

# BLOCK 1

	123	x
	123123	ix
	123123	viii
	123	vii
		vi
fstring	x	v
estring	vii	iv
dstring	viii	iii
cstring	ix	ii
bstring	x	i
astring	x	0

- $astring == bstring$  (same memory location)  
True
- ~~Not~~ False (Diff memory location, diff content)
- False  
dstring creates a new String object @ runtime  
as one is a variable. even though content is same  
for dstring we have a new String object
- False  
estring creates a new String object @ runtime  
∴ diff from astring.
- False  
estring creates a new String object @ runtime  
∴ even though both LHS & RHS content wise are same they diff <sup>addresses</sup>
- False  
same reason as above creates a new String object @ runtime  
∴ even though both content same address differs
- False  
same reason as above - estring  
creates a new String object @ runtime  
∴ even though both content same address differs.

## BLOCK 2

123	xi
123	x
123	ix
123	viii
123123	vii
	vi
	v
String	iv
eeString	iii
ddString	ii
ccString	i
bbString	0
aaString	

• False

aaString = "123"

bbString = creates a new String object  
 $\therefore$  though content same address differs

• False

ccString = new String object created for ccString  
 $\therefore$  though content same address differs

• True

both retain are strings 123

• False

aaString is a String  
 eeString a String object created  
 $\therefore$  though content same address differs

• False

String is a String object  
 but 123123 is a String  
 $\therefore$  address differs even though content same

• No parentheses False

No parentheses  
 so concatenates which is not i.e. "13, 123123" == "123123"  
 neither content / address will be same

• 14. 123123123123