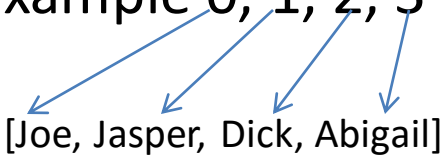
ArrayList class

An ArrayList can be visualized as a linear list of objects at index positions

- for example 0, 1, 2, 3



ArrayList is a data structure that grows and shrinks gracefully

- you can add/remove objects as necessary

ArrayList is a collection of objects

- we will focus on objects of type String
- to hold data of a primitive data type (e.g. int) you need to use a wrapper class (e.g. Integer, Double, Boolean, Character) where wrapper objects contain data of a primitive data type

ArrayList class

ArrayList methods

Method	Description plus examples using: `ArrayList<String> people = new Arraylist()`

`add(...)`	Can be used to either
	a) append a given element to the end of a list, or,
	b) if a position is specified insert the given element at the specified position (following elements are shifted *down*).
	`people.add("Jaime");`
	`people.add(4, "Jaime");`

`clear()`	Removes all elements from a list. `people.clear();`
`contains(...)`	Returns true if this list contains the specified element. `boolean found =` *people.contains("Jaime");*
`get(...)`	Returns the element at a specified position in this list. `String person = people.get(4);`
`indexOf(...)`	Returns the index of the first occurrence of the specified element in this list, or -1 if this list does not contain the element. `int pos = people.indexOf("Jaime");`

ArrayList class

ArrayList methods

`isEmpty()`	Returns true if the list has no elements. `boolean empty = people.isEmpty();`

`remove(...)`	Can be used to remove either
	a) the element at a specified position in this list, or
	b) the first element matching a given object;
	returns the deleted element and shifts other elements *up*.
	`String removed = people.remove(4);`
	`String removed = people.remove("Jaime");`

`set(...)`	Replaces an element with another element; returns the previous element. `String previous = people.set(4, "Jaime");`
`size()`	Returns the number of elements in this list. `int numElts = people.size();`

Focus on: add, contains, get, indexOf, set, size

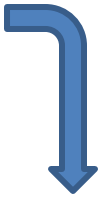
ArrayList class

Displaying a list

```
ArrayList < String > people =new ArrayList () ;
// add some names
people . add ("Joe") ;
people . add ("Jasper") ;
people . add ("Dick") ;
people . add ("Abigail") ;

System.out.print(people);
```

whole list is displayed



[Joe, Jasper, Dick, Abigail]

ArrayList class

The *enhanced* **for**

a variation on the **for**

used to iterate through all elements

cannot change anything

Example: iterate through an ArrayList

```
people =new ArrayList () ;
// add some names
people . add ("Joe") ;
people . add ("Jasper") ;
people . add ("Dick") ;
people . add ("Abigail") ;

for(String s : people) System.out.print(s+"   ");
```

simpler syntax than for

ArrayList class

Example

Consider the program DisplayReadme.java from previous chapter

Let's modify it

…… replace readme.txt with a Shakespeare work
google it … shakespeare mit

…… to put the tokens into a list

…… to put each token into list only once (list is now a vocabulary)

…… add another list to keep track of frequencies for each word
Which word appears most often… highest frequency