program?
  - Solve problems

Problem: Compute Area of a Shape

- Computing the area of a shape is a common problem we face in real life
    - How big is the apartment?
    - is it “cheaper” to buy an extra large pizza or to buy large pizza?
    - Is the room big enough so that we can maintain a safe distance?

Problem to Solution: Compute Area of a Circle

- Solution: an algorithm
  - Read in radius (hard code in the program)
  - Compute area
  - Display the area

Solution to Code: Compute Area of a Circle

- Let’s implement the solution in Java

```
public class ComputeArea {
  public static void main(String[] args) {
    double radius;
    double area;

    // Assign a radius
    radius = 20;

    // Compute area
    area = radius * radius * 3.14159;

    // Display results
    System.out.println("The area for the circle of radius " +
      radius + " is " + area);
  }
}
```

Trace the program

- Compile
- Run
- Trace the program

Questions?

- Any questions?
- What if we want to compute the area of a different circle?

Reading Input from Console

- Reading Input from the Console
  - Create a Scanner object
  - Scanner input = new Scanner(System.in);
- Use the method nextDouble() to obtain to a double value. For example,
  - System.out.print("Enter a double value: ");
  - Scanner input = new Scanner(System.in);
  - double d = input.nextDouble();

Locating the Scanner class

- Using the fully qualified class name
- Using import
  - Implicit import
    - import java.util.* ; // Implicit import
  - Explicit import
    - Import java.util.Scanner; // Explicit Import

Computing the Area of Circle Again!

Questions?

Dissecting the Example Programs

- Identifiers
- Variables
- Assignment statements

Identifiers

- An identifier is a sequence of characters that consist of letters, digits, underscores (_), and dollar signs ($).
- An identifier must start with a letter, an underscore (_), or a dollar sign ($). It cannot start with a digit.
- An identifier cannot be a reserved word.
  - See Appendix A of the textbook, “Java Keywords,” for a list of reserved words.
- An identifier cannot be true, false, or null (they are not keywords, but you cannot use them to name identifers).
- An identifier can be of any length.

Variables

- Represent values that may be changed in the program

Examples of Variables

```
// Compute the first area
radius = 1.0;
area = radius * radius * 3.14159;
System.out.println("The area is “ + area + " for radius "+radius);

// Compute the second area
radius = 2.0;
area = radius * radius * 3.14159;
System.out.println("The area is “ + area + " for radius "+radius);
```

Declaring Variables

```
int x;     // Declare x to be an
           // integer variable;
double radius; // Declare radius to
               // be a double variable;
char a;     // Declare a to be a
            // character variable;
```

Assignment

```
x = 1;          // Assign 1 to x;
radius = 1.0;   // Assign 1.0 to radius;
a = 'A';        // Assign 'A' to a;
```

Declaring and Initializing in One Step

```
int x = 1;

double d = 1.4;
```

Questions?

Lab Exercise

- Compute GPA given GPA value (instead of letter grades) of two courses and credit hours obtained from user's input
  - See (PDF) pages 62 in the Undergraduate Bulletin
  - GPA $= \frac{(GPA\,Value\ 1)\times(Credits\ 1)+(GPA\,Value\ 2)\times(Credits\ 2)}{(Credits\ 1)+(Creits\ 2)}$