



Yes, but only static final fields! ( final by default )

**Is that legal to have fields in interfaces?**

No, since it's a final or constant fields you cannot modify it after the initialization!

However, if you assign an object to that, you can modify the object until the reference is not changed.

**Can modify their values?**

```
public interface Vehicle {  
  
    static final ArrayList<String> manufacturedVehicles = new ArrayList<>();  
  
}
```

manufacturedVehicles

ArrayList@779

Fields

elementData

Object[0]@781

size

0

modCount

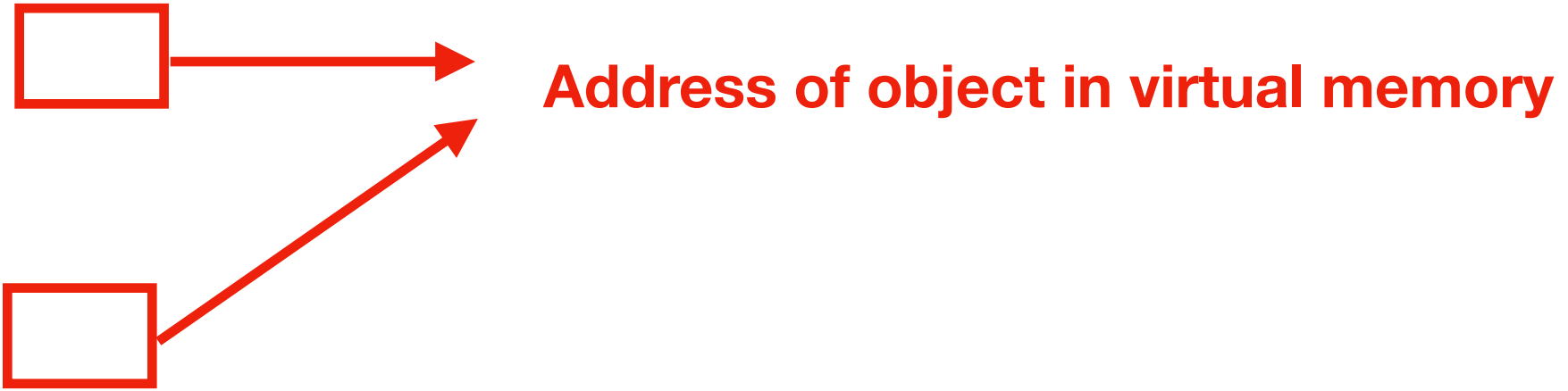
0

Once you run the program, this static field is created and appears in the memory.





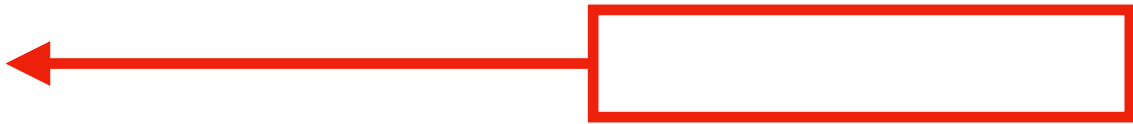






The capacity of the array

**An array of objects which holds  
the ArrayList items**



An integer-typed field that stores the  
current number of our ArrayList items



**An integer-typed field that stores the number of modifications have applied to our ArrayList object.**





**Each instance of the ArrayList  
contains these 3 fields**

















public interface Vehicle {





```
static final ArrayList<String> manifestVehicless = new ArrayList<>();
```

elementData

Object[0]@781



manufacturedVehicles

ArrayList@779

Fields

elementData

Object[0]@781

size

0

modCount

0