# 16.02 Virtual Lecture Notes

Every replacement algorithm usually involves a traversal of some kind. Let us consider our **`inventory`** array from **TestInventory.java**.

Now let us assume we have the same five items as in the previous lesson:

```
      InventoryItem[] inventory = new InventoryItem[5];

      // create inventory
      inventory[0] = new InventoryItem("Towel", 200);
      inventory[1] = new InventoryItem("Cleaning Cart", 30);
      inventory[2] = new InventoryItem("Toiletry Sets", 100);
      inventory[3] = new InventoryItem("Coffee Set", 300);
      inventory[4] = new InventoryItem("Pillows", 50);
```

Waylan would like a way to change out an inventory item for a new one. For example, say, instead of Coffee Set, we now have Beverage Gift Sets. How do we replace that item?

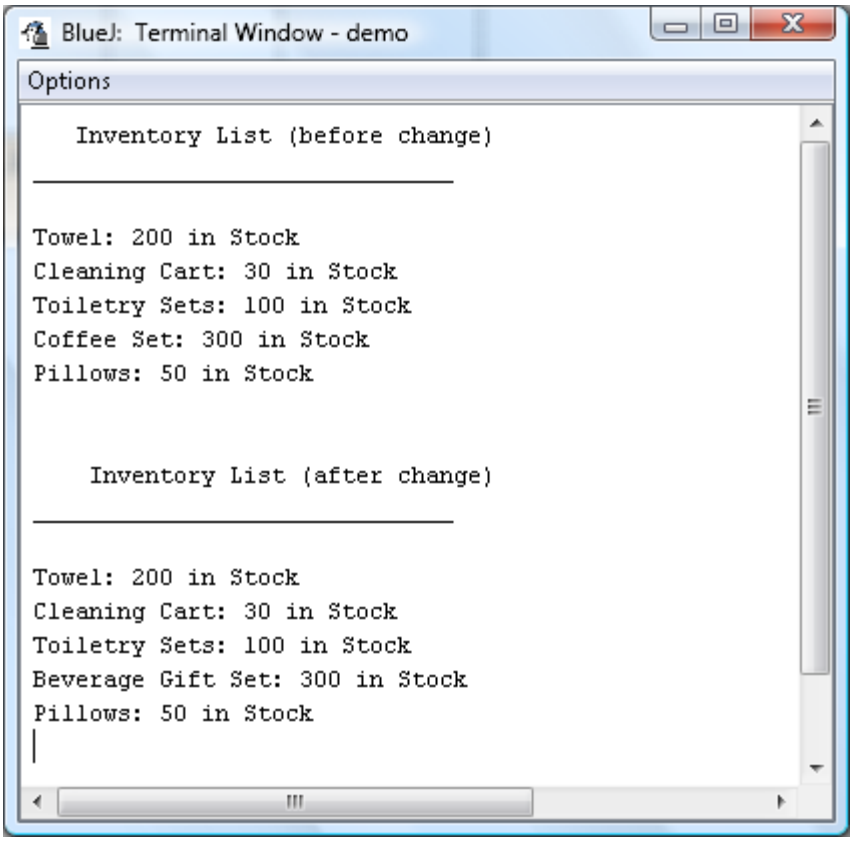
The answer is that first we traverse through the list to find the item. This would require a loop that traverses through the array and then uses an if statement to see if the name matches Coffee Set. Then we set the name to Beverage Gift Sets.

```
   for(int index = 0; index < inventory.length; index++)
       if (inventory[index].getName().equals("Coffee Set"))
               inventory[index].setName("Beverage Gift Sets");
```

We can make a method out of it, as well. Take a look at **`changeItem()`**:

```
    public static void changeItem(InventoryItem[] list,
                                  String find, String replace)
    {
       for(int index = 0; index < list.length; index++)
          if (list[index].getName().equals(find))
               list[index].setName(replace);
    }
```

If we test it by first printing an inventory list before changing, and then after changing, we get the following output:




- Examine the code for this program by conducting a thorough desk check.
- Run the file and make sure you understand it before you continue.

Now, how about an ArrayList? Take a look at the demo program **TestInventory4.java**. Notice that it is the same except for using ArrayList operations.

- Compare the two version of the program side-by-side so you understand how the task is handled differently by arrays and ArrayLists.
- Run the file and make sure you understand it before you continue.