

# **Honors Computer Programming 1-2**

## **Chapter 3 Activities**

## 1A. Correct the errors in the following variable declarations:

a) `double 3.14159;` → `double num = 3.14159;`

b) `int;` → `int num;`

c) `double = 314.00;` → `double num = 314.00;`

d) `int double_value = 314.00;` → `double double_value = 314.00;`

e) `int_value 314;` → `int value = 314;` or

f) `double working_value 314` | `int double_value = (int) 314.00;` → `double workingValue = 314;`

**1B Complete the following table by renaming the bad variable names and providing descriptions.**

Bad Variable Declaration	Improved Declaration	Description
double current profit;	double currentProfit;	the current profit
	int numSold;	counts products sold this month
double %increase	double percentIncrease;	the percent increase
	double dollarsEarned;	dollars earned this month

**1C Give definitions for each of the constant descriptions listed below.**

Constant Defintion	Description
<pre>final int DAYS_IN_WEEK = 7;</pre>	Number of days in a week
<pre>final int WEEKS_IN_YEAR = 52;</pre>	Number of weeks in a year
<pre>final double WAGE_PER_HOUR = 9.00;</pre>	Minimum wage per hour

**2A** Fix the right-hand sides of the following assignments so that they are the correct type for the variable on the left side.

<code>int x = Math.sqrt(4);</code>	<code>int x = (int)Math.sqrt(4);</code>
<code>double y = "3";</code>	<code>double y = Double.parseDouble("3");</code>
<code>String z = 3.14;</code>	<code>String z = "" + 3.14;</code>

## 2B What is wrong with each of these assignments?

Statement	Error
<code>a + 2 = 3;</code>	left-hand side must be a variable
<code>Math.PI = 3;</code>	a constant cannot be modified
<code>x = x + 1;</code>	The assignment operator is a single =


2C Computations with floating-point numbers have finite precision. What values are assigned in the following assignments?

Statement	Value
<code>double x = Math.pow(10, 20) + 1;</code>	1.0E20

$$\begin{aligned} x &= 10^{20} + 1 \\ &= 10000000000000000000001 \end{aligned}$$

**2C Computations with floating-point numbers have finite precision. What values are assigned in the following assignments?**

Statement	Value
<code>double x = Math.pow(10, 20) + 1;</code>	1.0E20
<code>double y = x - 1;</code>	1.0E20


$$\begin{aligned}y &= x - 1 \\&= (10^{20} + 1) - 1 \\&= 10^{20} \\&= 10000000000000000000000000000000\end{aligned}$$



**2C Computations with floating-point numbers have finite precision. What values are assigned in the following assignments?**

Statement	Value
<code>double x = Math.pow(10, 20) + 1;</code>	1.0E20
<code>double y = x - 1;</code>	1.0E20
<code>double z = x - y;</code>	0.0

$$\begin{aligned} z &= x - y \\ &= (10^{20} + 1) - 10^{20} \\ &= 1 \end{aligned}$$

### 3A Recalling what you've learned about integers and floating-point values, what value is assigned to x by each of the following?

Statement	Value
<code>int x = 6 / 3;</code>	2

$$\begin{array}{r} 2 \\ 3 \overline{) 6} \\ \underline{6} \\ 0 \end{array}$$

quotient is 2,  
remainder is discarded

### 3A Recalling what you've learned about integers and floating-point values, what value is assigned to x by each of the following?

Statement	Value
<code>int x = 6 / 3;</code>	2
<code>int x = 7 / 3;</code>	2

$$\begin{array}{r} 2 \\ 3 \overline{) 7} \\ \underline{6} \\ 1 \end{array}$$

quotient is 2,  
remainder is discarded

### 3A Recalling what you've learned about integers and floating-point values, what value is assigned to x by each of the following?

Statement	Value
<code>int x = 6 / 3;</code>	2
<code>int x = 7 / 3;</code>	2
<code>double x = 7 / 3;</code>	2.0

do division first

convert 2 to a double

$$\begin{array}{r} 2 \\ 3 \overline{) 7} \\ \underline{6} \\ 1 \end{array}$$

quotient is 2,  
remainder is discarded

### 3A Recalling what you've learned about integers and floating-point values, what value is assigned to x by each of the following?

Statement	Value
<code>int x = 6 / 3;</code>	2
<code>int x = 7 / 3;</code>	2
<code>double x = 7 / 3;</code>	2.0
<code>int x = 7 % 3;</code>	1

$$\begin{array}{r} 2 \\ 3 \overline{) 7} \\ \underline{6} \\ 1 \end{array}$$

remainder is 1,  
quotient is discarded

### 3A Recalling what you've learned about integers and floating-point values, what value is assigned to x by each of the following?

Statement	Value	Statement	Value
<code>int x = 6 / 3;</code>	2	<code>int x = 6 % 3;</code>	0
<code>int x = 7 / 3;</code>	2		
<code>double x = 7 / 3;</code>	2.0		
<code>int x = 7 % 3;</code>	1		

$$\begin{array}{r} 2 \\ 3 \overline{) 6} \\ \underline{6} \\ 0 \end{array}$$

remainder is 0,  
quotient is discarded

### 3A Recalling what you've learned about integers and floating-point values, what value is assigned to x by each of the following?

Statement	Value	Statement	Value
<code>int x = 6 / 3;</code>	2	<code>int x = 6 % 3;</code>	0
<code>int x = 7 / 3;</code>	2	<code>int x = 999 / 1000;</code>	0
<code>double x = 7 / 3;</code>	2.0		
<code>int x = 7 % 3;</code>	1		

$$\begin{array}{r} 0 \\ 1000 \overline{) 999} \\ \underline{\phantom{0}0} \\ 999 \end{array}$$

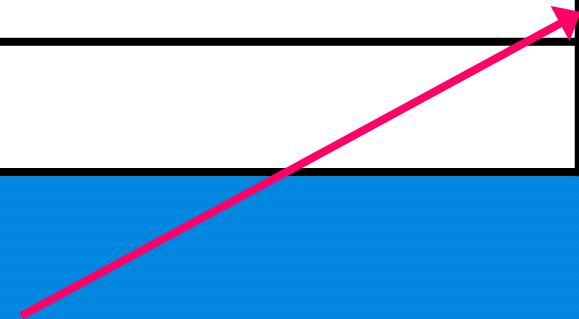
quotient is 0,  
remainder is discarded

### 3A Recalling what you've learned about integers and floating-point values, what value is assigned to x by each of the following?

Statement	Value	Statement	Value
<code>int x = 6 / 3;</code>	2	<code>int x = 6 % 3;</code>	0
<code>int x = 7 / 3;</code>	2	<code>int x = 999 / 1000;</code>	0
<code>double x = 7 / 3;</code>	2.0	<code>double x = 999.0 / 1000.0;</code>	.999
<code>int x = 7 % 3;</code>	1		

$$\begin{array}{r} 0.999 \\ 1000.0 \overline{) 999.000} \end{array}$$

ordinary decimal division





### 3A Recalling what you've learned about integers and floating-point values, what value is assigned to x by each of the following?

Statement	Value	Statement	Value
<code>int x = 6 / 3;</code>	2	<code>int x = 6 % 3;</code>	0
<code>int x = 7 / 3;</code>	2	<code>int x = 999 / 1000;</code>	0
<code>double x = 7 / 3;</code>	2.0	<code>double x = 999.0 / 1000.0;</code>	.999
<code>int x = 7 % 3;</code>	1	<code>int x = (int) (999 / 1000.0);</code>	0

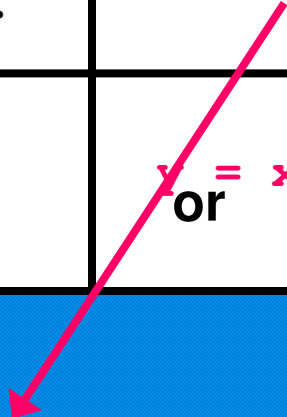
$$\begin{array}{r} 0.999 \\ 1000.0 \overline{) 999.000} \end{array}$$

ordinary decimal division

convert to an integer:  
drop the decimal part

### 3B Translate the following algebraic expressions into Java:

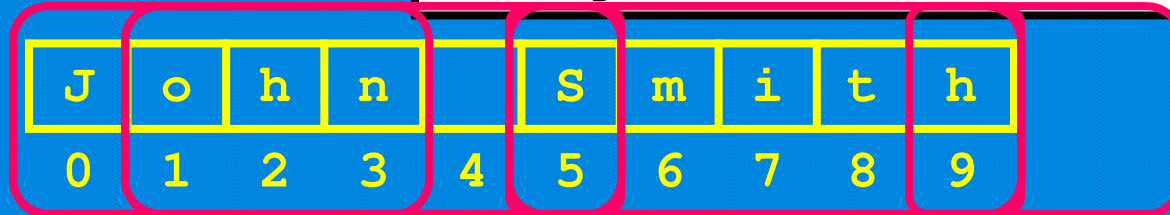
Algebraic Expression	Java Equivalent
$y = x + \frac{1}{2}$	<code>y = x + 1.0 / 2.0;</code>
$y = x^2 + 2x + 1$	<code>y = Math.pow(x, 2) + 2 * x + 1;</code>
$y = \frac{x}{1 - x}$	<code>y = x / (1 - x);</code> or



```
y = x * x + 2 * x + 1;
```

#### 4. What will be the resulting substring in the following examples?

Assume the declaration: `String name = "John Smith"`



Substring Expression	Result
<code>String firstName = name.substring(0, 4);</code>	John
<code>String lastName = name.substring(5, 6);</code>	S
<code>String name2 = name.substring(1, 11);</code>	out-of-bounds error
<code>String lastName = name.substring(5);</code>	Smith
<code>String lname = name.substring(0, 6);</code>	John S
<code>String lname = name.substring(9, 10);</code>	h
<code>String lname = name.substring(1, name.length());</code>	ohn Smith

5. Using `JOptionPane.showInputDialog`, ask the user to supply two integers `a` and `b`. Then print out the values `a / b` and `a % b`.

```
import javax.swing.JOptionPane; // necessary for input dialog
...
// input the first number
String input = JOptionPane.showInputDialog("Enter an integer:");
int a = Integer.parseInt(input);

// input the second number
input = JOptionPane.showInputDialog("Enter a second integer:");
int b = Integer.parseInt(input);

// output the quotient and remainder
System.out.println("Quotient = " + (a / b));
System.out.println("Remainder = " + (a % b));

// allow the program to exit automatically
System.exit(0);
```

6. Assume:

```
int x = 2;  
int y = 3;  
int z = 4;  
String line = "The answer is: ";
```

State the output for each of the following.

a) `System.out.println(line + x + y + z);`

The answer is: 234

b) `System.out.println(line + (x + y + z));`

The answer is: 9

c) `System.out.println(line + x + " " + z);`

The answer is: 2 4

d) `System.out.println(x + ") " + line + y + z);`

2) The answer is: 34

e) `System.out.println(line + ((x + y + z) / 3));`

The answer is: 3