Joshua McCann  
OOP Principles – LAB 4

1. The difference between an instance and static is static there is only one of, while instance there is one for every object created. A static variable inside of a class will hold only one value of the variable for every object to share. An instance variable is created for ever object. So far, every variable we’ve created in all classes have been instance variables. All methods in main, and the method IsNullOrEmpty() created in the Person class, have all been static methods. Static methods cannot touch non-static variables, so the class IsNullOrEmpty() accepted a string that was passed from a non-static method.
2. A value is a value that can be changed independently from other values while a reference variable points to another variable and changes the same shared values between the variables.  
   If an object is not instantiated and is set to another variable of the same time, then we’ve created a reference variable between the two variables.

Output between two objects vs an object with a reference variable:

|  |  |
| --- | --- |
| // create one new object Object objOne = new Object; Object objTwo;  //set objTwo to reference objOne. objTwo = objOne;  //Set a variable on objOne and objTwo objTwo.setName(“Object Two”); objOne.setName(“Not Object Two”);  //print out objTwo variable  System.out.println(objTwo.getName); | // create two new objects Object objOne = new Object; Object objTwo = newObject;  //clone objTwo from objOne objTwo = objOne;  //Set a variable on objOne and objTwo objTwo.setName(“Object Two”); objOne.setName(“Not Object Two”);  //print out objTwo variable  System.out.println(objTwo.getName); |
| System prints: Not Object Two | System prints:  Object Two |

1. Example code, setting up a HashTable, HashMap, and a HashSet.

|  |
| --- |
| //Create HashTable with silly laws  HashTable<Integer, String> sillyLawsTable = new HashTable();  sillyLawsTable.put(1, “Bingo games cannot last more than 5 hours in NC”);  sillyLawsTable.put(2, “It’s illegal to sing off-key in NC”);  sillyLawsTable.put(3, “It’s illegal to sell your eyeballs in Texas”);  sillyLawsTable.put(4, “In Minneapolis, MN, red cars not drive down Lake St.”);  sillyLawsTable.put(5, “Bear wrestling matches are prohibited in Alabama”);  sillyLawsTable.put(6, “You cannot buy meat of any kind on Sunday in Washington”);  //Create HashMap with silly laws  HashMap<Integer, String> sillyLawsMap = new HashMap();  sillyLawsMap.put(1, “Bingo games cannot last more than 5 hours in NC”);  sillyLawsMap.put(null, “It’s illegal to sing off-key in NC”);  sillyLawsMap.put(3, “It’s illegal to sell your eyeballs in Texas”);  sillyLawsMap.put(4, “In Minneapolis, MN, red cars not drive down Lake St.”);  sillyLawsMap.put(5, “Bear wrestling matches are prohibited in Alabama”);  sillyLawsMap.put(2, “You cannot buy meat of any kind on Sunday in Washington”);  //Create HashSet with silly laws  HashSet<String> sillyLawsSet = new HashSet();  sillyLawsSet.add(“Bingo games cannot last more than 5 hours in NC”);  sillyLawsSet.add(“It’s illegal to sing off-key in NC”);  sillyLawsSet.add(“It’s illegal to sell your eyeballs in Texas”);  sillyLawsSet.add(“In Minneapolis, MN, red cars not drive down Lake St.”);  sillyLawsSet.add(“Bear wrestling matches are prohibited in Alabama”);  sillyLawsSet.add(“You cannot buy meat of any kind on Sunday in Washington”); |