

CSIS 3103

Chapter 2: ArrayLists and Generics

The ArrayList Class

- Simplest class that implements the List interface
- Has an array to store elements but methods hide implementation details

A List of String objects:

```
List<String> myList = new ArrayList<String>();
myList.add("Bashful");
myList.add("Awful");
myList.add("Jumpy");
myList.add("Happy");
```

Generic Collections

- Generics allow defining a collection that contains references to objects of a specific type

```
List<String> myList =
    new ArrayList<String>();
```

specifies that *myList* is a List of String (String is a type parameter)

- Only references to objects of type String can be stored in *myList*, and all items retrieved would be of type String

```
myList.add(2, "Doc");
```

After insertion of "Doc" before the third element

```
myList.add("Dopey");
```

After insertion of "Dopey" at the end

```
myList.remove(1);
```

After removal of "Awful"

Example Application of ArrayList

```
ArrayList<Integer> someInts = new ArrayList<Integer>();
int[] nums = {5, 7, 2, 15};
for (int i = 0; i < nums.length; i++) {
    someInts.add(nums[i]);
}

// Display the sum
int sum = 0;
for (int i = 0; i < someInts.size(); i++) {
    sum += someInts.get(i);
}
System.out.println("sum is " + sum);
```