

CSE 20

Intro to Computing I

Lecture 3 – Variables (cont.)

Type Conversion



Announcements

- ▶ Today: Variables (cont.), Type Conversion
- ▶ Labs
 - Lab 2 due this week (9/22 – 9/28) with an additional 3 days grace period
 - Lab 3 (Input Output) assigned this week
 - Due in one week (plus additional **3 days** grace period)
 - Make sure to demo your work to a TA (or me) after submission
 - Demo is REQUIRED to receive credit for assignment
- ▶ Reading Assignments
 - Reading 01 (1.1 – 1.11, 2.1 – 2.5) due tonight (Sep 23) at midnight
 - Reading 02 (2.6 – 2.18, 2.20) due Oct 7
 - Complete Participation Activities in each section to receive grade towards Participation
 - IMPORTANT: Make sure to **submit score to CatCourses** by using link provided on CatCourses

Outputs in Java (review)

- ▶ Outputs: ways a computer to communicate with us
 - Displays (monitors), printers, speakers...
- ▶ To display a statement on a monitor:

```
System.out.print("Test print\t");
```

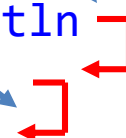
```
System.out.println("Test println\n");
```

```
System.out.print("Done");
```

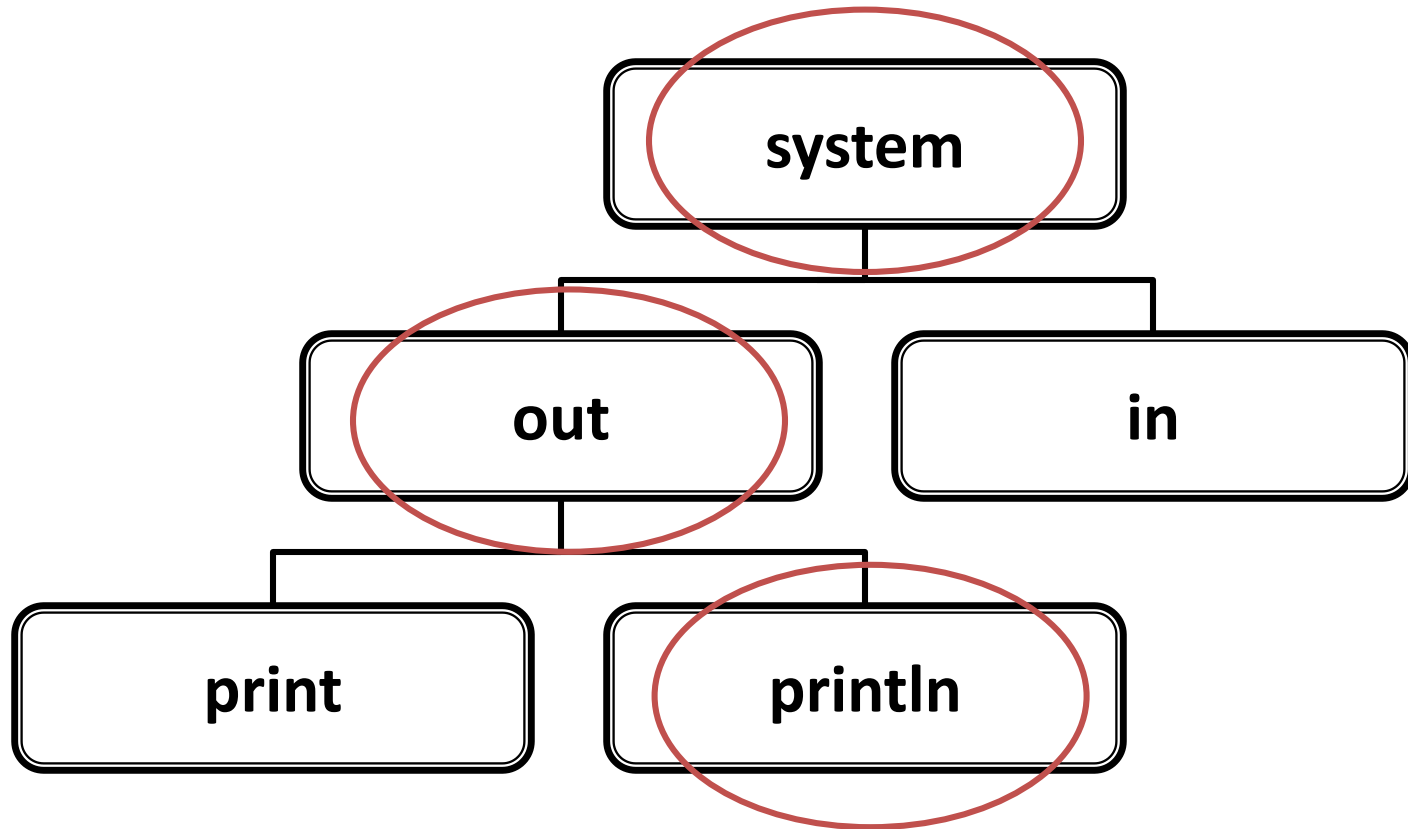
Output:

Test printTest println

Done



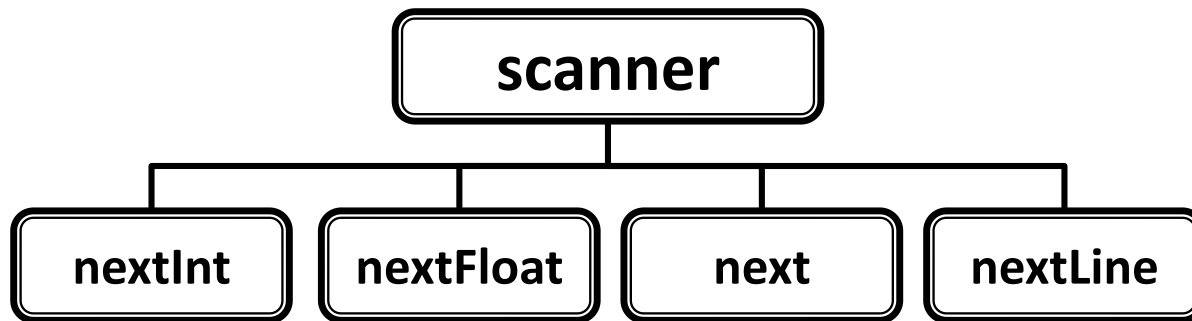
System Objects in Java (review)



System.out.println("World");

Input from keyboard – Scanner (review)

- ▶ We can interact with the program using input devices:
 - **Keyboards**, mice, microphones



```
Scanner input = new Scanner(System.in);  
input.nextInt();  
input.nextFloat();  
input.next();  
input.nextLine();
```

We will learn more about
this in future labs

Data Types (review)

- ▶ **boolean**: 1-bit
 - 2 values, range : 0-1
- ▶ **byte**: 8 bits (2 bytes)
 - 2^8 values, range : -128 to 127
- ▶ **short**: 16-bits (2 bytes)
 - 2^{16} values, range : -32,768 to 32,767
- ▶ **char**: 16-bits (2 bytes)
 - 2^{16} values, range : 0 to 65,535
- ▶ **int**: 32-bits (4 bytes)
 - 2^{32} values, range : -2,147,483,648 to +2,147,483,647
- ▶ **float**: 32-bits (4 bytes)
 - Scientific format : $\pm 3.4 \times 10^{\pm 38}$
- ▶ **long**: 64-bits (8 bytes)
 - 2^{64} values, range : -2^{63} to $+2^{63} - 1$
- ▶ **double** : 64-bits (8 bytes)
 - $\pm 1.7 \times 10^{\pm 308}$
- ▶ **String** : Any length (string of characters)

Variables (review)

- ▶ Add names/identifiers to each as a way of referring to them.
 - They can be any word.
 - Try to choose the names that make sense.
- ▶ Need to know the data types.



Variables

- ▶ "first" can be of what type?
 - short, integer, float, double
- ▶ "second" can be of what type?
 - float, double
- ▶ "result" can be of what type?
 - float, double



Code – Executable Statements

```
int first = 1;
```

```
double second = 0.5;
```

```
double result = first - second;
```



Code – Executable Statements

```
int first;           // Declaration (type name)
first = 0;           // Assignment (initialize)
first = 1;           // Assignment (reuse/override)
double second = 0.5; // Declare + Assign
double result = first - second;
```



Type Casting (Up Conversion)

`double first;`

`first = 0;`

`first = 1;`

`double second = 0.5;`

`double result = first - second;`

`// Use "higher" type`

`// 0 is also a valid double (0.0)`

`// 1.0`

Up Conversion -> no information loss



Type Casting - Down Conversion

```
double first;
```

```
first = 0;
```

```
first = 1;
```

```
double second = 0.5;
```

```
int result = (int)(first - second); // Using "lower" type needs  
// explicit cast
```

Forced it to be an **int** (explicitly)

A **double**

Down Conversion -> possible information loss



Type Conversions

► Implicit – Up conversion

- *double* d = 4; d is 4.0
- *char* a = '}' ; a is '}'
- *int* i = 'A'; i is 65
- *float* f = 'A'; f is 65.0
- *double* e = 'A'; e is 65.0

There is no loss of information

► Explicit – Down conversion

- a = (*char*)i; a is A
- a = (*char*)f; a is A
- a = (*char*)d; a is EOT
- i = (*int*)f; i is 65
- i = (*int*)e; i is 65
- f = (*float*)e; f is 65.0

There may be loss of information