Array List

2190101: Computer Programming

Objectives

- Understand the difference between array and ArrayList
- Be able to choose between array and ArrayList
- Be able to create and use ArrayList
- Be able to apply ArrayList in real-world problems

Outlines

1. A limitation of array [ArrayExample.java]
2. Know why to use ArrayList [ArrayListExample.java]
3. Commands in ArrayList [ArrayListExample2.java]
4. ArrayList of Integer [ArrayListExample3.java]
5. ArrayList Appication [SchoolLottery.java]
6. Exercise in ArrayList Application

1. A limitation of array [ArrayExample.java]

```
public class ArrayExample {

	public static void main(String[] args) {
		// Let's work with a simple array
		String[] simpleArray = new String[3];
		simpleArray[0] = "Somchai";
		simpleArray[1] = "Pat";
		simpleArray[2] = "Sam";

	    // simpleArray[3] = "Somjai";
	}
}
```

2. Know why to use ArrayList [ArrayListExample.java]

```
import java.util.*;

public class ArrayListExample {

	public static void main(String[] args) {
		ArrayList<String> students = new ArrayList<String>(3);
		students.add("Somchai");
		students.add("Pat");
		students.add("Sam");
		students.add("Somjai");
		System.out.println(students.toString());
	}
}
```

3. Commands in ArrayList [ArrayListExample2.java]

- toString()
- add(index, object)
- get(index)
- set(index, new object)
- remove(object)
- remove(index)
- indexOf(object)

```
import java.util.*;
public class ArrayListExample2 {

	public static void main(String[] args) {
		ArrayList<String> students = new ArrayList<String>(3);
		students.add("Somchai");
		students.add("Pat");
		students.add("Sam");

		// via "toString" method
		System.out.println(students.toString() + "\n");

		// add more student
		System.out.println("Add more student.");
		students.add(1,"Somjai");
		System.out.println(students.toString() + "\n");

		// get method
		System.out.println("Test get method.");
		System.out.println("Student at index 1 is "+students.get(1) + ".\n");

		// set method
		System.out.println("Test set method.");
		students.set(0,"Chai");
		System.out.println(students.toString() + "\n");

		// remove(object) method
		System.out.println("Let's remove one student from the ArrayList.");
		students.remove("Sam");
		System.out.println(students.toString() + "\n");

		// remove(index) method
		System.out.println("Remove more student.");
		students.remove(1);
		System.out.println(students.toString() + "\n");

		// indexOf method
		System.out.println(students.indexOf("Chai"));
		System.out.println(students.indexOf("Sam"));
	}
}
```

4. ArrayList of Integer [ArrayListExample3.java]

There are three ways to initialize ArrayList

- ArrayList() builds an empty arraylist
- ArrayList(Collection c) builds an array list that is initialized with the elements of the collection c
- ArrayList(int capacity) builds an array list with a specified capacity

```
import java.util.*;
public class ArrayListExample3 {

	public static void main(String[] args) {

		ArrayList<Integer> numbers = new ArrayList<Integer>(3);
		numbers.add(2);
		numbers.add(3);
		numbers.add(4);

		// via "toString" method
		System.out.println(numbers.toString() + "\n");

		// add more student
		System.out.println("Add more student.");
		numbers.add(1,99);
		System.out.println(numbers.toString() + "\n");

		// get method
		System.out.println("Test get method.");
		System.out.println("Number at index 1 is " + numbers.get(1) + ".\n");

		// set method
		System.out.println("Test set method.");
		numbers.set(0,999);
		System.out.println(numbers.toString() + "\n");

		// remove(index) method
		System.out.println("Let's remove one number from the ArrayList.");
		numbers.remove(3);
		System.out.println(numbers.toString());
	}
}
```

5. ArrayList Appication [SchoolLottery.java]

In this program, user can enter a list of gamblers from keyboard, which can be duplicated. The program stops getting the list when an empty name (“”) is entered. Then, it will draw the lucky one ☺.

```
import java.util.*;
public class SchoolLottery{
	private ArrayList<String> entries;	// holds Student references
	public SchoolLottery(){
		entries = new ArrayList<String>();
	}

	public void addStudents(){
		// prompts for student names
		// adds students to entries list
		// allow duplicate entries
		Scanner input = new Scanner(System.in);
		int studentNum = 0;
		System.out.println("Please Enter to end input");
		System.out.print("Name" + ++studentNum + ": ");
		String name = input.nextLine();
		while (!name.equals("")){		// signals end of data
			entries.add(name);
			System.out.println(name + " entered in the lottery.");
			System.out.print("\nName" + ++studentNum + ": ");
			name = input.nextLine();
		}
		pickWinner();
	}

	public void pickWinner(){
		// chooses a random entry and displays winners name
		int numEntries = entries.size(); // size of ArrayList
		if(numEntries == 0)
			System.out.println("*** No participants ***");
		else{
			Random random = new Random();
			String winner = entries.get(random.nextInt(numEntries));
			System.out.print("\n*** The winner is " + winner + " ***");
		}
	}

	public static void main(String[] args){
		SchoolLottery lottery = new SchoolLottery();
		lottery.addStudents();
	}
}
```

6. Exercise in ArrayList Application

Update program from the above example (Application in ArrayList) to obtain the list from file instead of keyboard. Also, a name in the list cannot be duplicated.

Tip! Compare to Assignment 9-2, do you still need to find out the number of lines in the file if you use ArrayList?