



INTRODUCTION TO JAVA



BY

SANJITHA.S

1. Given two arrays: array1={5,10,15,20,25,30}, array2={50,60,70,80,90,100}. Write a Java code to merge two arrays and display the result.

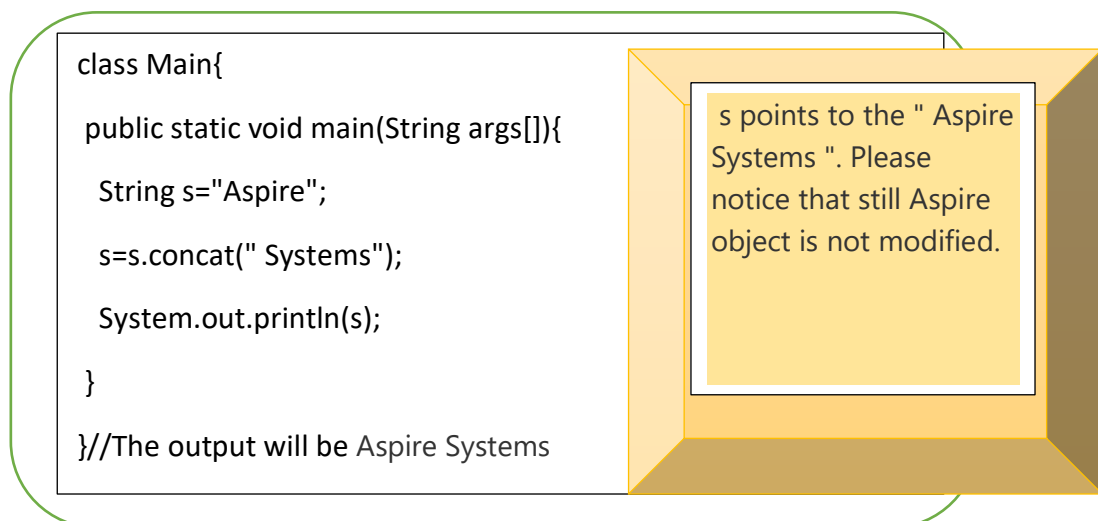
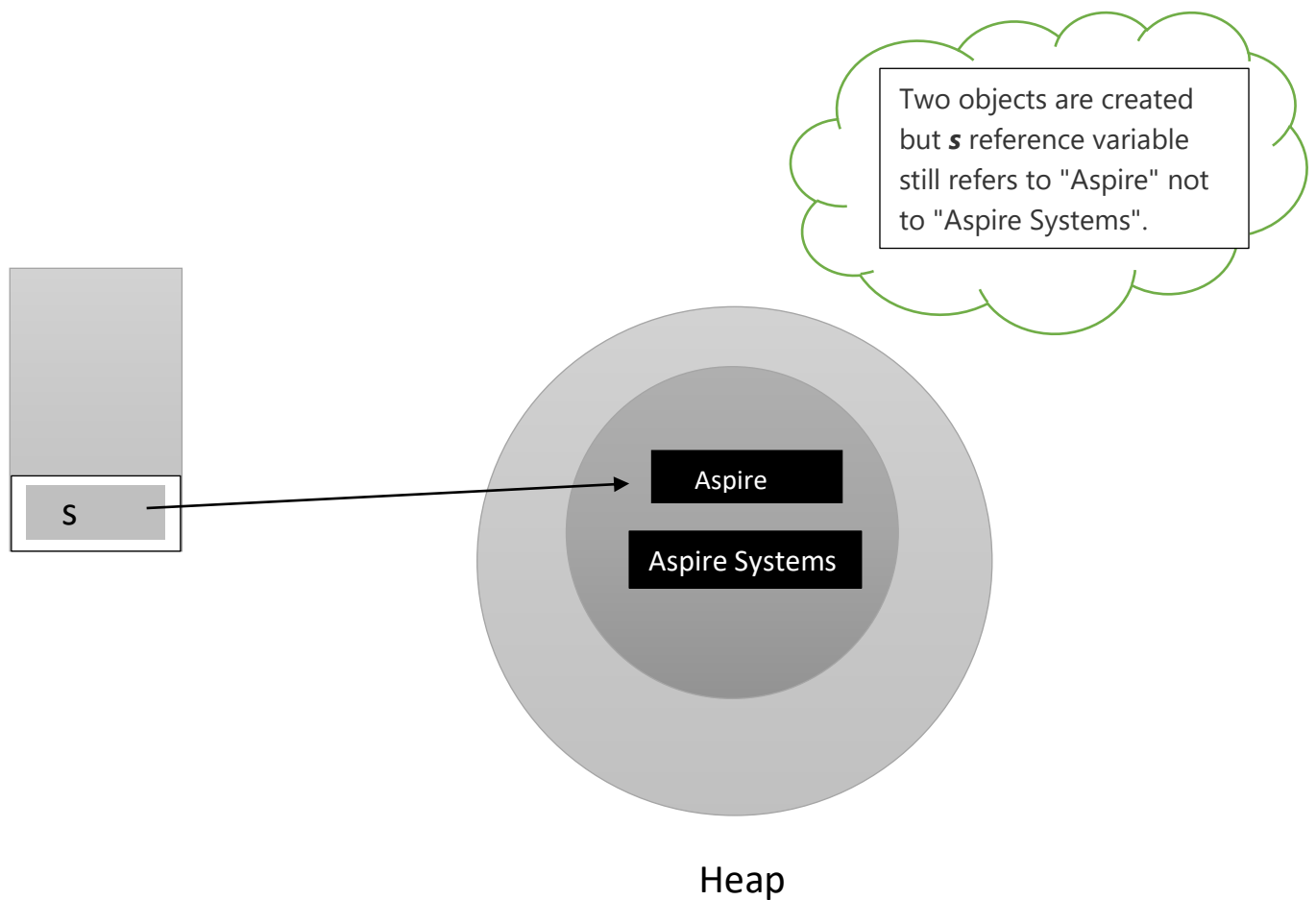
```
import java.util.Arrays;

public class Main
{
    public static void main(String[] args)
    {
        int[] array1 = {5,10,15,20,25,30};
        int[] array2 = {50,60,70,80,90,100};
        int la1 = array1.length;
        int la2 = array2.length;
        int[] result = new int[la1 + la2];
        System.arraycopy(array1, 0, result, 0, la1);
        System.arraycopy(array2, 0, result, la1, la2);
        System.out.println(Arrays.toString(result));
    }
}
```

2. Why String objects are immutable?

The String objects are cached in the String pool, and it makes the String immutable.

```
class Main{
    public static void main(String args[]){
        String s="Aspire";
        s.concat(" Systems");//concat() method appends the
        string at the end
        System.out.println(s);//will print Aspire because strings are
        immutable objects
    }
}
```



3. How to create an immutable class?

There are many immutable classes like String, Boolean, Byte, Short, Integer, Long, Float, Double etc. In short, all the wrapper classes and String class is immutable. We can also create immutable class by creating final class that have final data members.

Requirements to create an immutable class are:

Class->final:

So that child classes can't be created.

Data members->private:

So that direct access is not allowed.

Data members->final:

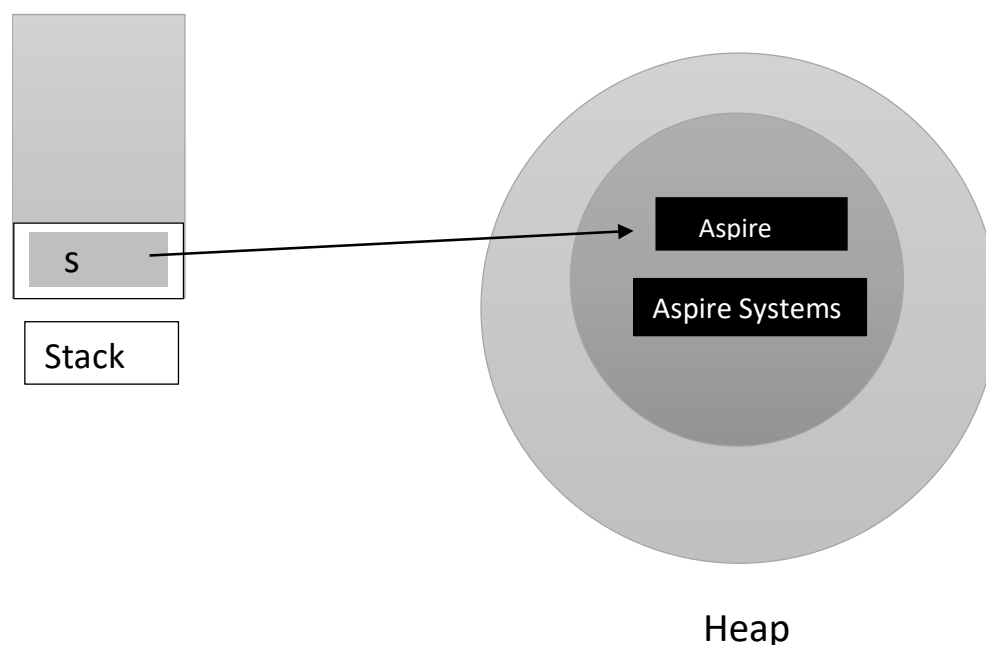
So that we can't change the value of it after object creation.

No setter method:

So that there will be no option to change the value of the instance variable.

4. What is string constant pool?

A string constant pool is a separate place in the heap memory where the values of all the strings which are defined in the program are stored. When we declare a string, an object of type String is created in the stack, while an instance with the value of the string is created in the heap.



5. What code is written by the compiler if you concatenate any string by + (string concatenation operator)?

```

public class Main {
    public static void main(String[] args) {
        String s="Aspire"+" Systems";
        System.out.println(s);
    }
}

```

6. What is the difference between StringBuffer and StringBuilder class?

StringBuffer	StringBuilder
Thread - safe	Not Thread-safe
Synchronized	Not Synchronized
Since Java 1.0	Since Java 1.5
Slower	Faster

7. How to prove String is immutable programatically?

In java, string objects are immutable. Immutable basically means unmodifiable or unchangeable. Once string object is created its data or state can't be changed but a new string object is created.

```

public class Main {
    public static void main(String[] args) {
        String str1 = "Aspire";
        String str2 = "Aspire";
        System.out.println("Before Modification");
        if (str1 == str2) {
            System.out.println("Both pointing to the same reference");
        } else {
            System.out.println("Both are pointing to different reference");
        }
    }
}

```

```

    }
    str1 = str1 + "Systems";
    System.out.println("After Modification");
    if (str1 == str2) {
        System.out.println("Both pointing to the same reference");

    } else {
        System.out.println("Both are pointing to different reference");
    }
}
}

```

8. How to swap two Strings without using a third variable?

Sample Input	Output
a=Aspire	a=Welcome
b=Welcome	b=Aspire

```

public class Main
{
    public static void main(String[] args) {
        String a = "Aspire";
        String b = "Welcome";
        System.out.println("Before swap: ");
        System.out.println("a="+a);
        System.out.println("b="+b);
        a = a + b;
        b = a.substring(0, a.length() - b.length());
    }
}

```

```
    a = a.substring(b.length());  
    System.out.println("After swap: ");  
    System.out.println("a="+a);  
    System.out.println("b="+b);  
    }  
}
```