

Tuesday Sep. 18
Lecture 4

cursor.item

ACCOUNT

⊗

id

STRING

F

e1 ⊗ e2

↓

is-equal

~~cursor.item.id.is-equal(id)~~

cursor.item.is-equal(id) X

ol.is-equal(~~~~~)

ol ~ ~~~~~

Programming with Expanded Types

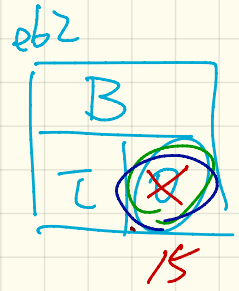
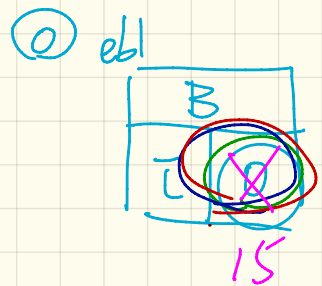
aliasing

```

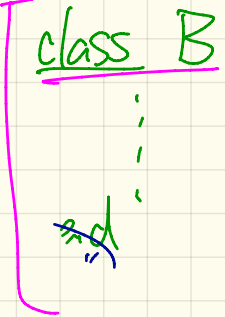
expanded class
  B
  feature
    change_i (ni: INTEGER)
      do
        i := ni
      end
  feature
    i: INTEGER
  end
  S: STRING
  
```

```

1 test_expanded: BOOLEAN
2 local
3   eb1, eb2: B
4 do
5   Result := eb1.i = 0 and eb2.i = 0
6   check Result end
7   Result := eb1 = eb2
8   check Result end
9   eb2.change_i (15)
10  Result := eb1.i = 0 and eb2.i = 15
11  check Result end
12  Result := eb1 /= eb2
13  check Result end
14  end
  
```



Result := eb1 /= eb2



fast local

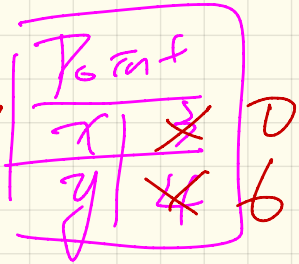
```

do eb1, eb2: B
  eb1 = eb2 T
  create eb1.make (1)
  create eb2.make (1)
  eb1 = eb2 F
  
```

→ Point p1 = new Point(3, 4);

Point p2 = p1;

→ println (p1.x + " " + p1.y); p2 3, 4

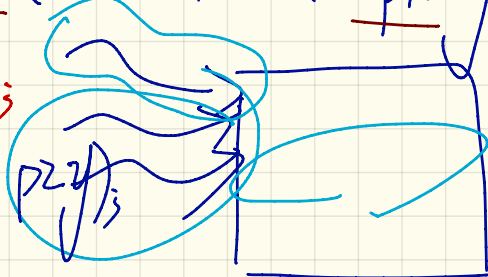


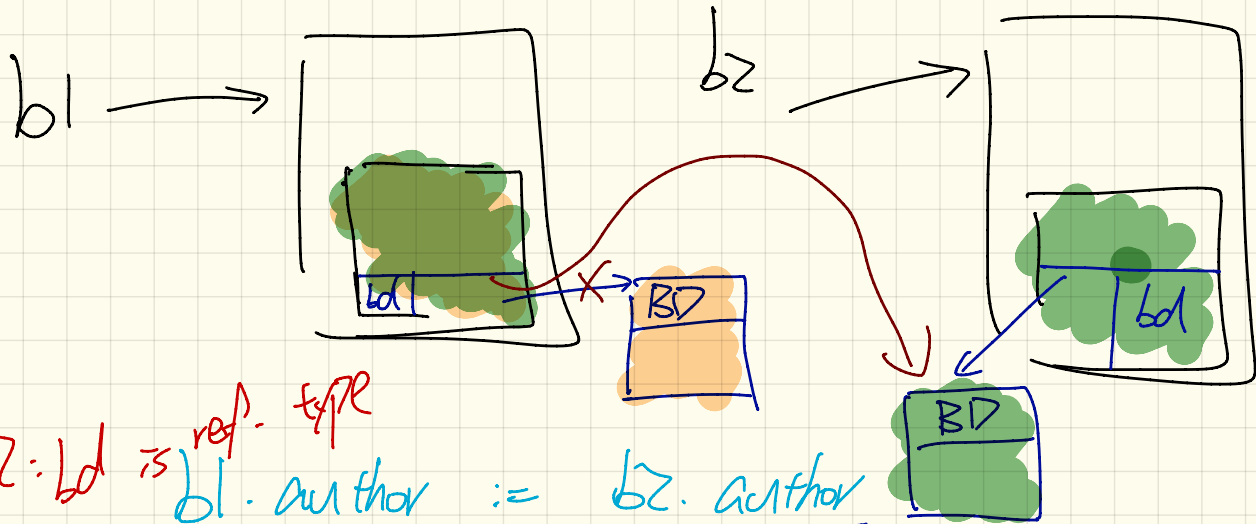
→ p2. moveUp(2);

println (p1.x + " " + p1.y); 3, 6

→ p1. moveLeft(3);

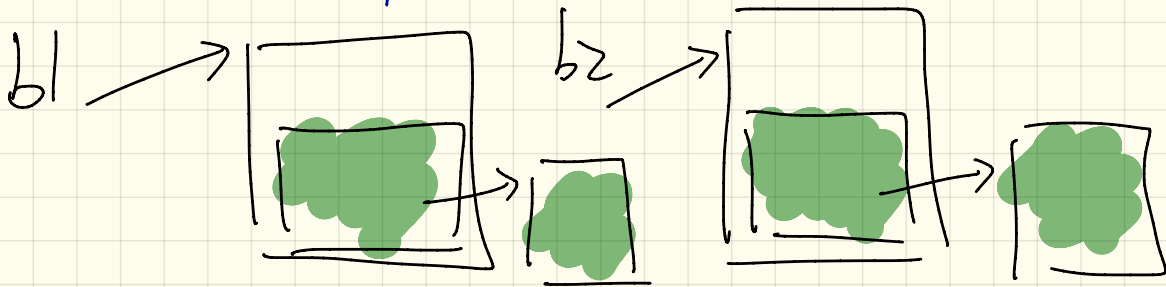
→ println (p2.x + " " + p2.y); 0, 6



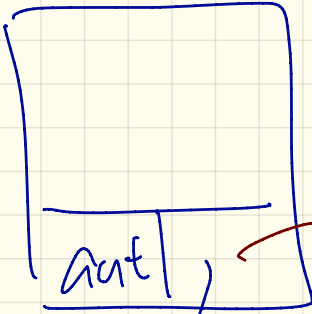


② Case 2: bd is ^{ref. type} $b1.author := b2.author$

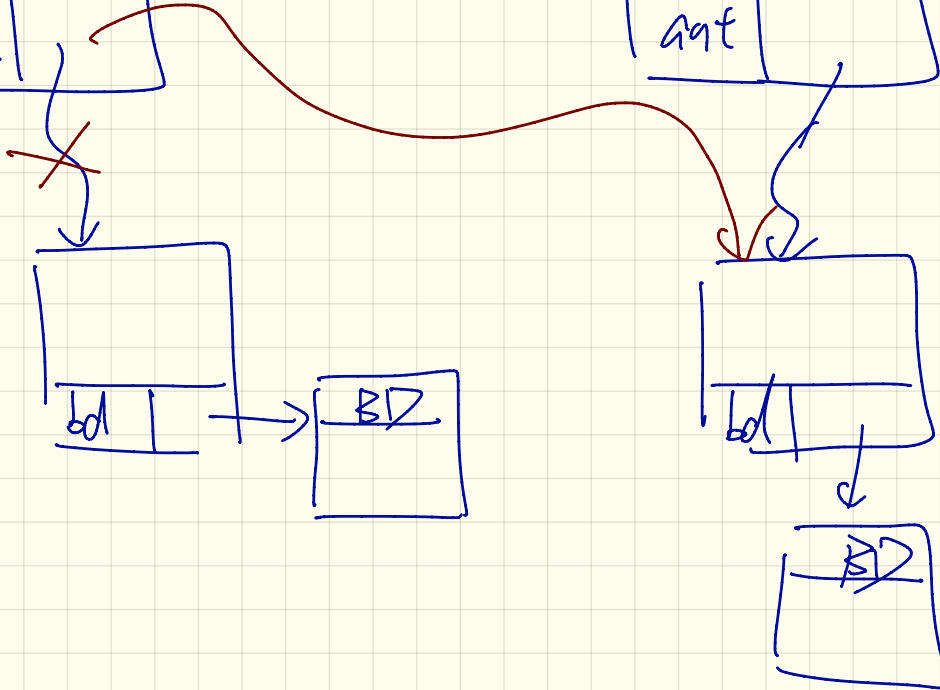
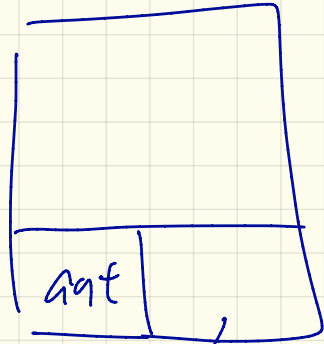
① Case 1: bd is expanded type



b1

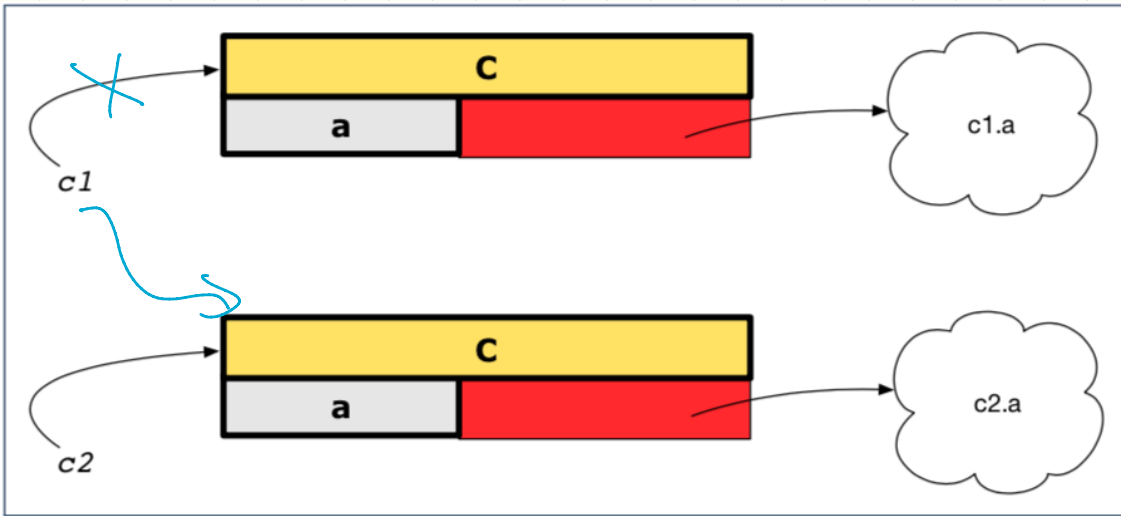


b2



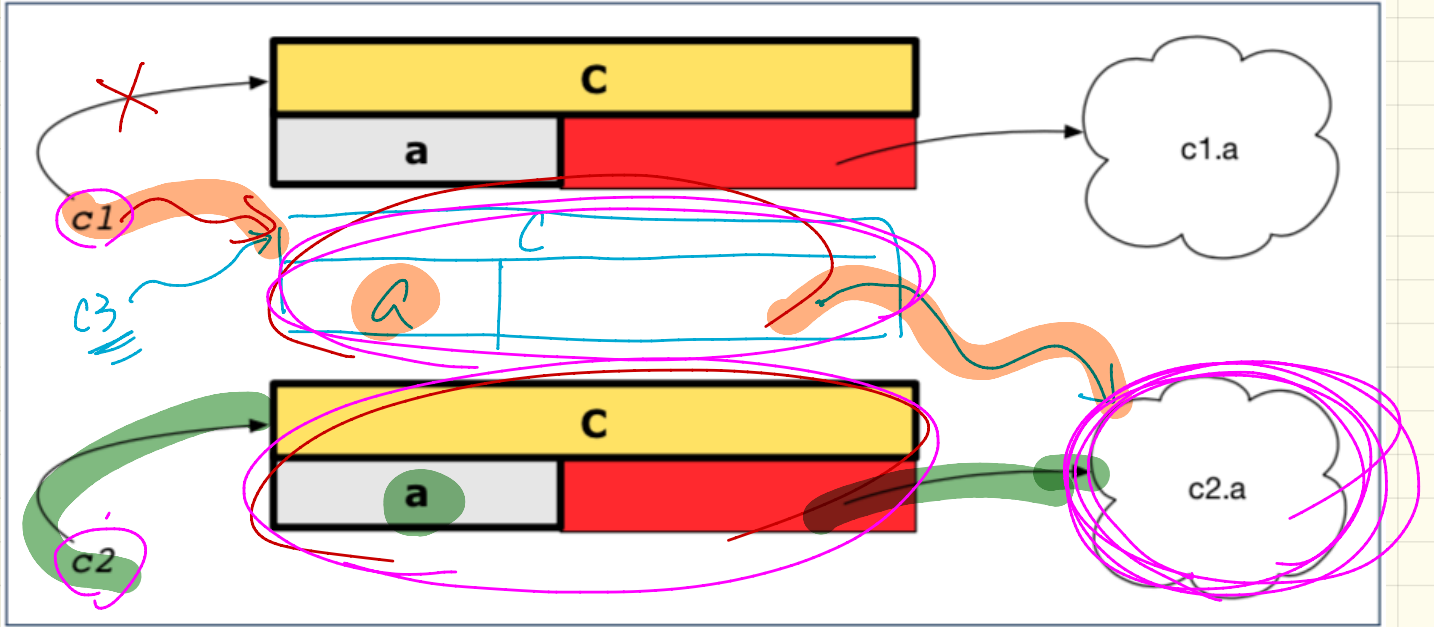
b1. author :=
b2. author

Reference Copy : $C_1 := C_2$



$$C_1 = C_2$$
$$C_1.a = C_2.a$$

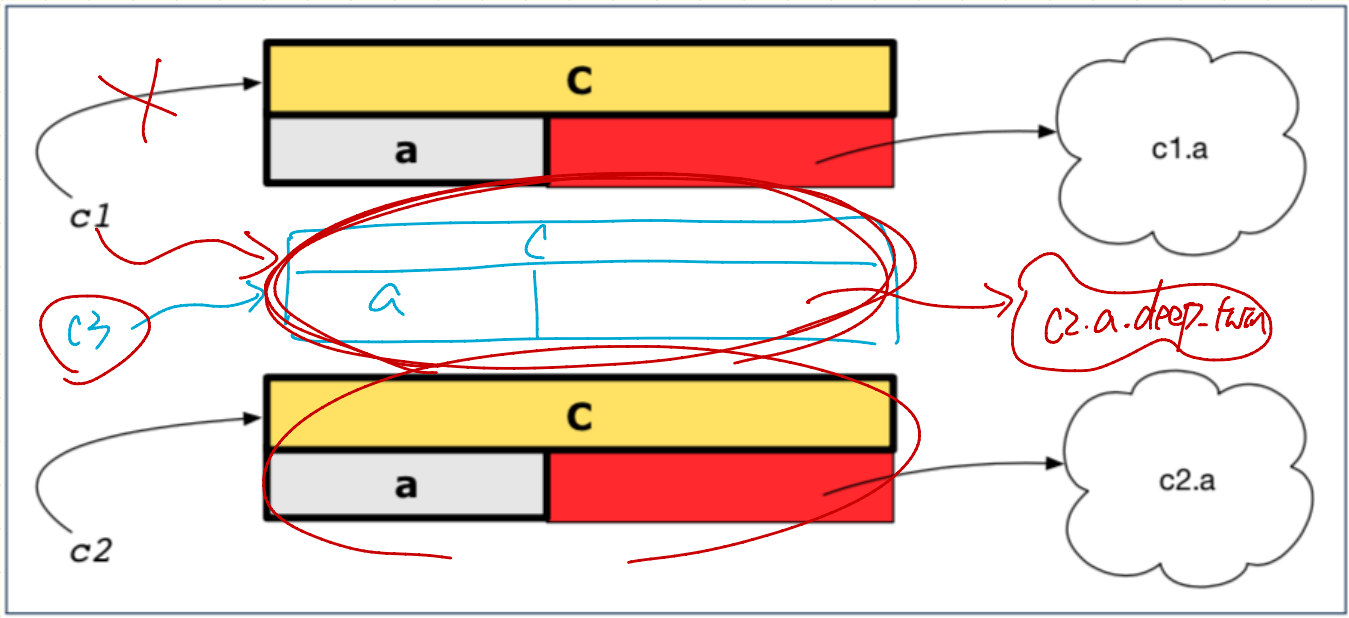
Shallow Copy : $c_1 := c_2.\text{twm}$



$c_3.a := c_2.a$
 $c_1 = c_2$ F

$c_1.a = c_2.a$ T

Deep Copy : $c1 := c2.\text{deep_twim}$



$c3.a := c2.a.\text{deep_twim}$
 $c1 = c2 \quad F \quad c1.a = c2.a \quad F$

Ref. vs. Shallow vs. Deep Copies

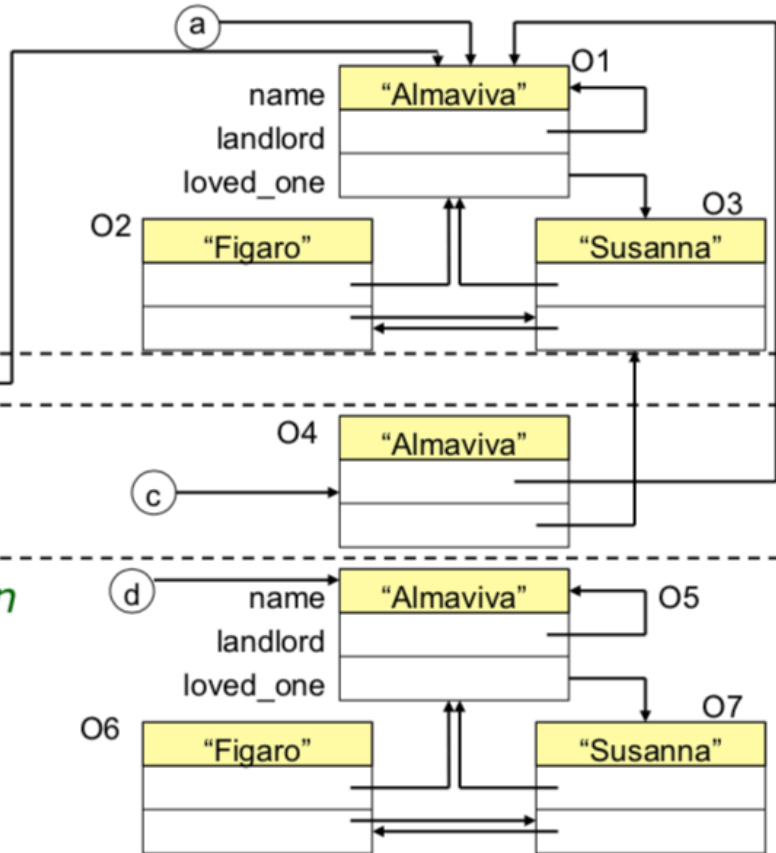
▪ Initial situation:

▪ Result of:

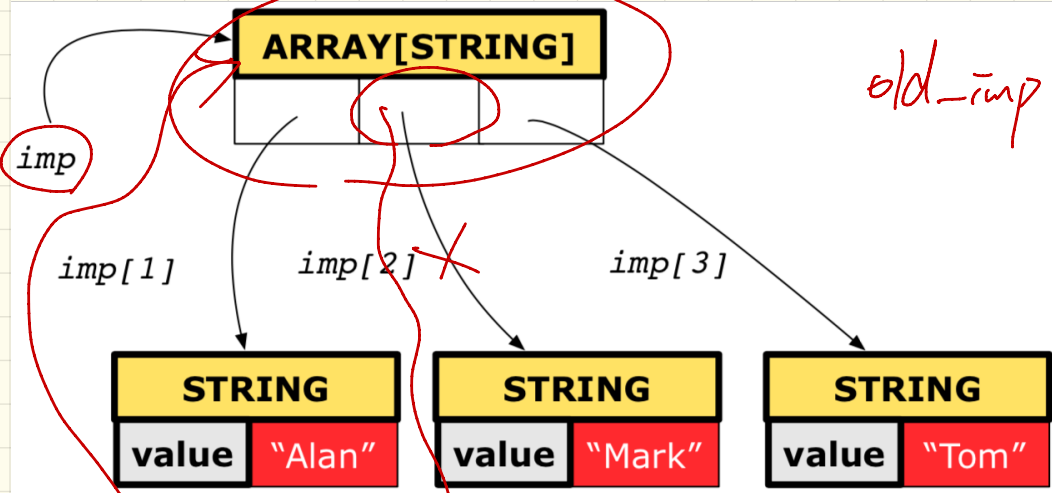
$b := a$

$c := a.twin$

$d := a.deep_twin$

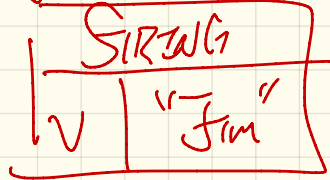


Copying Collection Objects: Reference Copy & Make Changes

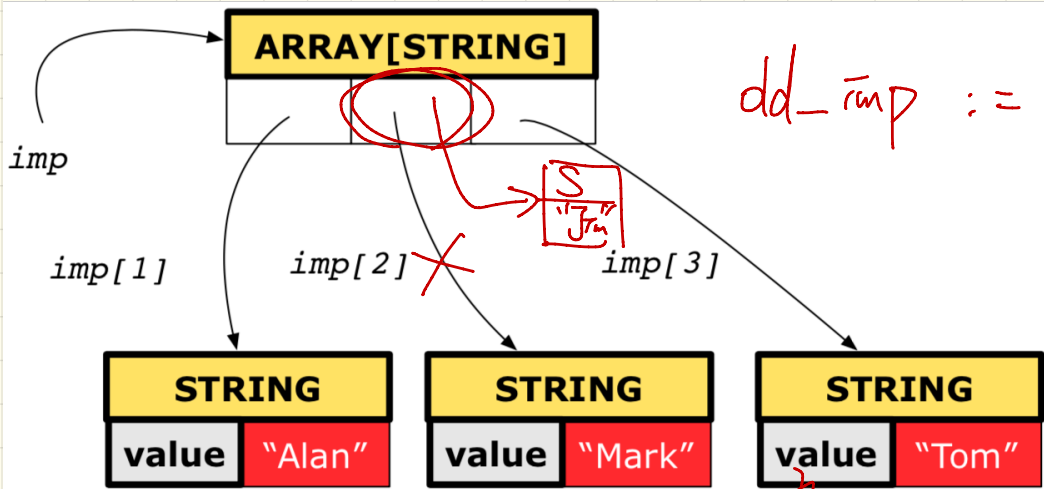


old_imp := imp

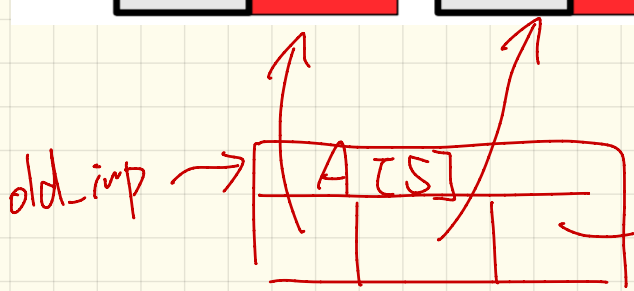
old_imp



Copying Collection Objects: Shallow Copy & Make 1st-level changes

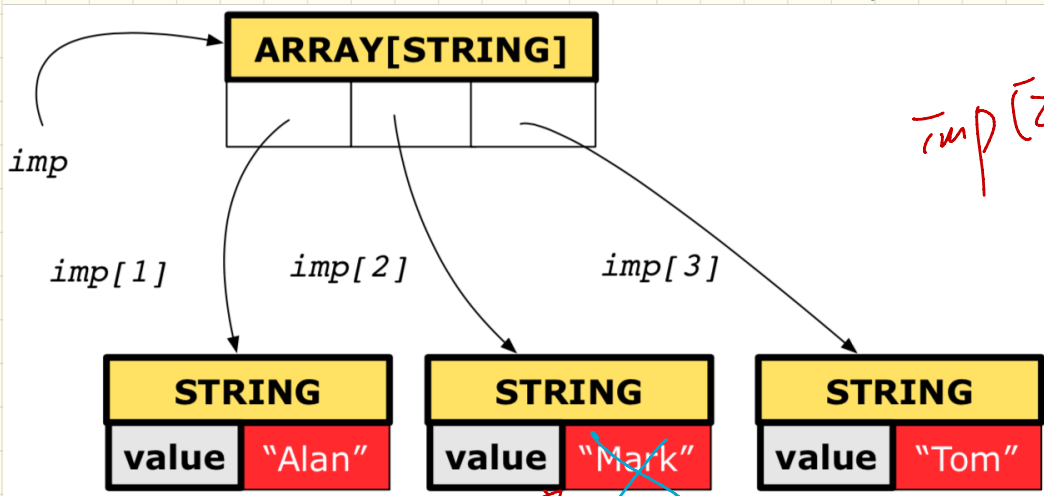


$dd_imp := imp.\text{toArray}$



$imp[i] = dd_imp[i] T$
 $imp = dd_imp T$

Copying Collection Objects: Shallow Copy & Make 2nd-level changes



imp[2].append("***")

