Lab 2c

CSC-121, Fall 2025

TF and SI Exercises

Warm up!

1. Which of the following are invalid variable names and why?

```
x
99bottles
july97
theSalesFigureForFiscalYear98
r&d
grade_report
INT
```

2. Is the variable name Sales the same as sales? Why or why not?

3. A program has a `float` variable named `total` and a `double` variable named `number`. Write a statement that assigns `number` to `total` without causing an error when compiled.

4. An expression adds a `byte` variable and a `short` variable. Of what data type will the result be?

Discussion

1. Is the following comment a single-line style comment or a multi-line style comment?

```
/* This program was written by M. A. Codewriter */
```

2. Is the following comment a single-line style comment or a multi-line style comment?

```
// This program
// was written by
// M. A. Codewriter
```

3. Describe what the phrase “self-documenting program” means.

4. Briefly describe what programming style means. Why should your programming style be consistent?

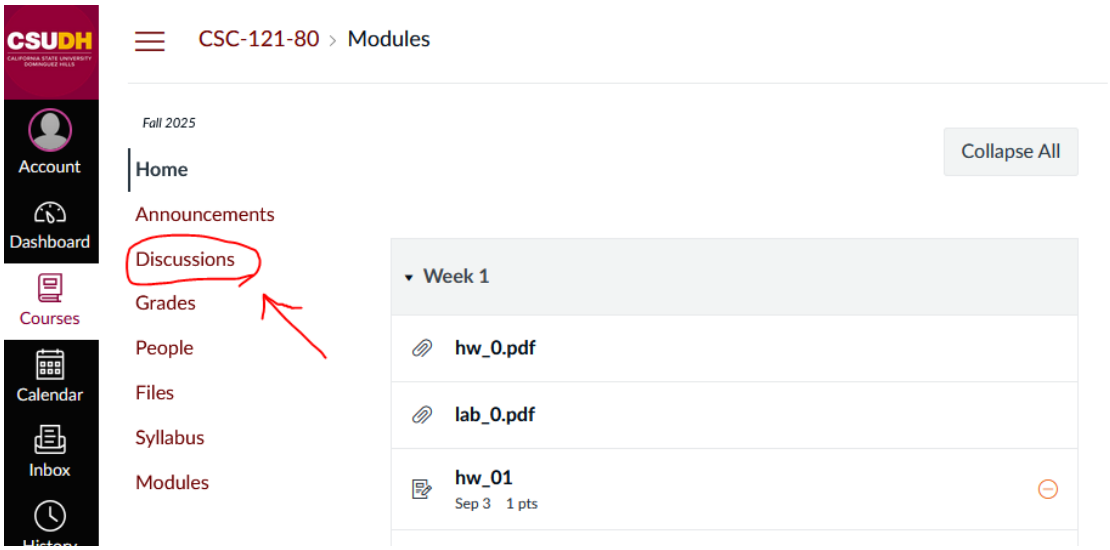
5. Assume that a program uses the named constant PI to represent the value 3.14. The program uses the named constant in several statements. What is the advantage of using the named constant instead of the actual value 3.14 in each statement?

Small Group Exercises

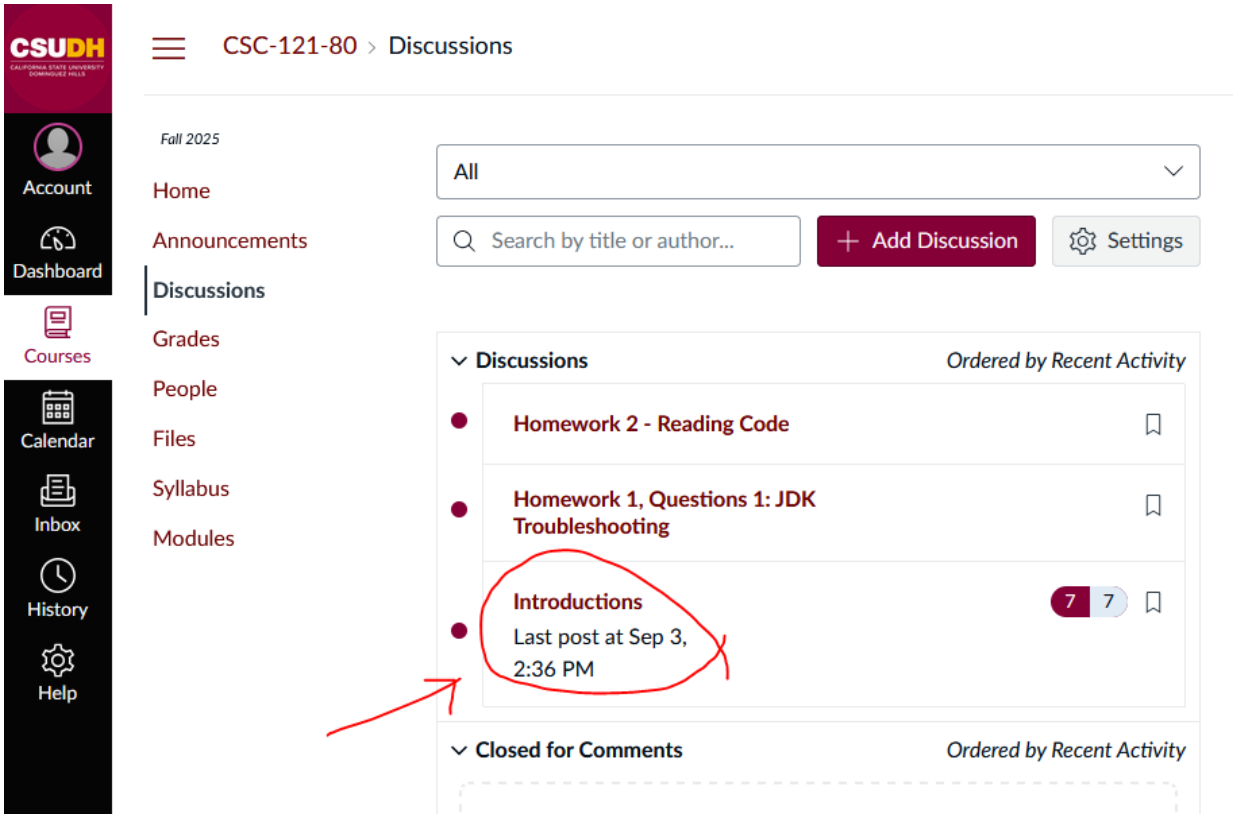
Work together in groups of 3 or 4 and talk through the following problems.

Canvas Discussions

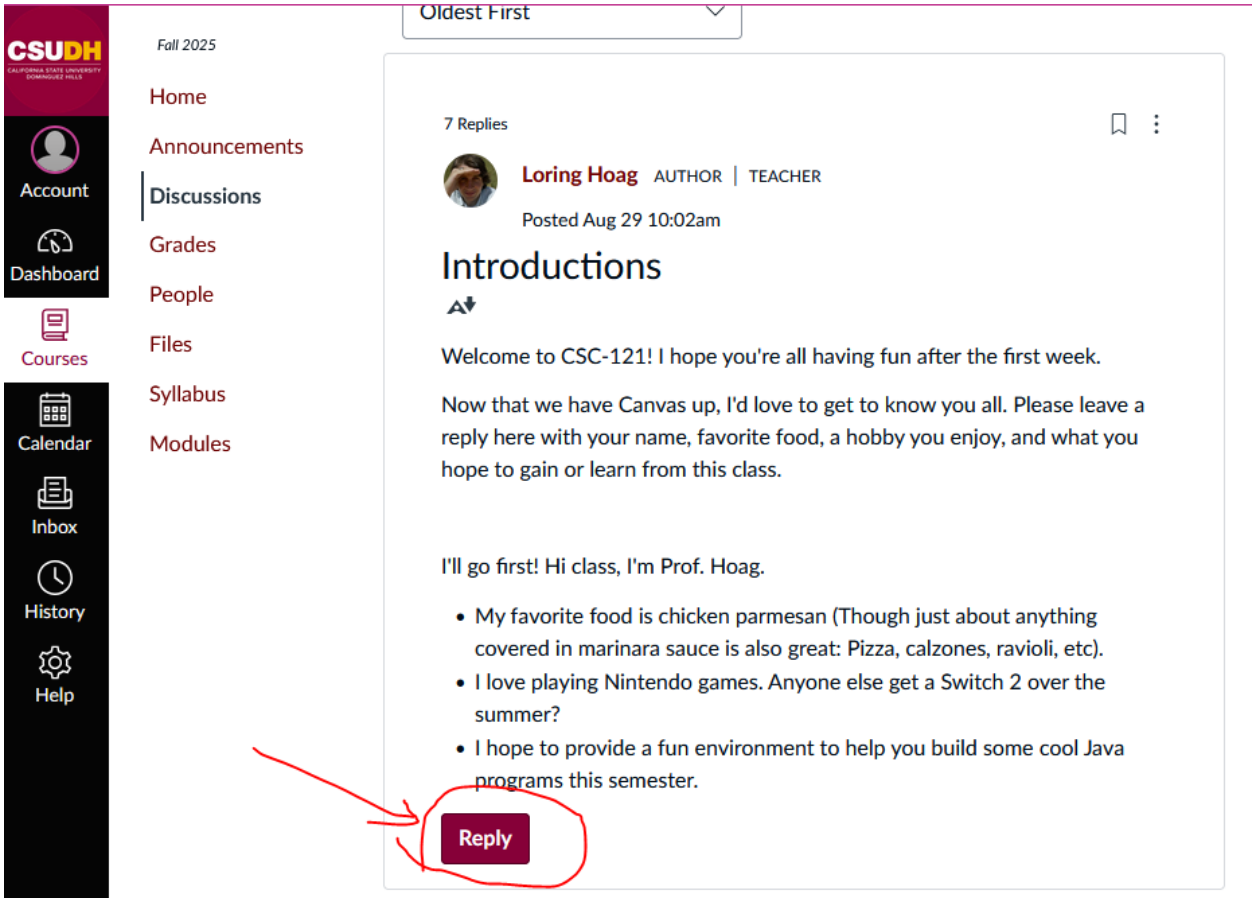
Did you know Canvas has a discussion forum where you can ask for homework and project help? For this exercise, log onto your Canvas, open the CSC-121 course, and click the Discussions button.



Now, look for the Introductions topic. (See next image)



Take a look at the main post and some of the replies, then add a new reply to the topic.(See next image)



If you have not already posted a reply to the topic, click the "Reply" button. Tell us a little about yourself. Great job!

Getting User Input with Scanners and Dialogs

1. Write the code to set up all the necessary objects for reading keyboard input with a Scanner object. Then write code that asks the user to enter his or her name and desired annual income. Store the input in string and double variables, and then print them with the `System.out.println()` method. The slides from lecture on using Scanner objects are included on the next page.

2. Let's write the same program again, but this time using Dialog Boxes. Use an Input Dialog instead of a Scanner, and print the output using a Message Dialog instead of `System.out.println`.

The example slides from lecture on Dialogs are included on the next page. The output of the Input Dialog is *always a String*, so you will need to use a parse method to convert the annual income amount from a String to a Double.

Don't forget that when using JOptionPane UI objects that you need to end your program with the `System.exit()` command. (If you're not sure what to use as the argument to System.exit(), ask a TA.)

The Scanner Class (3 of 3)

- So when you need to ask user for an input:

1. Add `import java.util.Scanner;` at the top of our programs. (only one time)
2. Create a `Scanner` object: (only one time)

```
Scanner keyboard = new Scanner (System.in);
```

3. Use a Scanner class method for reading strings, bytes, integers, long integers, short integers, floats or doubles. (Table 2-17 in the text. (for each input)
   EX: To read an integer type number use `nextInt()` method;

```
int number;
number = keyboard.nextInt();
```

OR for a String input use `nextLine()` method:

```
String name;
name = keyboard.nextLine();
```

Method
nextByte
nextDouble
nextFloat
nextInt
nextLine
nextLong
nextShort

The Parse Methods (2 of 2)

```
// Store 1 in bVar.
byte bVar = Byte.parseByte("1");

// Store 2599 in iVar.
int iVar = Integer.parseInt("2599");

// Store 10 in sVar.
short sVar = Short.parseShort("10");

// Store 15908 in lVar.
long lVar = Long.parseLong("15908");

// Store 12.3 in fVar.
float fVar = Float.parseFloat("12.3");

// Store 7945.6 in dVar.
double dVar = Double.parseDouble("7945.6");
```

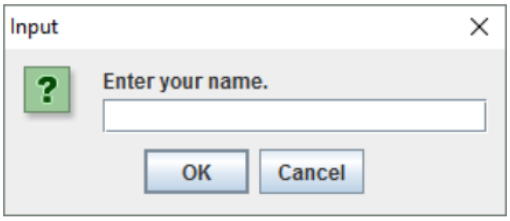
Input Dialogs (2 of 2)

Example:

```
String name;

name = JOptionPane.showInputDialog("Your message or question.");
```

- The argument passed to the method is the message to display.
- If the user clicks on the OK button, `name` references the string entered by the user.
- If the user clicks on the Cancel button, `name` references `null`.



Message Dialogs

- `JOptionPane.showMessageDialog` method is used to display a message dialog.

```
Ex: JOptionPane.showMessageDialog(null, "Hello World");
```

- The first argument will be discussed later.
- The second argument is the message that is to be displayed.