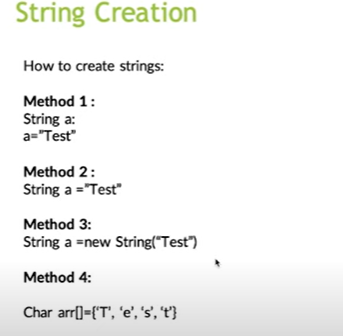
String, String buffer, String builder

String:

1. String is immutable means content can’t be modify.
2. String is a class having several methods.
3. Once we create object by using string latter we can’t do any changes in that object.
4. String always stores in String constant pool.

String object creation



Understand object storing based on string creation?

1. If we are creating string like below

String s =”lb”;

One object created in string constant pool.

But before creating SCP will check whether this content is present or not.

If content present it will reuse the same object, if not present one object will be created.

1. If we are creating String like below

String s = new String(“Lb”);

In this case two objects will be created.

First one is in heap memory, because when every we are using key word “new” one object will be created in heap memory.

Second object will be created in SCP without reference variable for reusable and for future, but jvm will internally maintain one variable.

Before 1.7v SCP is present in method area.

After 1.7v SCP is moved to heap area.

Why java provided SCP to string why not for string buffer?

because in java projects every where we use string that’s why scp concept was given to string, not for string buffer.

Why string is immutable and why string buffer is mutable?

Because if we create object in string SCP, for that object one variable will get assign. If we want to store same data in SCP with that reference variable assign to old object.

If we change one object data then that object assigned variable also change so that’s why string is immutable.

Is there any immutable concepts?

Wrapper class objects are also immutable.

String methods

1. Concat

Combine two different strings.

1. Length

To find string length.

There is no concept of index while counting the string. It always starts from 1.

1. charAt()

to find specific character in string. Here is index concept so counting always starts from 0.

1. Equals()

It will check the content, even case sensitive also matters here.

1. equalsIgnorecase()

It will check the content, but case sensitive is not matter.

1. Replace(target, replacement)

Replace one character to another character.

1. Substring(begin index number, end index number)

Based on given index number, from there string will be print.

1. Split(regex)

We can split based on string spaces, etc..

Understand about “==” and “equals” in both string and SB

In String class:

1. Object is super class having equals method.

If we call object class equals method always compare address.

If we call String class equals method it overrides from object class and logic is written to compare content.

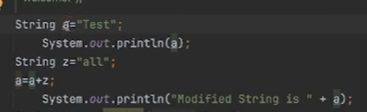
1. In string “==” always compares address only.

In String buffer

1. Object is a super class. And there is no concept of overriding from object class, so equals always compare address only.
2. In string buffer there is no concept of comparing content.
3. “==” also compare address only.

String s= “lb”; // creates in string constant pool.

String b = new String(“lb”) // create on heap memory.



Understand string is immutable?

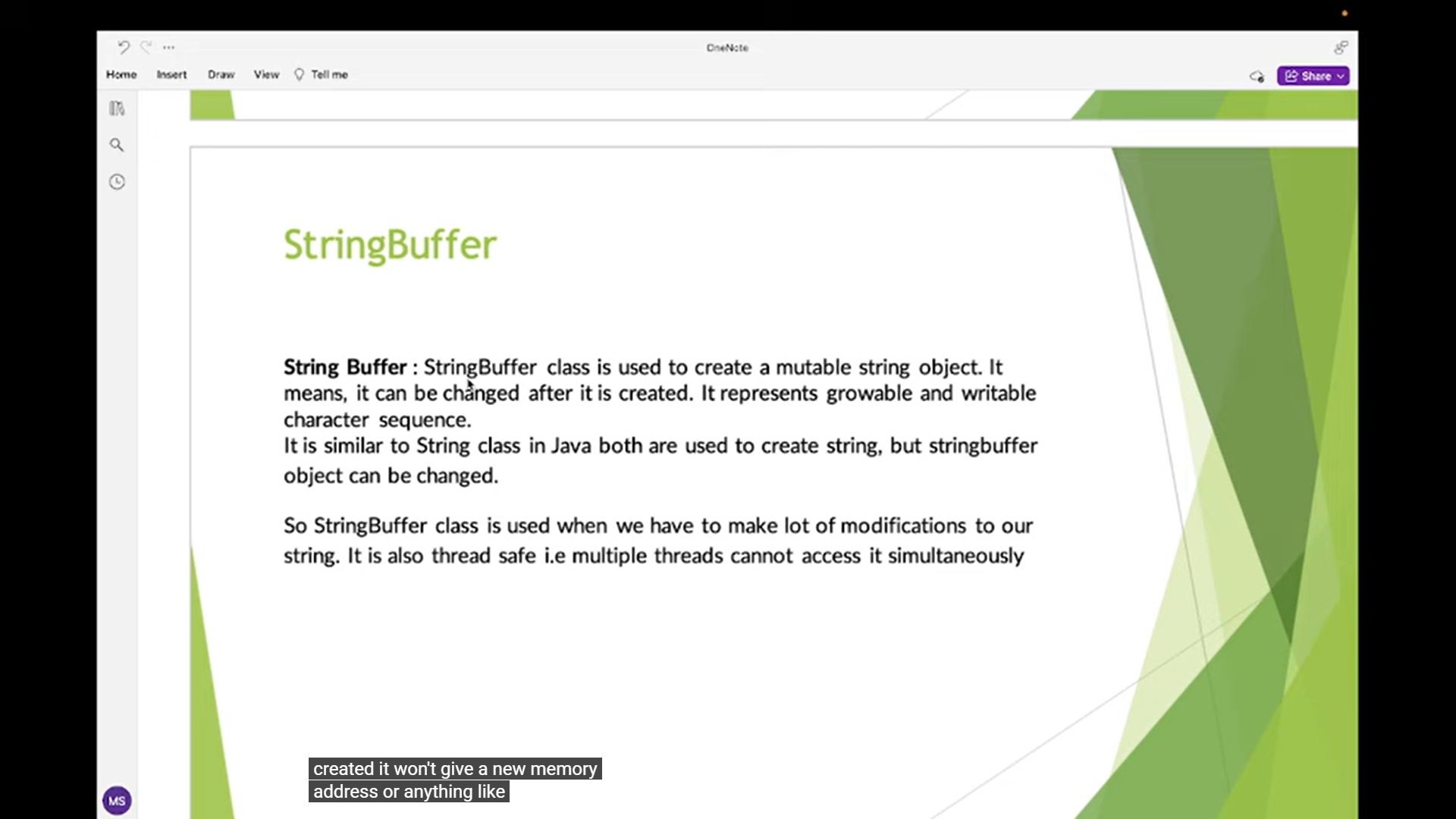
When we create String a =”Test” in constant pool for reference variable address has been allocated.

When I use a = a+z , the test reference a moved to “testall” and create address to variable.

Now for “test” there is no reference, then garbage collector will delete data(test) from string pool.

Note: here we are not modifying a variable.

Immutablity is a final class.



Methods:

1. Append()

Used to add strings.

1. Reverse()

We can reverse a string.



