2.3 Authors: Rudram Joshi;
Block 1 Moinuddin Memon

```
int counter = 0;
int theNumberThree = 3;

String aString = "123";
String bString = "123";
String cString = "12" + "3";
String dString = "12" + theNumberThree
String eString = "123" + aString
String fString = "123123"
```

Address		Variable
10	"123"	
11	"123"	
12	"123123"	
13	"123123"	
	13	fString
	12	eString
	11	dString
	10	cString
	10	bString
	10	aString

Explanation of Output:-

1. true

Since the two Strings point to the same address (10), the output is True.

2. true

The concatenation of two literals have no effect on the pointing to the address 10 by aString.

3. false

The addition of literal "12" with variable theNumberThree creates a new String which points to address 11 in the memory model.

4. false.

The addition of variable aString with literal "123" creates a new String pointing to new address (12). Since, address is differente, the '==' operation returns false.

25	"123"	
26	"123"	
27	"123"	
28	"123123"	
29	"123123"	
	28	ffString
	25	eeString
	25	ddString
	27	ccString
	26	bbString
	25	aaString

Explanation of Output

5. false
Since aaString & bbString are two new String objects their addresses are different . Hence, the output is false.

6. false.
Address pointed by ccString is different from that of aaString, the output is false.

7. ~~false~~ true
aaString points to the address of literal "123". ddString also points to the same value as method1 returns "123" with same address.

8. true
Method2 returns a method call to method1 which in turns returns "123" pointing to the same address of aaString.

9. false.
Addition of output of method 1 with that of method 2 creates a new String object which points to a different address. Hence, the output is false.