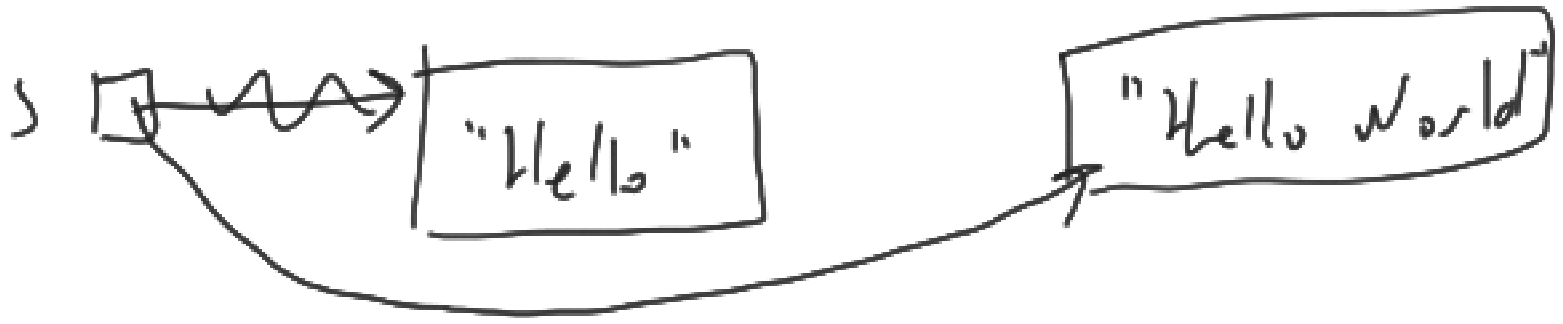
References

- like pointers

- An object that is uninitialized is null by default.

- There is no arrow operator ->

```
String s = "Hello";
```



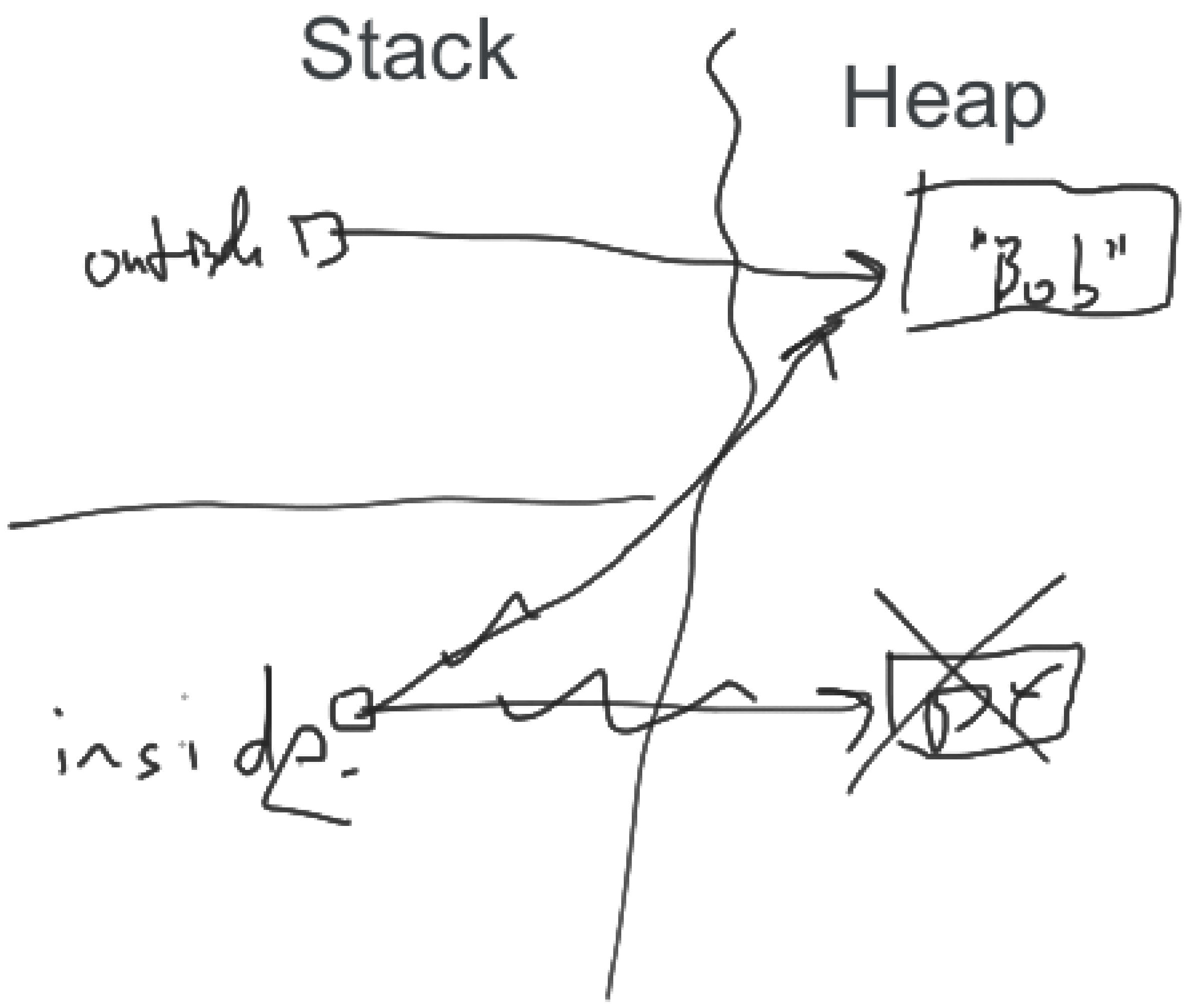
```
s = s + " World";
```

More on passing Strings

```
~~void~~ String change (String inside) {

    inside = "Diff";
    return inside;
}
```

```
String outside = "Bob";
outside = change (outside);
System.out.println(outside);   // prints "Bob"
```



Memory Management

- All objects/instances must be created explicitly with "new"
- No values are on the stack
- All objects are on the heap

- Automatic Garbage Collection
  - no "delete" keyword
  - when an object has nothing pointing to it, it is garbage collected automatically

Java Classes

- Single file structure
  - Each class is in a separate file
  - Each class is in a single file with the implementation and the declaration together

- Other Differences
  - "protected" has a different meaning
  - no "virtual" keyword because all methods are virtual
  - If you don't want a method overridden, you mark it as "final"
  - calling methods on a superclass uses the "super" keyword

Java Style

- camelCase is preferred
- functions and variables are lower-case