ArrayLists Methods: size & add

mylist.size()

- returns the number of items in mylist.
- Note contrast to String: .size() vs .length()

```
UI.printf("The garden had %d flowers\n",  this.flowers.size());
```

item must be of the right type

mylist.add(item)

- adds the item at the end of the ArrayList

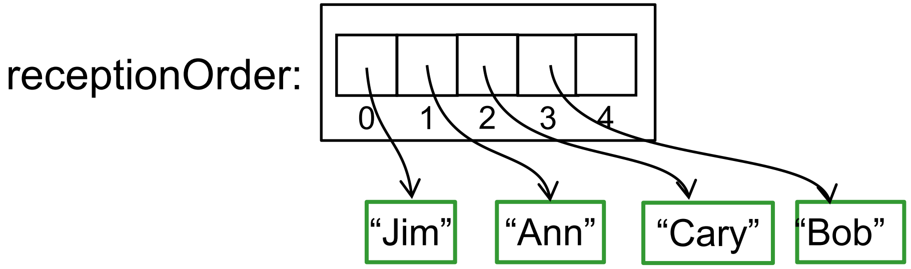
```
balloons.add( new Balloon(x, y, color) );
```

item must be of the right type

mylist.add(index, item)

- inserts the item at position index (0 .. size)

```
int pos = UI.askInt("Where do you want "  +name);
receptionOrder.add(pos, name);
```



ArrayLists Methods: get & set

mylist.get(index)

- returns the item at position *index* (0 .. size-1)

mylist.set(index, item)

- replaces the current value at position *index* with *item*
- returns the old value at *index*
- *index* must be 0 .. size-1

```
UI.print("Which two units do you want to swap?");
int first = UI.askInteger("first");
int scnd = UI.askInteger("second");
```

```
if (first>=0 && first< attackList.size() &&
 scnd>=0 && scnd<attackList.size() &&
 first!=second){
```

```
String temp = attackList.get(first);
attackList.set(first, attackList.get(scnd));
attackList.set(scnd, temp);
```

Or

```
attackList.set(first, attackList.set(scnd, attackList.get(first)));
```

ArrayLists Methods: isEmpty & clear

mylist.isEmpty()

- returns true iff there are no items in mylist.

```
// Make each unit advance, if there are any units
if ( ! attackList.isEmpty() ) {
    for (Unit unit : attackList) {
        unit.checkPath();
        unit.advance(3);
    }
}
```

mylist.clear()

- removes all values from the list.

```
// Restart and clear the list of all elements.
public void doRestart() {
    UI.clearGraphics();
    this.flowers.clear();
}
```

ArrayLists Methods: contains & remove[1]

mylist.contains(item)

- returns true if the item is somewhere in mylist

mylist.remove(item)

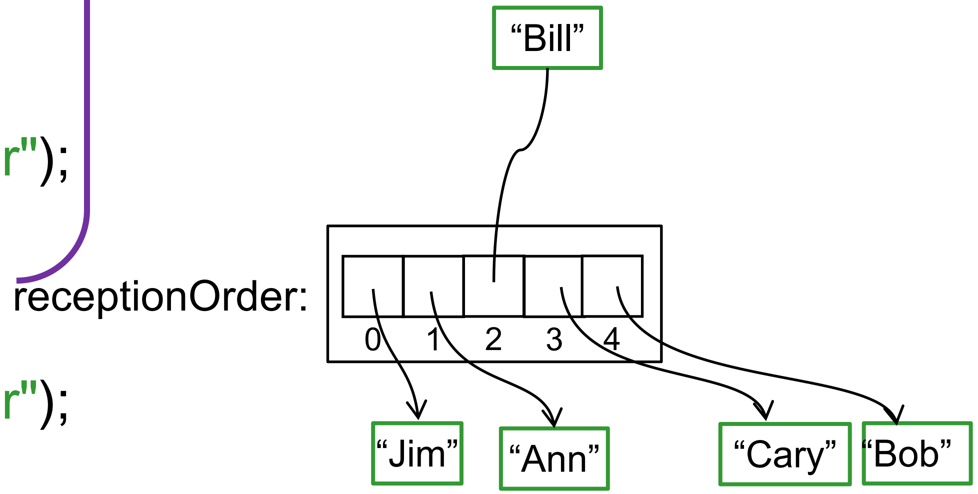
- removes the item, if it is present, and shuffles later items down
- returns true iff item was removed

first occurrence of item if item is at several places in the list

```
// Respond to a “Remove Person” button
String name = UI.askString("Person to remove");
if (receptionOrder.contains(name)){
    receptionOrder.remove(name);
}
else {
    UI.println("That person is not in the reception order");
}
```

Or

```
if ( ! receptionOrder.remove(name) ){
    UI.println("That person is not in the reception order");
}
```



ArrayLists Methods: indexOf & remove^2

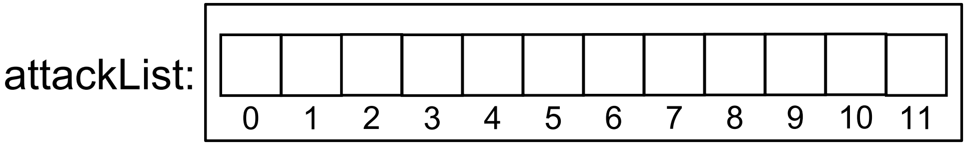
mylist.indexOf(item)

- returns the position of item in mylist
- returns –1 if the item is not present

```
// Report position on waiting list
String name = UI.askString("Your name:");
int index = waitingList.indexOf(name);
if (index == –1) {  UI.println("You are not on the waiting list"); }
else                  {  UI.println("You are number " +  index + " in order"); }
```

mylist.remove(index)

- removes the item at position *index* (0 .. size-1)
- returns the value that was removed



```
// Remove every third unit from attackList
  for (int index=2;  index<attackList.size(); index =index +2){
       attackList.remove(index);
  }
```

Removing many items from an ArrayList

Remove balls past the right edge:

```
for (Ball ball : this.balls) {
    if (ball.getX()>RIGHT) {
        balls.remove(ball);
    }
}
```

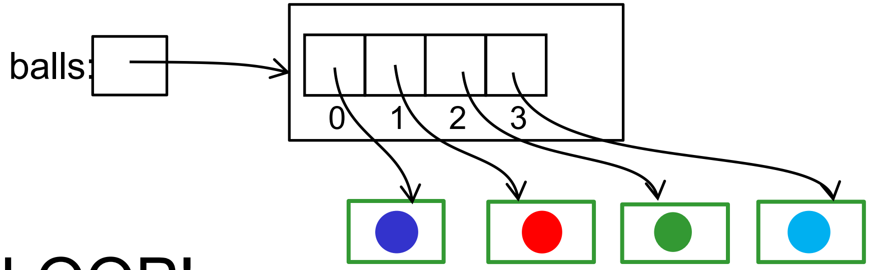
Program will CRASH!

Not allowed to change the list while iterating with a "for each" loop

Must use a standard for loop

```
for (int index = 0 ;  index < this.balls.size();  index++) {
    if (this.balls.get(index).getX()>RIGHT) {
        this.balls.remove(index);
    }
}
```

Won't crash,
But still won't work properly!

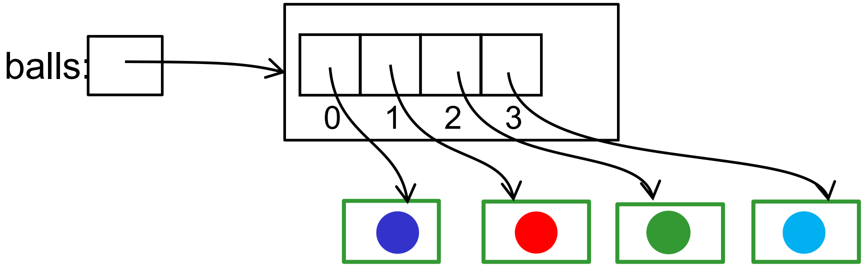


BE CAREFUL WHEN MODIFYING THE LIST IN A LOOP!

Removing items from ArrayList in a loop

1) Step back after removing item

```
for (int num = 0 ;  num < balls.size();  num++) {
        if (balls.get(num).getX() > RIGHT) {
        balls.remove(num);
        num = num - 1;
    }
}
```



Or

2) iterate backwards from the end.

```
for (int num = balls.size()-1 ;  num >= 0;  num--) {
        if (balls.get(num ).getX() > RIGHT) {
        balls.remove(num);}
}
```