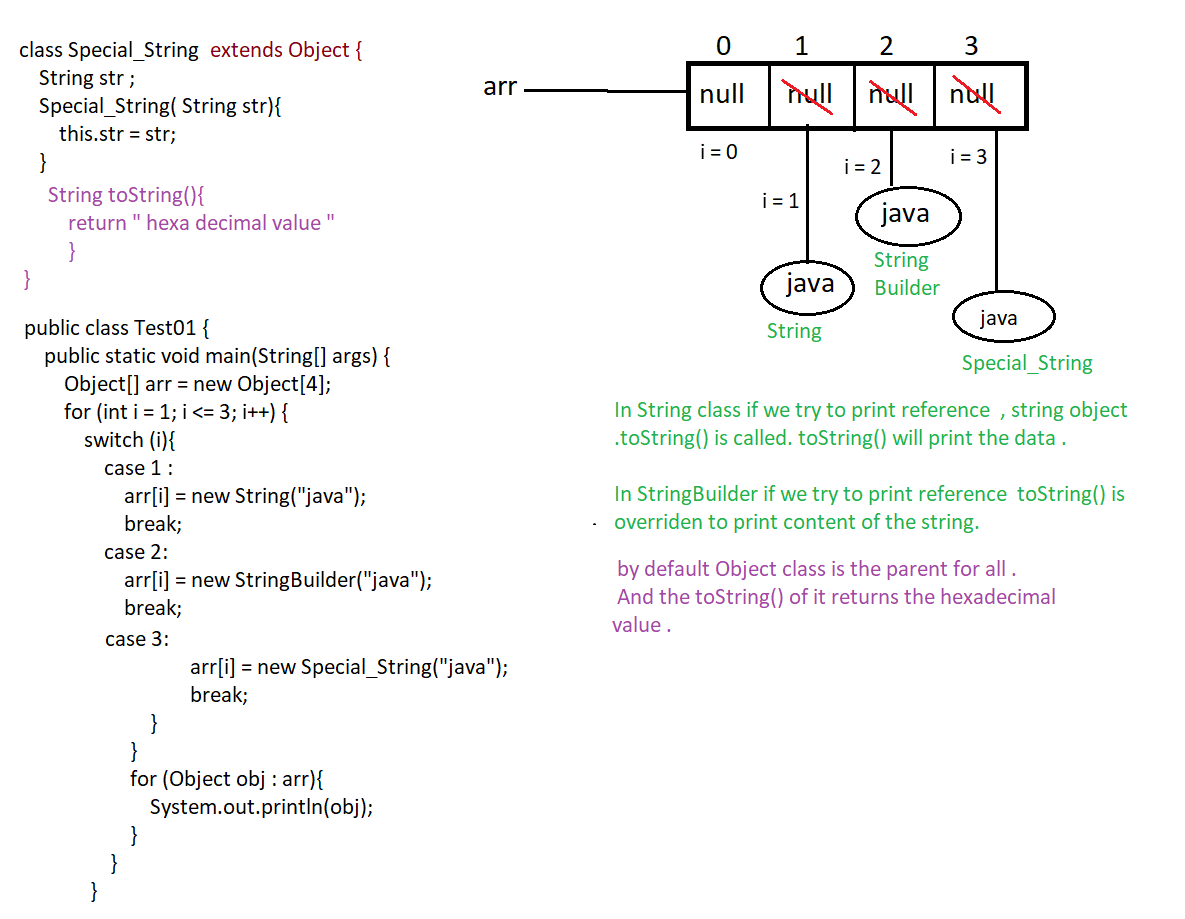
Eg: String\_Eg76

When you try to print object reference ,internally .toString() is called on it to print the reference . but manually we had written toString() which returns null value . so it is added to “text” variable .

Eg: String\_Eg77

// go through the code , it is based on above program String\_Eg76 .

Eg: String\_Eg78



Eg: String\_Eg79

// go through the command .

Eg: String\_Eg80

// go through the code.

Eg: String\_Eg81

// go through the code.

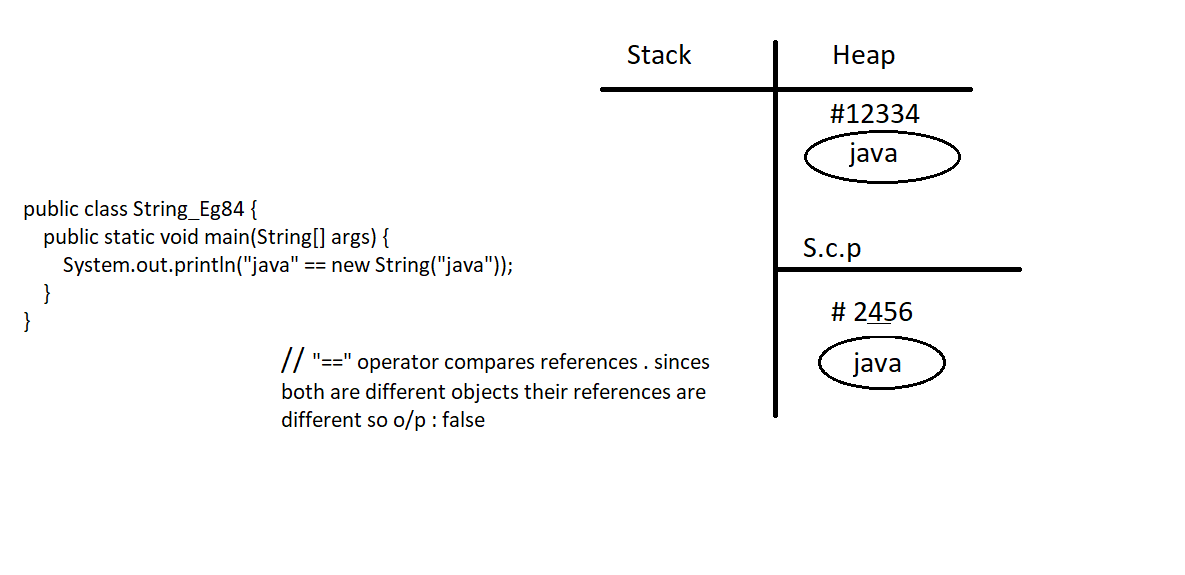
Eg: String\_Eg82

// go through the code.

Eg: String\_Eg83

// go through the code.

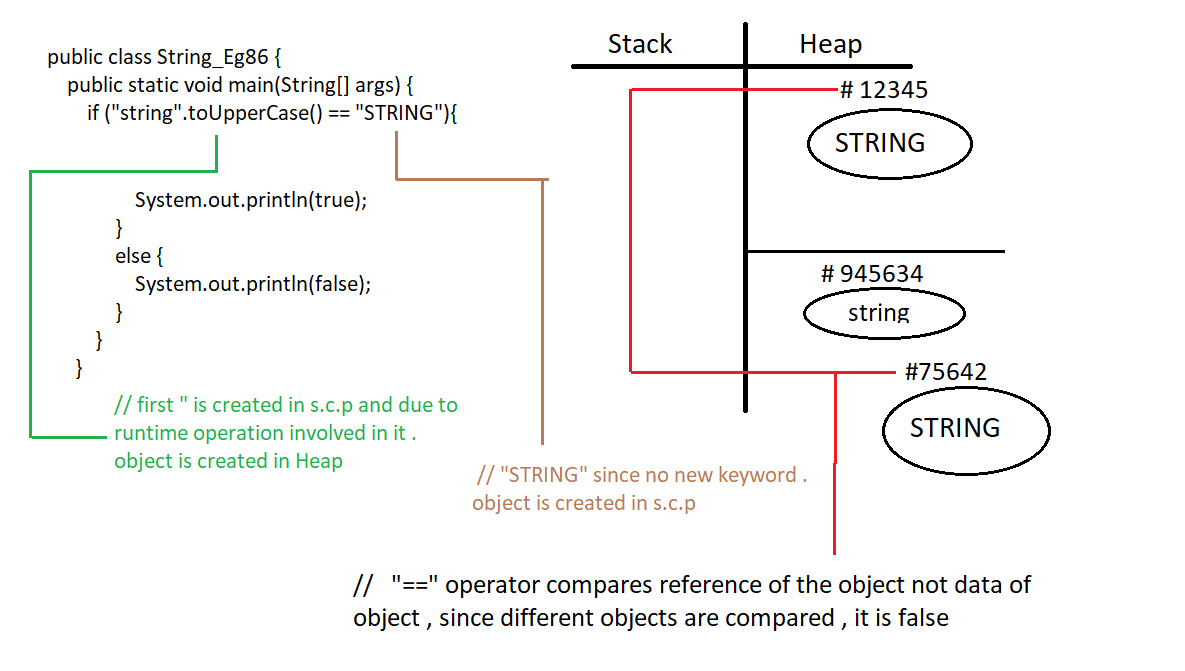
Eg: String\_Eg84



Eg: String\_Eg85

// go through the code

Eg: String\_Eg86



Note : String , StringBuilder , StringBuffer all these classes are final by default.

Eg:String\_Eg87

// go through the code.

Note: When the string concatenation is very frequent and code is need not be thread safe , we can opt for StringBuider(1.5v)

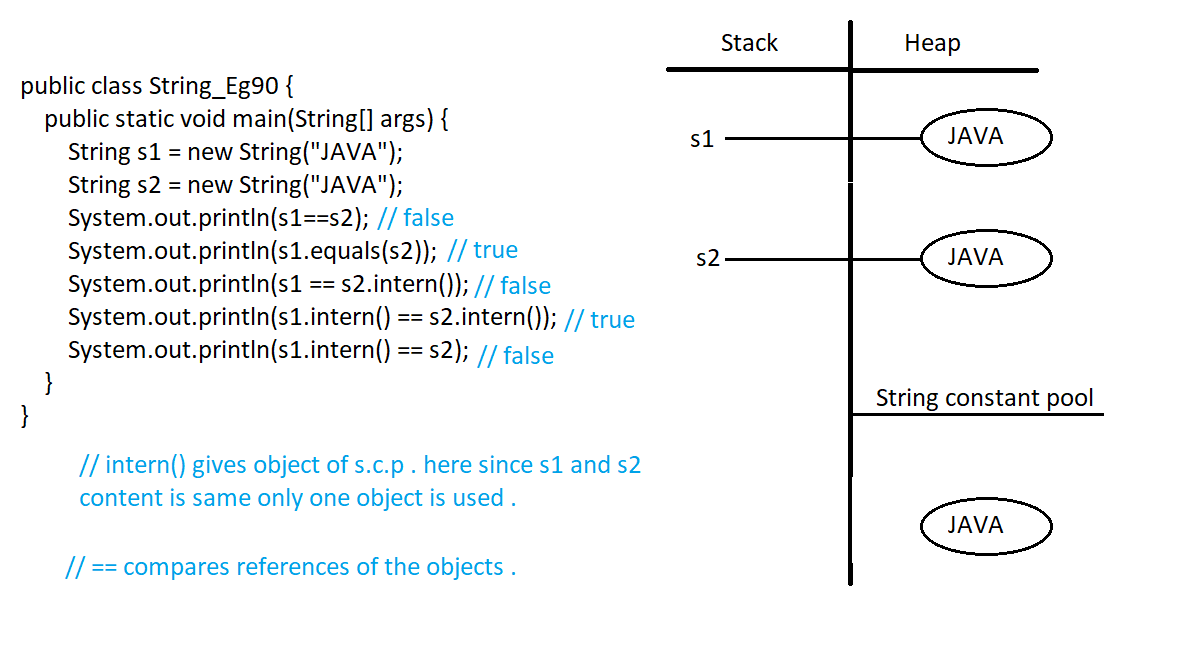
Eg: String\_Eg88

//go through the code.

Eg: String\_Eg89

// go through the code

Eg: String\_Eg90

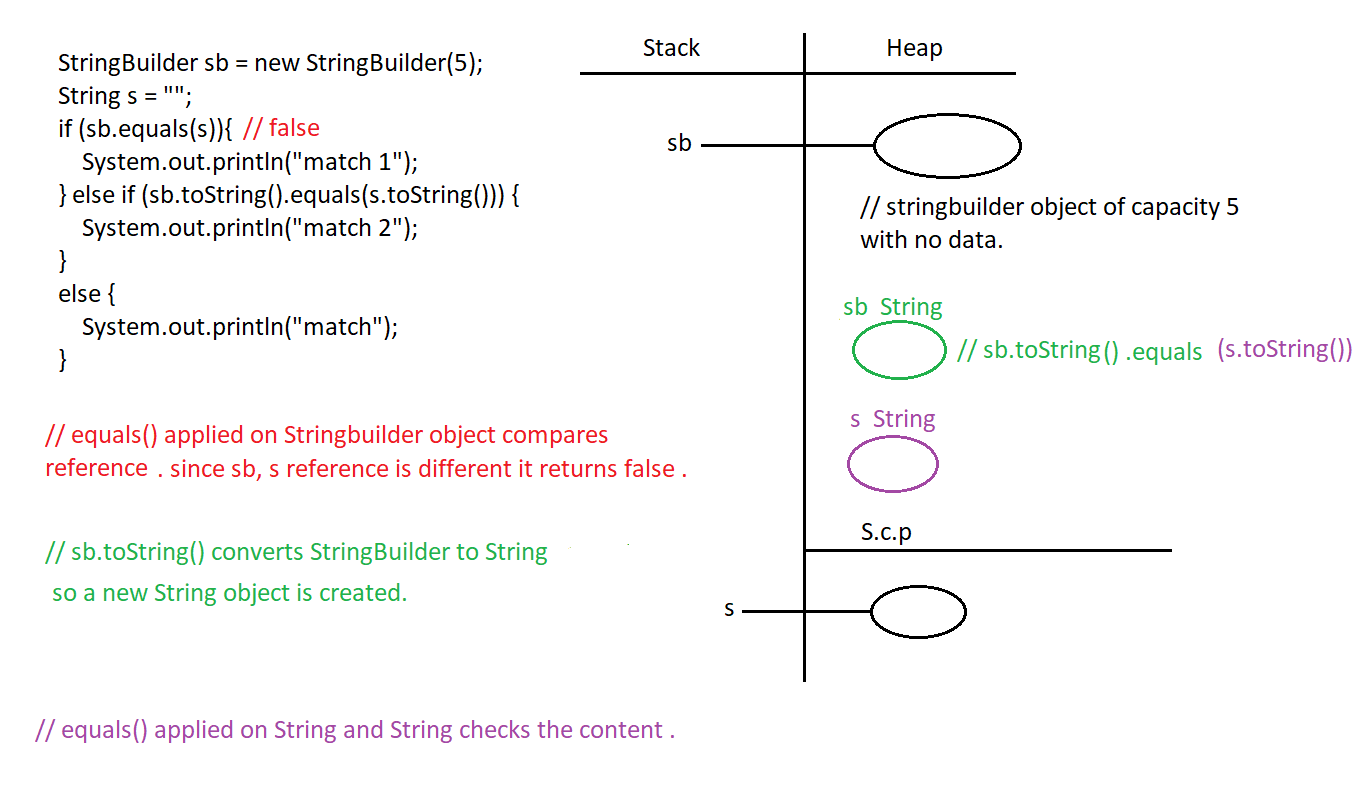


Eg: String\_Eg91

// go through the code .

Note : StringBuilder( ) constructor different types of parameters , check it in the IDE.

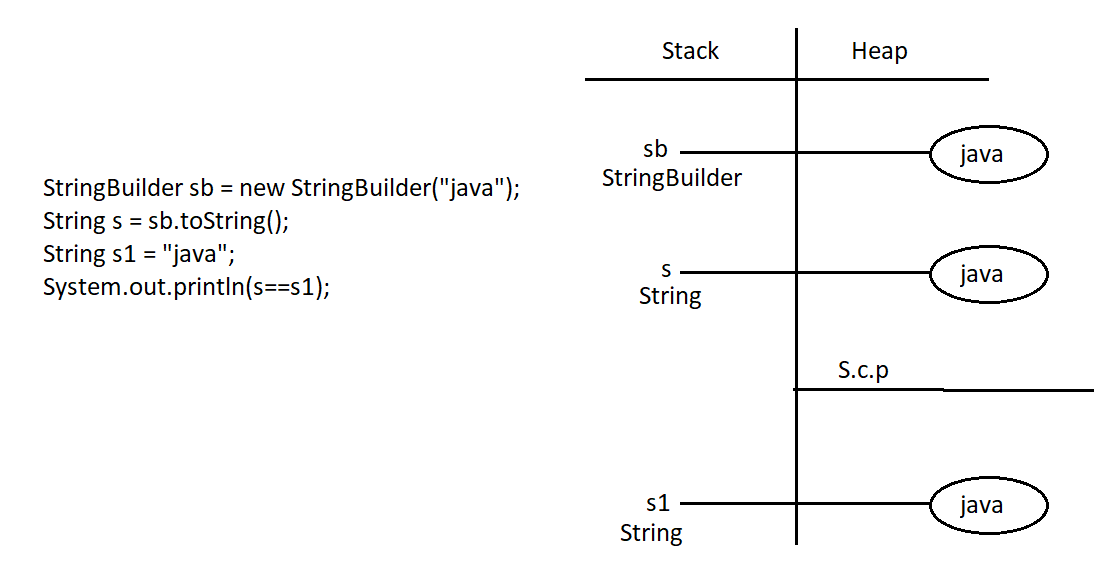
Eg: String\_Eg92



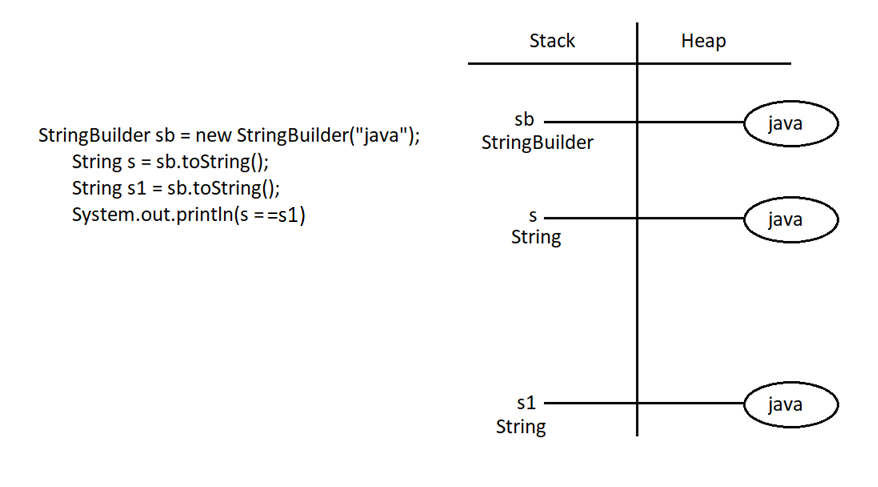
Eg: String\_Eg93

// go through the code.

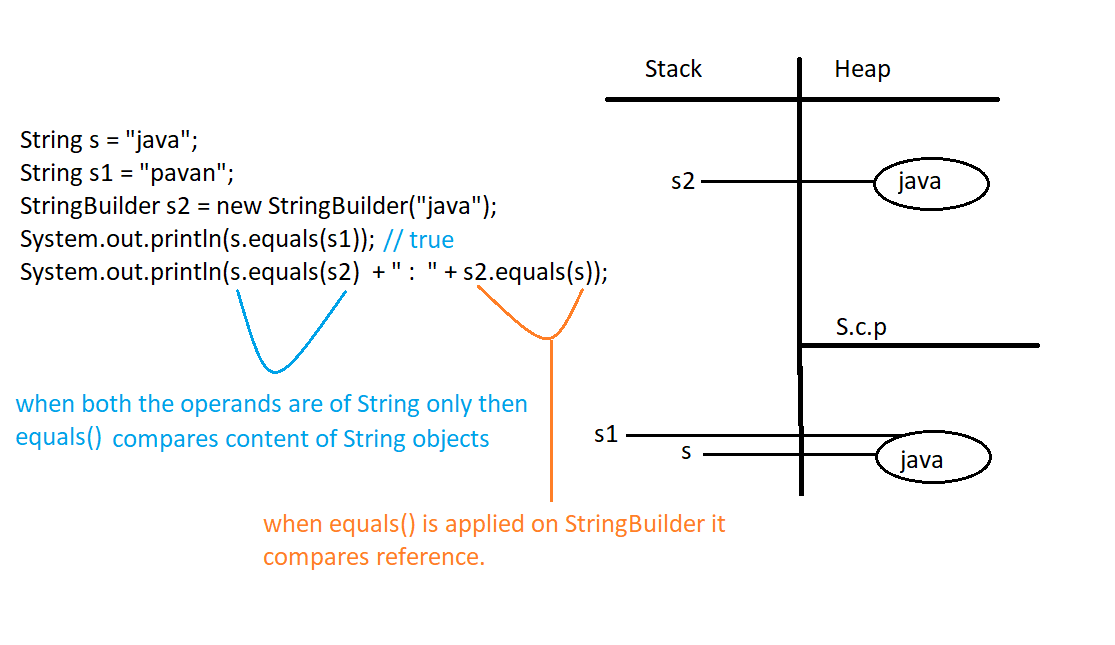
Eg: String\_Eg94



Eg: String\_Eg95



Eg:String\_Eg96



Eg:String\_Eg97

// go through the code.

Eg: String\_Eg98

// go through the code

Eg:String\_Eg99

// go through the code

Eg: String\_Eg100

// go through the code

Eg: String\_Eg101

// go through the code.