

LEC 01

CSE 121

Printing, Strings, Variables, Datatypes, Expressions

Questions during Class?

Raise hand or send here

sli.do #cse121



BEFORE WE START

*Talk to your neighbors:**What is your favorite handheld food?***Respond on sli.do!**

Instructor: Hayden Feeney**TAs:** Anisha Savannah William
Cayden Tamsyn

Agenda (1/7)

- **Announcements, Reminders**
- HelloWorld Review
- Printing, Strings, Variables Review
- Variables Example
- Datatypes and Expressions Review
- Expressions Practice
- C0 Overview

Announcements, Reminders

- Links to all materials and activities are available on the [website](#)
- **Creative Project 0 (C0)** will be released tonight
 - due Wednesday, July 1st
- Fill out the [welcome survey](#) by **tonight** at 11:59 PM
 - gives us valuable insight!
 - important that Google Forms works in CSE 121
 - Google Forms [troubleshooting guide](#) on website
- New Ed tool: [Sandbox](#) (write all the code you want!)
- First PCM is much longer than future PCMs

Aside: TA Office Hours!

The [IPL](#) (TA office hours) will open on Monday (June 29) at 1:30 PM.

- one of the best parts of the course!
- but, TAs are instructed to not just give you the answer!
 - why not? you wouldn't be learning!
 - e.g. “my code doesn't work” versus “I tried X, expected Y, but got Z. Thoughts on what to try next?”
 - also true for instructor office hours ;)

Expect an announcement on Ed with a detailed schedule soon!

Agenda (2/7)

- Announcements, Reminders
- **HelloWorld Review**
- Printing, Strings, Variables Review
- Variables Example
- Datatypes and Expressions Review
- Expressions Practice
- C0 Overview

Code Quality

"Programs are meant to be read by humans and only incidentally for computers to execute." –Abelson & Sussman, SICP

Code is about *communication*. Writing code with good **code quality** is important to communicate effectively.

Different organizations have different *standards* for code quality.

- Doesn't mean that any one standard is wrong! (e.g., APA, MLA, Chicago, IEEE, ...)
- Consistency is very helpful within a project
- See our [Code Quality Guide](#) for the standards we will all use in CSE 121

Agenda (3/7)

- Announcements, Reminders
- HelloWorld Review
- **Printing, Strings, Variables Review**
- Variables Example
- Datatypes and Expressions Review
- Expressions Practice
- C0 Overview

PCM: Printing, Strings

- `System.out.print(...)`
 - Prints the given text to the console
- `System.out.println(...)`
 - Prints the given text to the console, and then moves to the next line
- String literals: a sequence of characters that are *strung* together, begin and end with `""`
 - Example: `"hello"`



Practice: Think

[sli.do](#)

#cse121

How many lines of output would the following code produce?

```
System.out.println("hello");  
System.out.print("bonjour");  
System.out.println("pryvit");  
System.out.print("nihao");  
System.out.println("hola");  
System.out.print("namaste");
```

a) 1

b) 2

c) 3

d) 4

e) 5

f) 6



Practice: Pair

sli.do

#cse121

How many lines of output would the following code produce?

```
System.out.println("hello");  
System.out.print("bonjour");  
System.out.println("pyvrit");  
System.out.print("nihao");  
System.out.println("hola");  
System.out.print("namaste");
```

a) 1

b) 2

c) 3

d) 4

e) 5

f) 6

Escape sequences

Escape sequence: a special sequence of characters used to represent certain special characters in a String.

- `\"` to produce `"` in a String
- `\\` to produce `\` in a String
- `\n` to produce a new line character (or line break) in a String
 - note: in our class, we will ask you not to use this
- and many more!

PCM: Variables

- Variables allow us to give a name to a specific value
 - 3 parts: declaration, initialization, usage
 - Example:

```
String food = "burrito";  
System.out.println(food); // prints burrito
```

Agenda (4/7)

- Announcements, Reminders
- HelloWorld Review
- Printing, Strings, Variables Review
- **Variables Example**
- Datatypes and Expressions Review
- Expressions Practice
- C0 Overview

Agenda (5/7)

- Announcements, Reminders
- HelloWorld Review
- Printing, Strings, Variables Review
- Variables Example
- **Datatypes and Expressions Review**
- Expressions Practice
- C0 Overview

PCM: Data Types & Expressions

- Programming is about data; we tell Java what **type** of data we have!
- Data types (so far): `int`, `double`, `String`, `boolean`
 - note: only `String` is capitalized!
- All values in a (Java) program have a type!
 - some are “obvious”, e.g. `42` or `"hello world"`
 - aside: these are called “**literals**”
 - some are more complicated **expressions**!

PCM: Operators

We learned a *ton* of operators!

Numerical:

- + Addition
- - Subtraction
- * Multiplication
- / Division (tricky!)
- % Modulo (or “mod”)
- <, >, <=, >=, ==, != Relational

Strings:

- + Concatenation (not addition!)

Booleans:

- ! Logical Not
- && Logical And
- || Logical Or
- == and != Relational

PCM: Precedence

Operators have precedence (an order of operations).

In Math:

1. **P**arentheses
2. **E**xponent
3. **M**ultiplication
4. **D**ivision
5. **A**ddition
6. **S**ubtraction

In Java:

1. Parentheses
2. Logical not
3. Multiplication, Modulo, Division
4. Addition (and concatenation), Subtraction
5. Relational operators
6. Equality operators
7. Logical AND
8. Logical OR

Expressions in “little steps”

$$\begin{array}{ccccccc} 5 & + & 2 & * & 4 & & \\ & & \underbrace{} & & & & \\ & & & 8 & & & \\ \underbrace{} & & & & & & \\ & 13 & & & & & \end{array}$$

$$\begin{array}{ccccccc} 1 & + & 2 & / & 3 & & \\ & & \underbrace{} & & & & \\ & & & 0 & & & \\ \underbrace{} & & & & & & \\ & 1 & & & & & \end{array}$$

$$\begin{array}{ccccccc} 6 & * & 5 & \% & 7 & & \\ & & \underbrace{} & & & & \\ & & 30 & & & & \\ & & & \underbrace{} & & & \\ & & & & 2 & & \end{array}$$

PCM: Conversions

When mixing types in an expression, Java will convert one type to the other and then perform the operation “normally”.

Some conversions are straightforward:

- `ints` can be converted to `doubles` (add `.0`)
- `ints` and `doubles` can be converted to `Strings` (add `""`)

So, Java does these for you!

New: Conversions (Gone Wrong!!)

Other conversions are “lossy”, because you lose data.

- e.g. to make 3.14 an `int`, you’d probably pick either 3 or 4 – but either one loses data
- Java won’t do this automatically for you – you need to “ask”.
 - called a **type cast**: you’ll see this in Wednesday’s PCM + in P0

Some conversions don’t make sense.

- how would you convert "Beyoncé" to an `int`? `double`?
- Java really doesn’t let you do these...

Expression example with mixing types

$2 + 2 + \text{"hello"} + 3 * 5 + \text{"10"}$

$\underbrace{2 + 2}_{\text{"4"}}$

$\underbrace{3 * 5}_{\text{"15"}}$

$\underbrace{\text{"4"} + \text{"hello"}}_{\text{"4hello"}}$

$\underbrace{\text{"4hello"} + \text{"15"}}_{\text{"4hello15"}}$

$\underbrace{\text{"4hello15"} + \text{"10"}}_{\text{"4hello1510"}}$

Agenda (6/7)

- Announcements, Reminders
- HelloWorld Review
- Printing, Strings, Variables Review
- Variables Example
- Datatypes and Expressions Review
- **Expressions Practice**
- C0 Overview

Work on Expressions and Types Practice (1)

- Ed lesson linked from course calendar
- Work with folks around you!
- TAs & I will walk around and help!

```
5 * 3 + 1.0
```

```
8 / 3 * 2.0
```

```
8.0 / 3 * 2
```

```
"Hello" + "world"
```

```
1 + "2" + 3
```

```
1 + 2 + "3"
```

```
1 + "2" + (3 + 4)
```

Part 1 Walkthrough (steps)

"Hello" + "world"
"Helloworld"

5 * 3 + 1.0
15.0
16.0

8 / 3 * 2.0
2.0
4.0

8.0 / 3.0 * 2.0
2.666...
5.333...

"1" + "2" + "3"
"12"
"123"

1 + 2 + "3"
"3"
"33"

"1" + "2" + (3 + 4)
"12" "7"
"127"

Work on Expressions and Types Practice (2)

- On the Ed lesson for today, and on your handout
- Work with folks around you!
- TAs & I will walk around and help!

```
5 * 3 < 12
```

```
10 % 3 == 10 / 3
```

```
5 < 9 || (7 != 7)
```

```
!(1 + 2 == 3 && 10 % 4 > 2)
```

Part 2 Walkthrough (1)

$\underbrace{5 * 3}_{15} < 12$
15 < 12
false

$\underbrace{10 \% 3}_1 == 10 / 3$
1 == $\underbrace{10 / 3}_3$
1 == 3
false

$5 < 9 \ || \ \underbrace{(7 \neq 7)}_{\text{false}}$
 $\underbrace{5 < 9}_{\text{true}} \ || \ \text{false}$
true || false
true

Part 2 Walkthrough (2)

!(1 + 2 == 3 && 10 % 4 > 2)

!(1 + 2 == 3 && 2 > 2)

!(3 == 3 && 2 > 2)

!(3 == 3 && false)

!(true && false)

!(false)
true

Agenda (7/7)

- Announcements, Reminders
- HelloWorld Review
- Printing, Strings, Variables Review
- Variables Example
- Datatypes and Expressions Review
- Expressions Practice
- **C0 Overview**

Creative Project 0 (“C0”): Hello Bugs?! (1/2)

By release date of assignment, all the relevant content will be covered.

- e.g. C0 is just about printing, strings, and escape characters

Assignments are *partially* about “does your program work”, but also:

- test your ability to read a specification
- are graded on **code quality**
- include a **graded reflection** (don't leave this to the last minute)
 - some metacognition, some societal impact & ethics content

Creative Project 0 (“C0”): Hello Bugs?! (2/2)

This specific assignment...

- has two parts: “Basic Task” and “Creative Extension” (do both!)
- has an optional set of code quality slides (to help you practice)
- intentionally gentle onboarding to computer programming
 - **is not meant to be time-consuming or stressful**
 - but also, not representative of all assignments (or programs)

We look forward to seeing your bugs!!