

Activity 1: Introduction

In this course, you will work in teams of 3–4 students to learn new concepts. This activity will introduce you to the process. We'll also take a first look at how to store data in Java programs.

Content Learning Objectives

After completing this activity, students should be able to:

- Describe the responsibility of each team role.
- Identify components of the "hello world" program.
- Write Java code to declare int and double variables.
- Explain what it means to assign a value to a variable.

Process Skill Goals

During the activity, students should make progress toward:

- Leveraging prior knowledge and experience of other students. (Teamwork)



Copyright © 2017 Chris Mayfield and Helen Hu. This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

Model 1 Team Roles

Decide who will be what role for today; we will rotate the roles each week. If you have only three people, one should have two roles. If you have five people, two may share the same role.

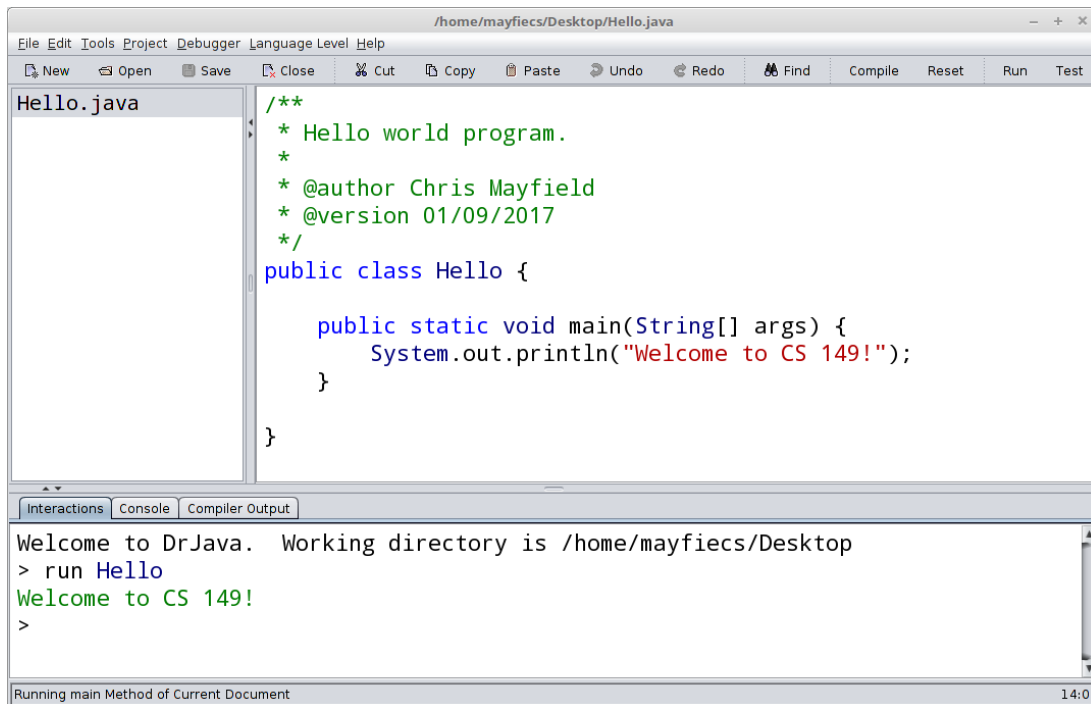
Manager:
Presenter:
Recorder:
Reflector:

Questions (15 min)

Start time: _____

1. What is the difference between **bold** and *italics* on the role cards?
2. Manager: invite each person to explain their role to the team. Recorder: take notes of the discussion by writing down key phrases next to the table above.
3. What responsibilities do two or more roles have in common?
4. For each role, give an example of how someone observing your group would know that a person is not doing their job well.
 - Manager:
 - Presenter:
 - Recorder:
 - Reflector:

Model 2 Hello, World!



The screenshot shows a Java IDE window titled `/home/mayfiecs/Desktop/Hello.java`. The menu bar includes File, Edit, Tools, Project, Debugger, Language Level, and Help. The toolbar contains icons for New, Open, Save, Close, Cut, Copy, Paste, Undo, Redo, Find, Compile, Reset, Run, and Test. The editor displays the following code in `Hello.java`:

```
/**
 * Hello world program.
 *
 * @author Chris Mayfield
 * @version 01/09/2017
 */
public class Hello {

    public static void main(String[] args) {
        System.out.println("Welcome to CS 149!");
    }

}
```

Below the editor is a console window with tabs for Interactions, Console, and Compiler Output. The Console tab is active, showing the following output:

```
Welcome to DrJava. Working directory is /home/mayfiecs/Desktop
> run Hello
Welcome to CS 149!
>
```

The status bar at the bottom indicates "Running main Method of Current Document" and the time "14:0".

Questions (10 min)

Start time: _____

5. What is the name of the class? What is the name of the file? What directory is it in?
6. How many lines of code is the above program? How many statements does it have?
7. What is the purpose of the first six lines? What is the purpose of the two blank lines?
8. Describe in your own words what `System.out.println` does. Be very specific.

Model 3 Variables

Most programs store and manipulate data values, and we use *variables* to give them meaningful names. The following code *declares* and *assigns* three variables. Each variable is stored in the computer's memory, represented by the boxes on the right.

Java code

```
int dollars;  
int cents;  
double grams;  
  
dollars = 1;  
cents = 90;  
grams = 3.5;
```

Computer memory

dollars	1
cents	90
grams	3.5

Questions (10 min)

Start time: _____

9. Identify the Java *keyword* used in a variable declaration to indicate

- a) an integer:
- b) a real number:

10. Consider numbers of dollar bills, cents, and grams. Which of these units only makes sense as an integer, and why?

11. What would you expect the following statements to print out?

- a) `System.out.println(dollars);`
- b) `System.out.println(cents);`
- c) `System.out.println(grams);`

12. In the previous question, how does the third printed line (c) differ from the first two?

13. What do you think is the purpose of a variable declaration?

14. What is output by the following code, and why?

```
double one;  
one = 1;  
System.out.println(one);
```

Model 4 Assignment

Declaring a variable instructs the computer to reserve space for it in memory.

Java code

```
1 int dollars;  
2 int cents;
```

Computer memory

dollars	<input type="text"/>
cents	<input type="text"/>

Variables cannot be used until they are *initialized* (assigned for the first time).

```
3 dollars = 2;  
4 System.out.println(dollars); // OK  
5 System.out.println(cents); // error
```

dollars	<input type="text" value="2"/>
cents	<input type="text"/>

Each time you assign a variable, you are *updating* its value stored in memory.

```
6 dollars = 3;  
7 dollars = 4;  
8 cents = 49;
```

dollars	<input type="text" value="4"/>
cents	<input type="text" value="49"/>

Questions (10 min)

Start time: _____

15. How many times is each variable in Model 4 assigned?

16. What is the error in the second `System.out.println` statement? (Don't just repeat the text in Model 4; explain in your own words what the problem is.)

17. What is the value of `dollars` right before it's assigned for the last time? What is the value of `cents` before it's assigned for the last time?

18. Consider the statement: `cents = dollars;`

a) Compare this code to lines 6–8 in Model 4. What value do you think `cents` and `dollars` will have after running this statement?

b) Which side of the equals sign (left or right) was assigned a new value?

19. In Java, the `+` and `-` symbols are used to perform addition and subtraction. For example, the statement `dollars = dollars + 1;` adds one to the current value of `dollars`.

a) What is the value of `dollars` (in memory) after running this statement?

b) Do you consider the equals sign in Java an operation to be performed? (like `+`)
If so, explain the operation. If not, explain why not.

c) Do you consider the equals sign in mathematics an operation to be performed?
If so, explain the operation. If not, explain why not.

20. In your own words, explain how you should read the `=` sign in Java. For example, the Java statement `x = a + b;` should be read as "x _____ a plus b."