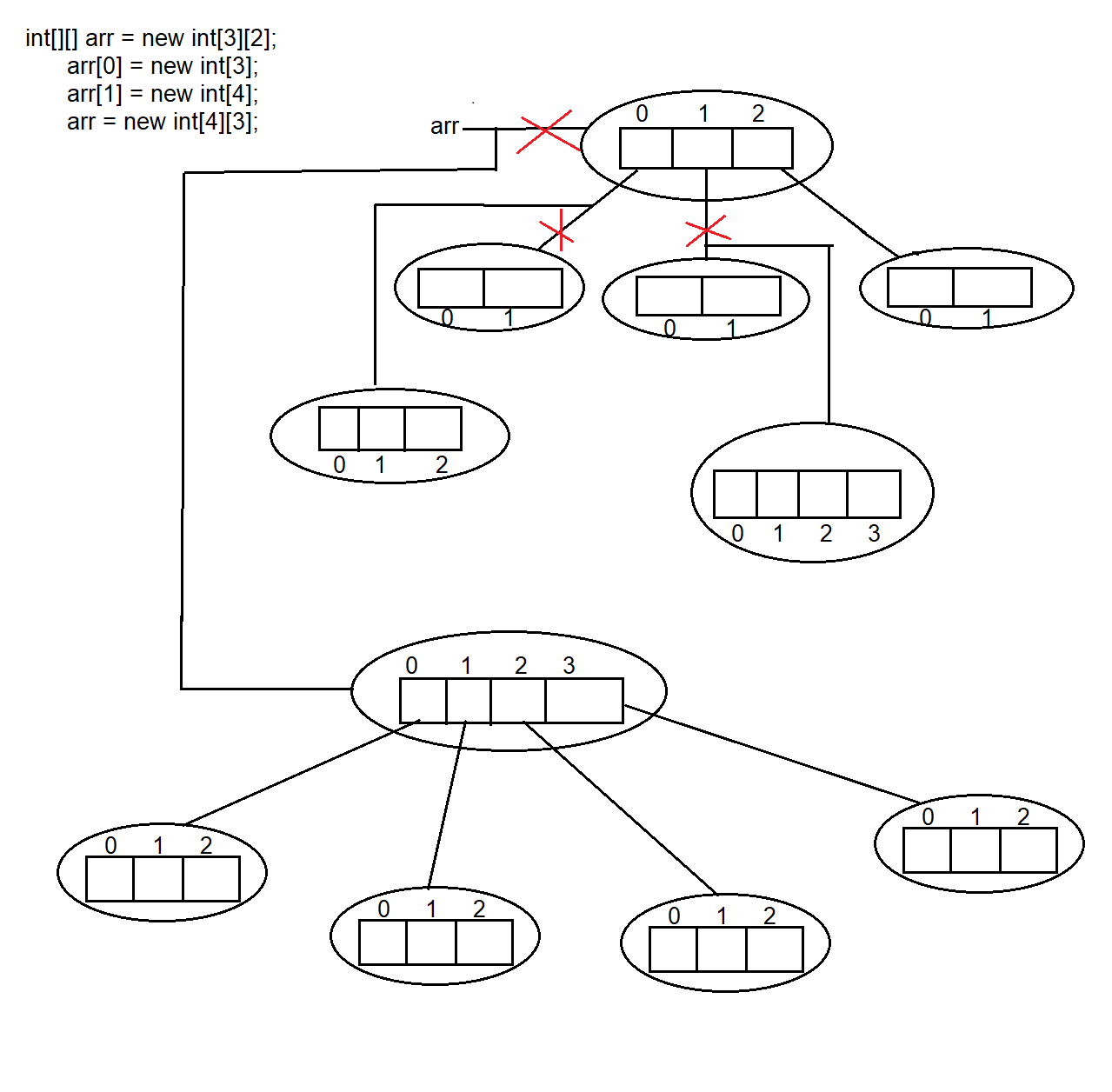
Eg: Array\_Eg17

// in array re-assigning is possible

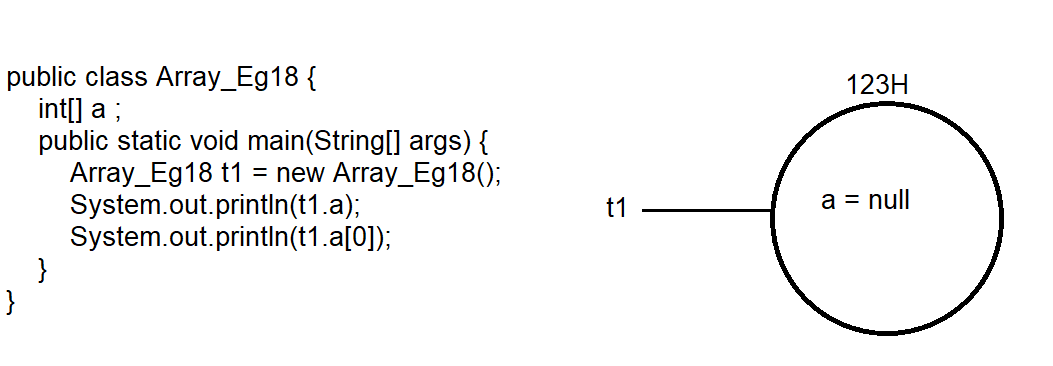


Total objects created = 11 ( circles )

Objects eligible for garbage = 6

Red coloured cross marks are reassigned , since objects have no reference variable to store their address , they are cleaned by garbage collector .

Eg: Array\_Eg18



Here first main method is executed and object for t1 is created , and in that object an int type array reference is created ( array object will be created only when new keyword is used in declaration )

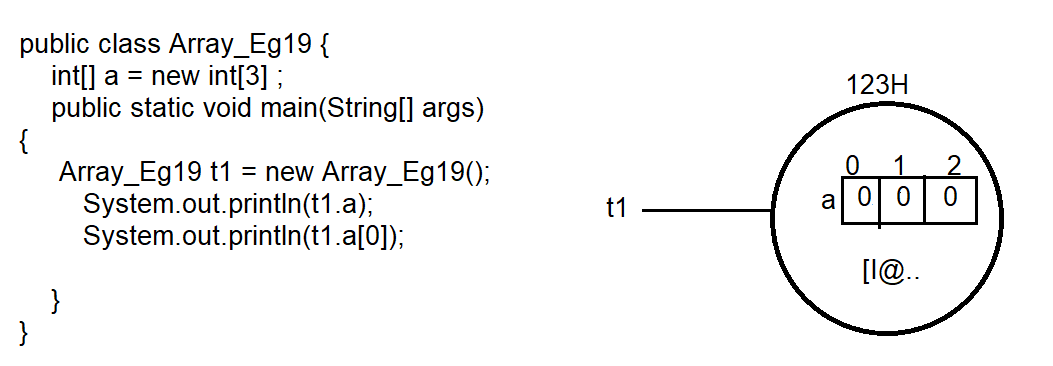
Here array object is not created , jvm identifies it as reference variable (a) , and gives default value of reference variable

Note : for all data types reference variable default value is null , if object is created using new keyword.

When tried to print t1.a null is printed

In t1.a[0] since we are trying to perform an operation null , it leads to null pointer exception

Eg: Array\_Eg19



here object for array a is created , and size is declared .

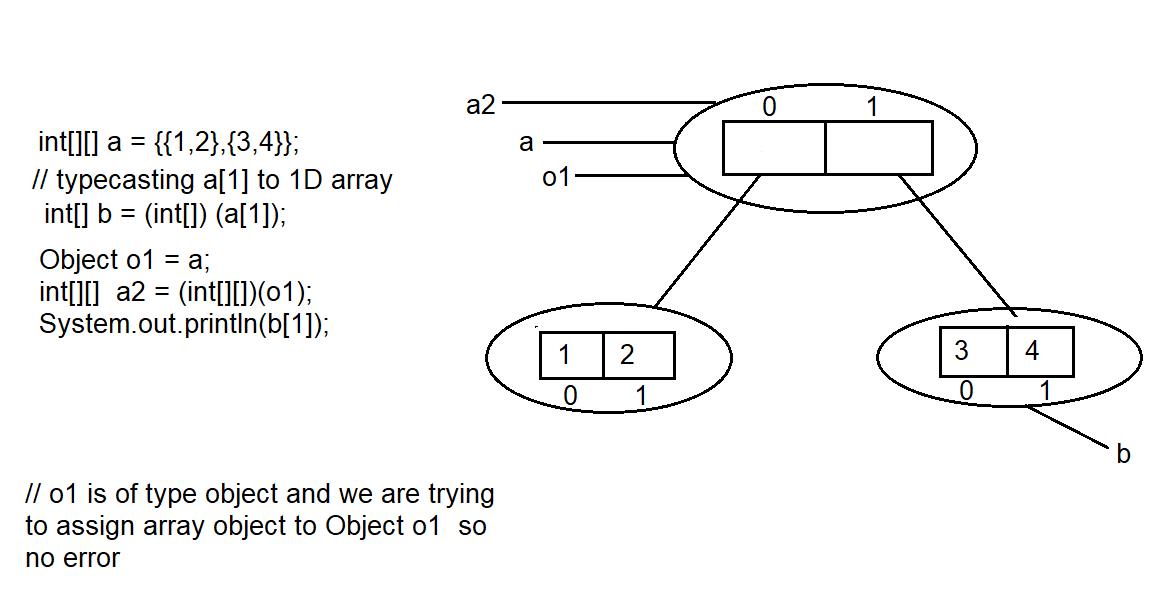
when you try to print a it gives [I@123H ( since we are trying to print object of the array a )

t1.[0] gives the default value at 0th location w.r.t data type

Eg: Array\_Eg20

Here a is local variable , it cannot be used without intialization .

Eg: Array\_Eg21



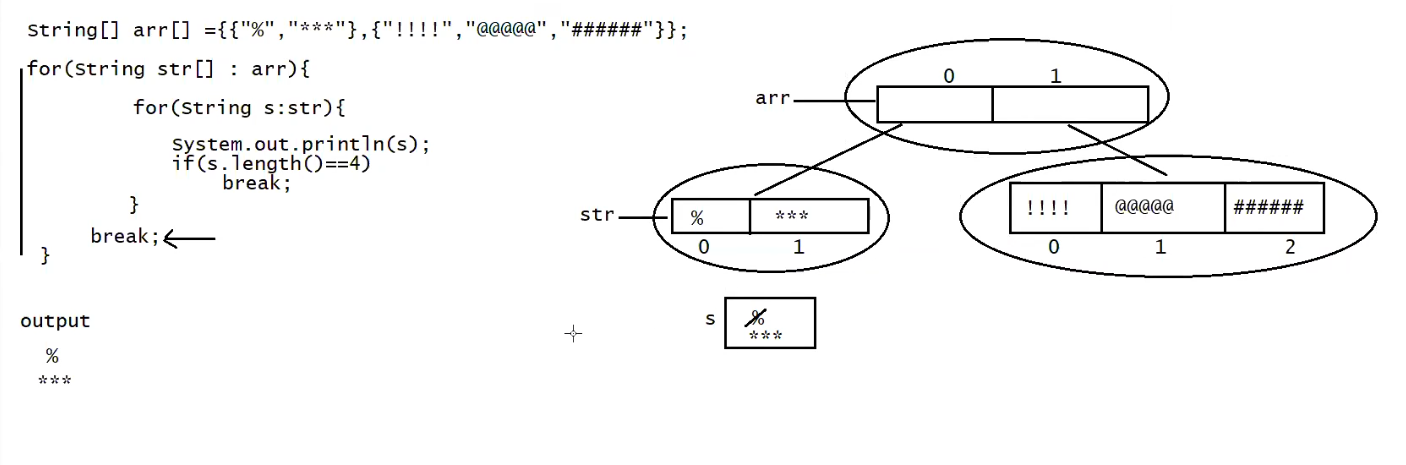
Here a[1] is type casted to 1 dimensional int array and stored in 1dimansional int array b

Similarly o1 is 2-Dimensional array it is type casted to 2-dimensional int array and stored in 2-dimensional int array a2

Eg: Array\_Eg22

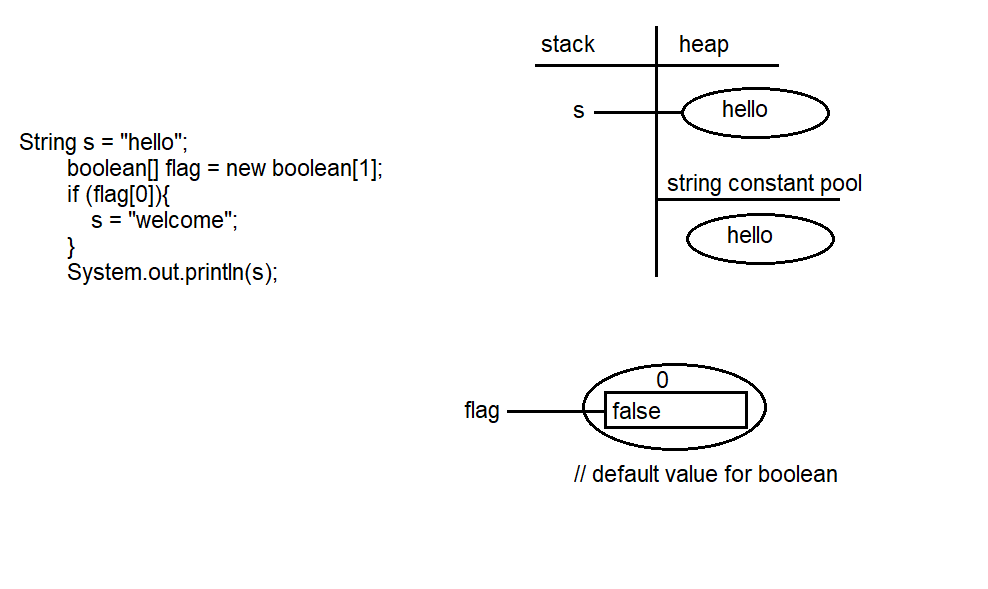
// go through the program .

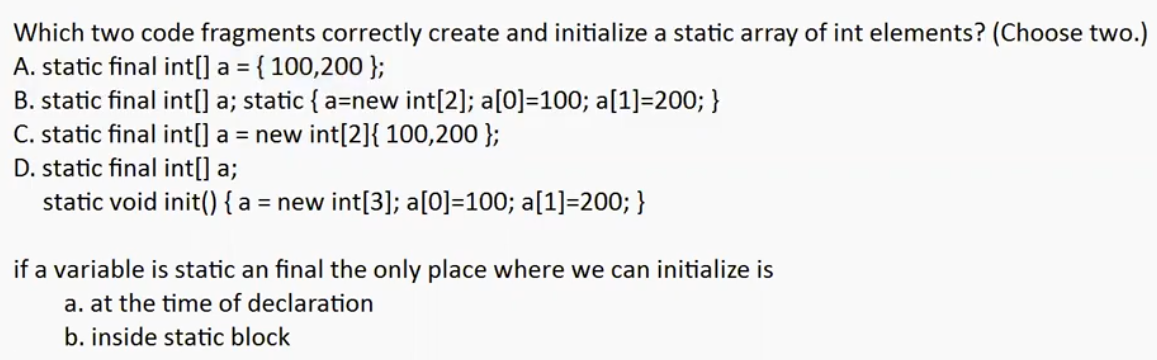
Eg: Array\_Eg23



The break in the outer loop terminates the outer loop .

Eg: Array\_Eg24





In option b when elements in the array are initialized , size can’t be declared . if it is done it may lead to compile time error.