# ArrayList

CS 1400 2024
Cody Henrichsen

1

## ArrayList Vocabulary

- ▷ < >
- ▷ Type
- ▷ Index
- ▷ Indices
- ▷ .size()
- ▷ .add(item)
- ▷ .add(index, item)
- ▷ .set(index, item)
- ▷ .remove(index)
- ▷ .get(index)

2

## ArrayList Info

- ▷ Part of the Java Collections Framework
- ▷ Only holds Objects
  - ▷ No primitives!!!!
    - ▷ Wrapper classes
- ▷ Resizable!!
- ▷ Needs a <Type>
  - ▷ Can only hold one Type

3

### Constructor

▷ No parameter!
▷ `<Type>` goes after the word `ArrayList` in code
▷ Creates an empty list that holds the `<Type>` only
▷ Cannot create a list with a certain number of spots!!!!

4

### How Big?

▷ The `listName.size()` method tells how many items are in the list.
▷ This is an ordinal (counting) number aka 0,1,2,3
▷ The last index is always `listName.size() – 1`

5

### Add to the list

`.add(item)`                    `.add(index, item)`

▷ Adds the supplied item to the end of the list
▷ This means it will be placed at index of `listName.size()`

▷ Adds the specified item to the index
▷ All items previously at the index and above are incremented by one
▷ Index must be greater than or equal to zero **AND** less than or equal to `listName.size()`

6

### Retrieve from the list

▷ The method used to access individual items is `listName.get(index)`
  ▷ Retrieves the item stored at the specified index
  ▷ Does NOT affect the structure of the list
  ▷ Index must be between zero and less than the size of the list
  ▷ Quite possibly the most used method

7

### Replace Items

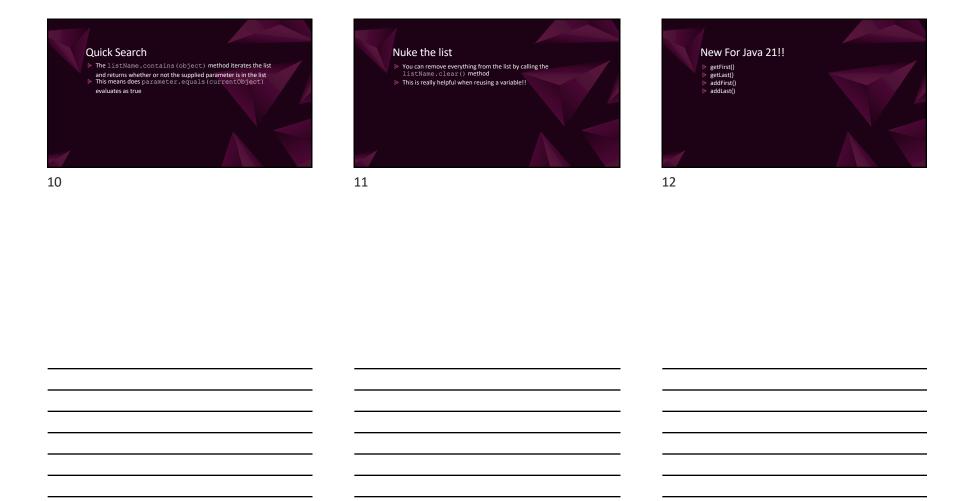▷ The method to replace items is `listName.set(index, item)`
▷ It returns the item stored at the location and places the parameter item at the specified index!
▷ This cannot be used to add to the list!!!!
  ▷ Maximum value for index parameter is size() - 1

8

### Remove Items

▷ The `listName.remove(index)` method removes the item at the specified index
▷ This makes the list smaller 😵
▷ All indices above the index are decremented by one
▷ The item that was previously at the index is returned
  ▷ You do not have to use it but it is often helpful

9

### Quick Search
▷ The `listName.contains(object)` method iterates the list and returns whether or not the supplied parameter is in the list
▷ This means does `parameter.equals(currentObject)` evaluates as true

10

### Nuke the list
▷ You can remove everything from the list by calling the `listName.clear()` method
▷ This is really helpful when reusing a variable!!

11

### New For Java 21!!
▷ getFirst()
▷ getLast()
▷ addFirst()
▷ addLast()

12

## Helpful Tips

▷ Indices start at zero and go to .size() – 1
▷ .size() gives the ordinal value not index!!
▷ Remember to specify the <Type>
▷ Indices less than zero always throw exceptions!!

13

_____
_____
_____
_____
_____
_____
_____