

Warm-Up for Introduction to OOP (Object-Oriented Programming)

1. Compare and contrast Line #1 and Line #2 below:

```
Line #1:      int num_students;  
Line #2:      int numStudents = 33;
```

Line #1 declares a variable but doesn't initialize it. Line 2 uses different style for naming the variable identifies

2. The code below exists within the same program. Compare and contrast the method defined on Line #4³ with the method defined on Line #5.

```
Line #2:      String name = 'Kelly';  
Line #3:      public void printName()  
Line #4:      {      System.out.println(name); }
```

```
Line #5:      public void printName(String name)  
Line #6:      {      System.out.println(name); }
```

The method in line #3 takes no parameter variable, but the method in line #5 takes one String parameter. Both are void!

3. Write the lines of code needed to "call" the two methods defined above. How does the program know which method to "call" when both methods have the same name?

How do we use Java class constructors to instantiate objects?

1. Compare and contrast the following lines of code:

Line 1: String cartoon = "Care Bears";

Line 2: int numBears = 10;

Notes: Both lines are statements that declare variables. Line 1 initializes the string variable 'cartoon' to "Care Bears", and line 2 initializes the int data for 'numBears' to 10. *initialize*

2. Compare and contrast Lines 1 and 2 with Line 3 in the following lines of code:
 3 4 5

Line 3: String cartoon2;

Line 4: cartoon2 = "Inspector Gadget";

Line 5: String cartoon3 = "Teenage Ninja Turtles";

Notes: Line 3 declares 'cartoon2' and line 4 initializes 'cartoon2', but line 5 declares and initializes 'cartoon3' in ONE statement.

3. Compare and contrast the lines of code below:

Line 6: String cartoon4 = new String("Heathcliff");

Line 7: Scanner in = new Scanner(System.in);

Line 8: Person p1 = new Person()

Notes: All three lines construct objects. Lines 6 and 7 construct objects w/ constructors that take one parameter, but line 8 constructs a 'no args' object for the Person class.