



# Object Oriented Programming with Java (OOPJ)

Session 10: String  
Kiran Waghmare

final	finally	finalize
Used to apply restrictions on class, method, and variable. Final class cannot be inherited, final method cannot be overridden, and final variable value cannot be changed.	Used to place important code, it will be executed whether exception is handled or not.	Used to perform clean up processing just before object is garbage collected.
This is a keyword	This is a block	This is a method

Object

getClass

equals

notifyAll

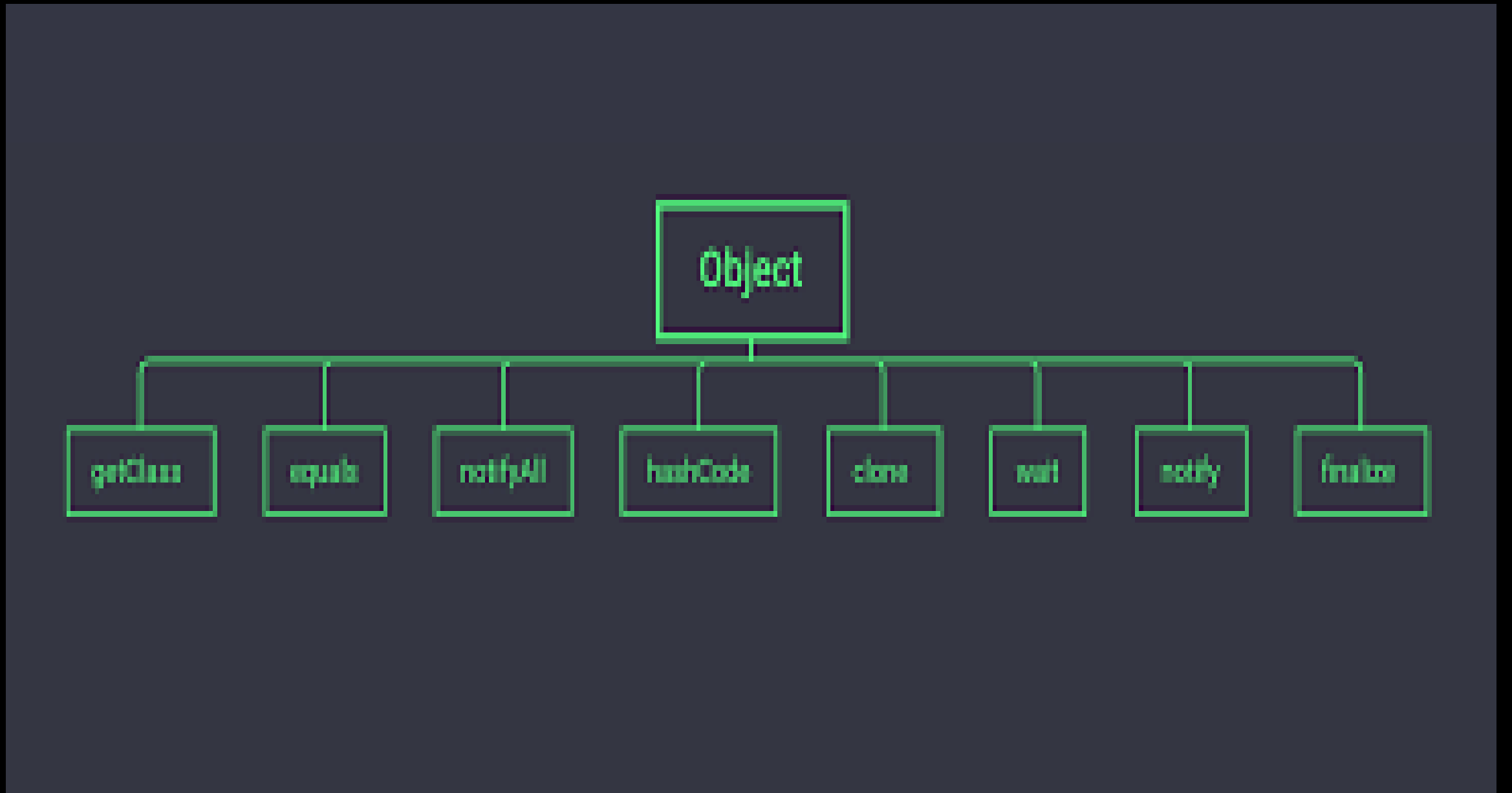
hashCode

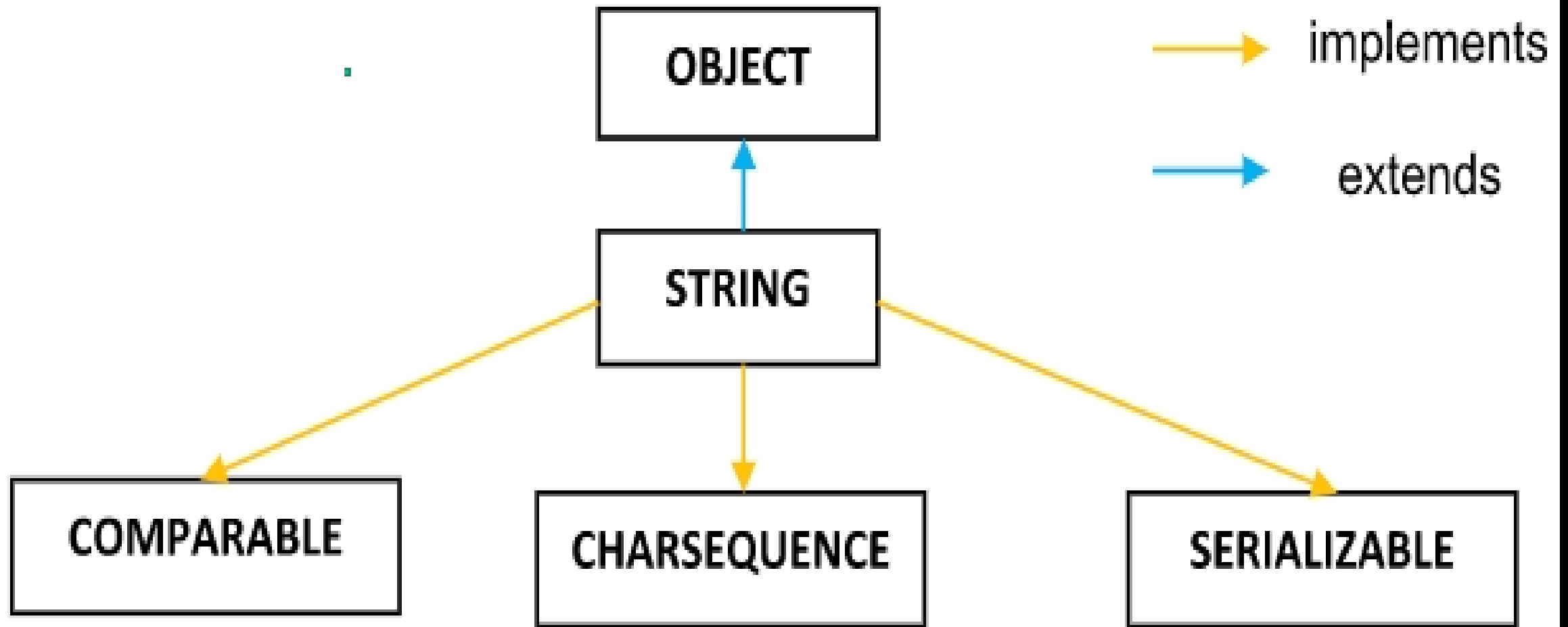
clone

wait

notify

finalize

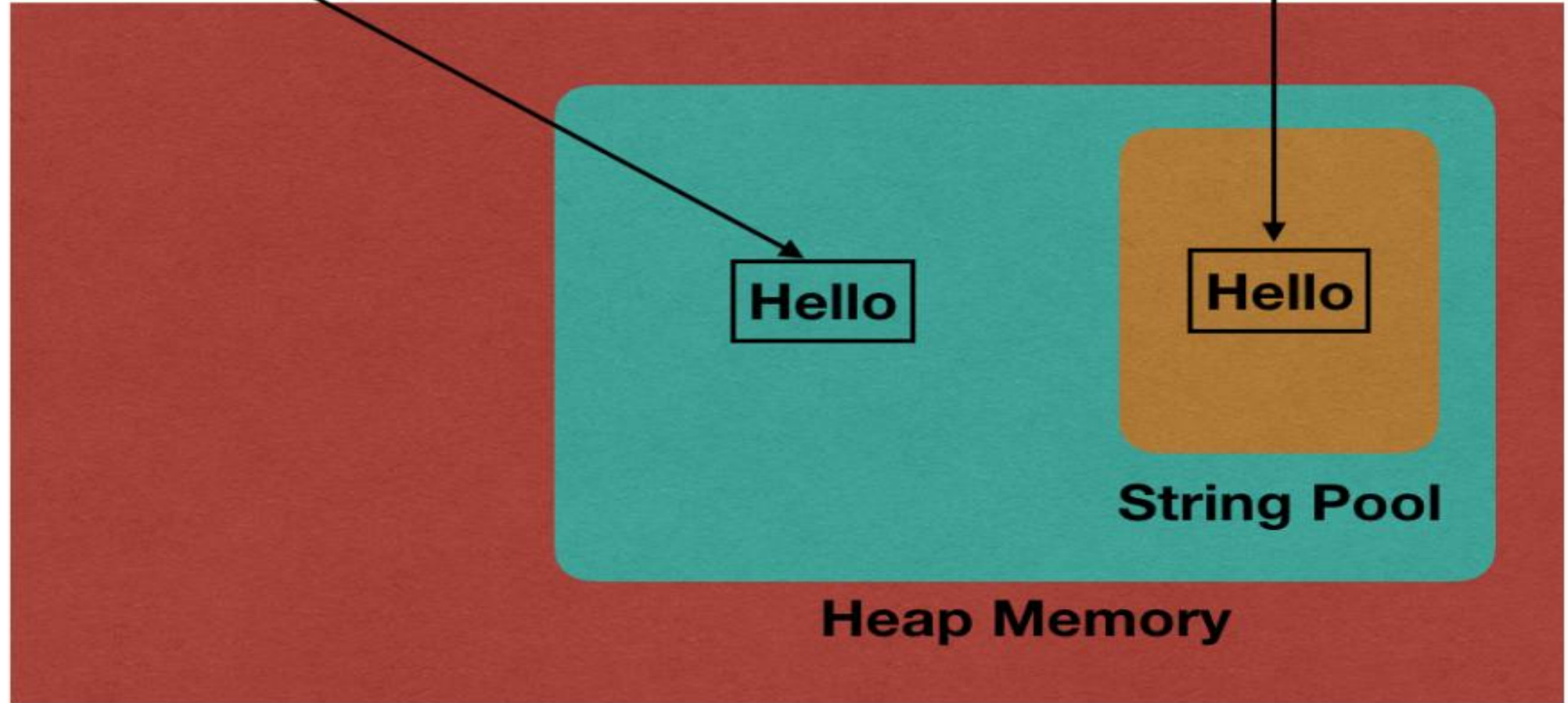




**STRING CLASS IN JAVA**

**String s2 = new String("Hello");**

**String s1 = "Hello";**



**JVM Memory**



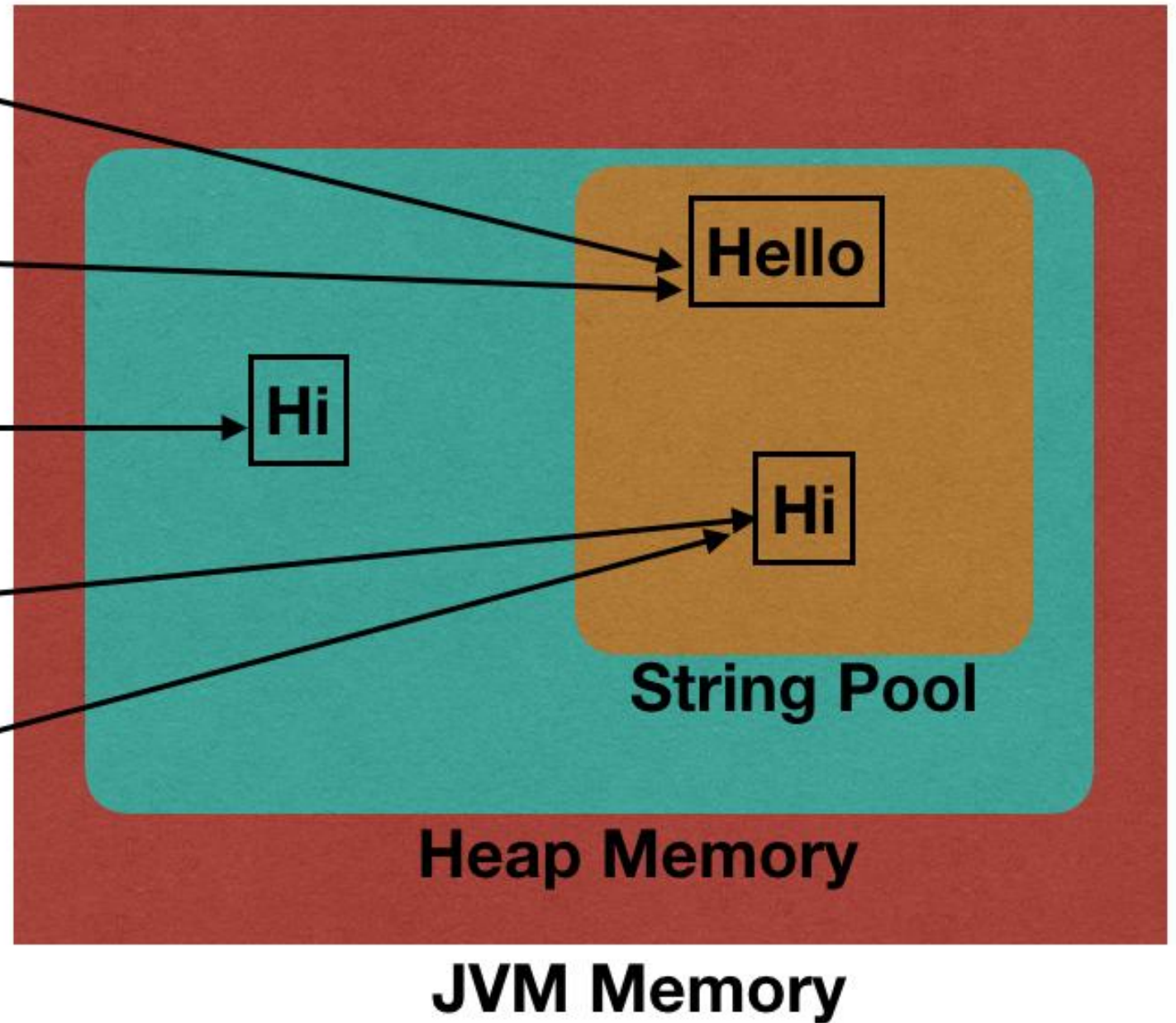
**String s1 = "Hello";**

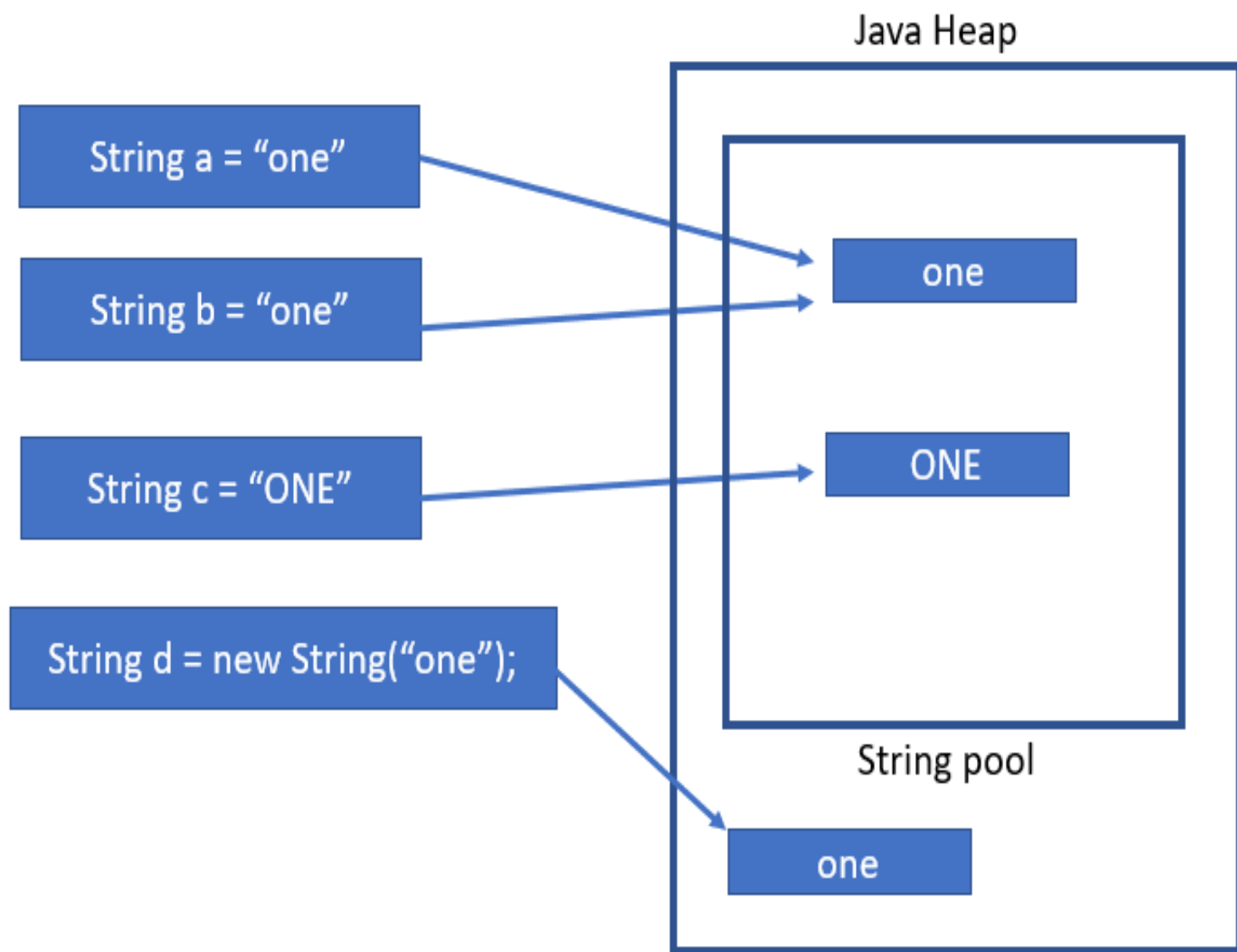
**String s2 = "Hello";**

**String s3 = new String("Hi");**

**String s4 = "Hi";**

**s3 = s3.intern();**





`a == b; // true`

`a == c; // false`

`a.equals(b); // true`

`a.equalsIgnoreCase(c); // true`

`a == d; // false`

`a.equals(d); // true`

# Java String Methods

- |                     |                     |                     |
|---------------------|---------------------|---------------------|
| ( 1 ) length()      | ( 8 ) toUpperCase() | ( 15 ) compareTo    |
| ( 2 ) charAt()      | ( 9 ) split()       | ( 16 ) startsWith() |
| ( 3 ) trim()        | ( 10 ) substring()  | ( 17 ) endsWith()   |
| ( 4 ) indexOf()     | ( 11 ) equals()     | ▪                   |
| ( 5 ) lastIndexOf() | ( 12 ) getBytes()   | ▪                   |
| ( 6 ) replace()     | ( 13 ) concat()     | ▪                   |
| ( 7 ) toLowerCase() | ( 14 ) contains()   | etc.                |



String	StringBuffer	StringBuilder
Stored in String Pool (if interned) or Heap	Stored in Heap	Stored in Heap
Immutable	Mutable	Mutable
Synchronized	Synchronized	Non-Synchronized
Thread Safe	Thread Safe	Not Thread Safe
Less Efficient (due to immutability)	Efficient (due to synchronization)	More Efficient (no synchronization overhead)

```
String s = new String("Hello");
```



Fig 1: String object

```
StringBuffer sb = new StringBuffer("Hello");
```

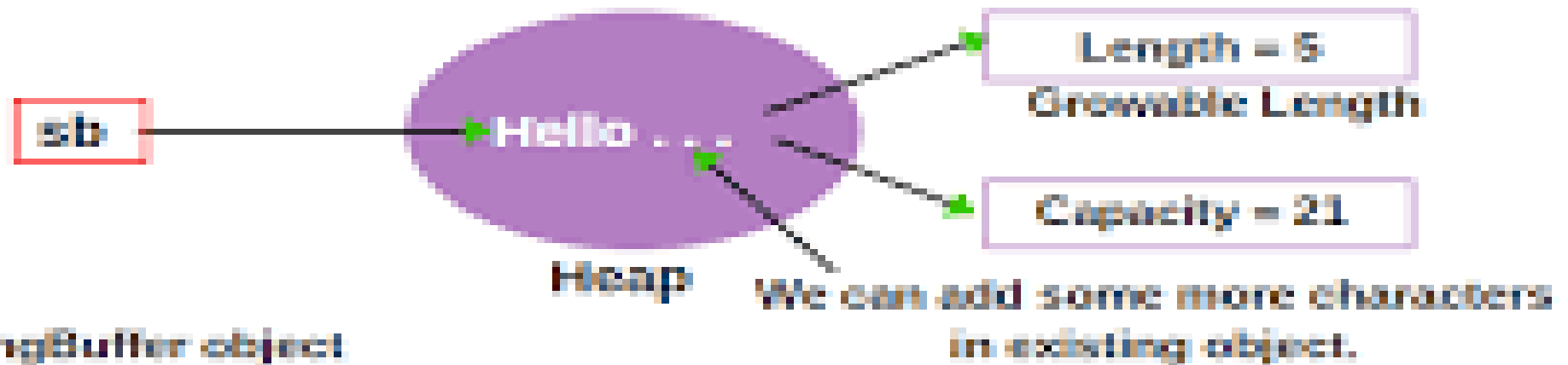
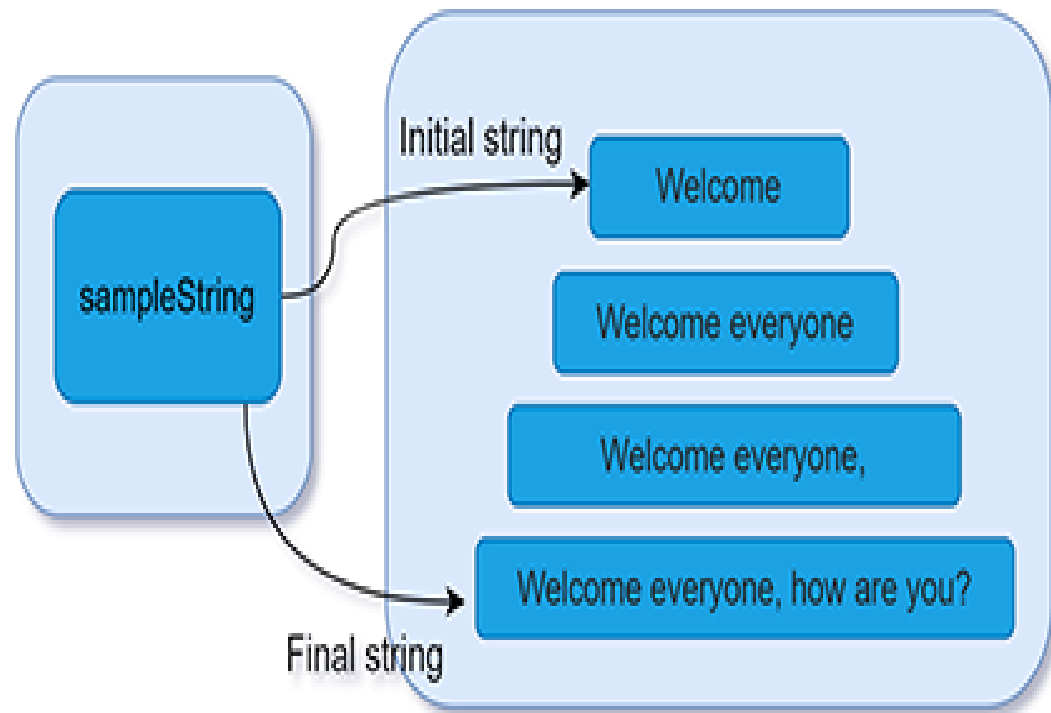


Fig 2: StringBuffer object

## String



## StringBuilder

