
CSE 142
Simple Input and Output

2/7/2002 (c) University of Washington 2001-2 H-1

Overview

- **Topics**
 - Communicating with the outside world
 - Output to System.out
 - Simple input for CSE142
- **Reading**
 - Dugan notes: end of Ch. 5

2/7/2002 (c) University of Washington 2001-2 H-2

Input & Output

- Most interesting programs perform both *input* and *output*.
 - Think about a video game with no display, or no joystick.
 - Or an ATM that doesn't let you enter your PIN!
- Output can go to a variety of places:
 - the screen, speakers, disk, network, printer...
- Input can come from a variety of places:
 - the mouse, keyboard, disk, network...
- We know how to do some output (painting shapes), but what about input or other kinds of output?

2/7/2002 (c) University of Washington 2001-2 H-3

Text Output: System.out

- Java provides an object, **System.out**, which has methods to print text to the console window.
- Some operations:

```
System.out.println(<expression> ); // print argument, then advance to next line
System.out.print(<expression> ); // print argument without terminating
// the current line
```
- Argument can be an int, double, char, boolean, or String (or any object – more below)
- Several calls of the print method can be used to print multiple values on one line.

```
System.out.print("Good "); System.out.print("Morning ");
System.out.println("Class ");
```

2/7/2002 (c) University of Washington 2001-2 H-4

Printing Objects on System.out

- Any object can be printed on System.out

```
Rectangle rect = new Rectangle(30,50,100,150,Color.blue,true);
System.out.println(rect);
```
- Can be very useful for debugging
 - Put System.out.print or println method calls in your code to display a message when that place is reached during execution
 - Particularly useful if the string version of the object has useful information in a readable format

2/7/2002 (c) University of Washington 2001-2 H-5

Object Representation on System.out

- What really happens when an object is printed
 - The toString method belonging to the object is used to get the string to be printed
 - All classes have a default toString() (not very descriptive)
 - You can provide a custom version of toString() in your classes

```
/* Return a string representation of this bank account */
public String toString() {
    String description = "BankAccount(name = " + this.accountName +
        ", account # = " + this.accountNumber +
        ", balance = " + this.balance + ")";
    return description;
}
```

2/7/2002 (c) University of Washington 2001-2 H-6

Input in Java

- Input in native Java is a bit complicated.
- For CSE142, there's a local library in `uwcse.jar` that contains a class named `Input`.
- When you need to do input, create a *single* object of type `Input`:

```
Input in = new Input();
```
- Then send the object appropriate messages. It knows how to get data from the keyboard.
 - See the documentation page on the web.

2/7/2002

(c) University of Washington 2001-2

H-7

Input: Some Messages

- Here's a simple message understood by `Input` objects:

```
/* Display message and return integer input; repeat until valid integer entered.
 * @param prompt message to display with input request
 * @return integer value typed by user */
int readInt(String prompt);
```
- There are similar messages for reading values of other simple types

```
String readString (String prompt);
double readDouble(String prompt);
char readChar (String prompt);
```
- To read many values, create one `Input` object, then call its methods as many times as needed

2/7/2002

(c) University of Washington 2001-2

H-8

Input: An Example

- An example interaction with the interpreter:

```
JevaESCL-0> Input in = new Input();
JevaESCL-1> int width = in.readInt("What is the width of the box?");
What is the width of the box?
8
JevaESCL-2> int height = in.readInt("What is the height?");
What is the height?
hello
Not a valid integer, please try again.
4
JevaESCL-3> height * width
32
```
- Also works in BlueJ, using a separate *console (terminal) window*.
 - Must import the I/O library when writing a class that uses `Input`, e.g.:

```
import uwcse.io.*;
```

2/7/2002

(c) University of Washington 2001-2

H-9