Take Assessment: Practical Quiz 6

Please answer the following question(s).
If the assessment includes multiple-choice questions, click the "Submit Answers" button when you have completed those questions.

You have 120 minutes to take this assessment.
Please complete this assessment by Thu Dec 20 2007 22:11:52 GMT+0800.

1. Go to bottom of question.

Using Arrays

Prerequisites, Goals, and Outcomes

Prerequisites: Before you begin this exercise, you need mastery of the following:

- *Arrays*
    - Knowledge of arrays
        - Declaring arrays
        - Initializing arrays
        - Accessing array elements
        - Using for-loops to process arrays

Goals: Reinforce your ability to use arrays

Outcomes: You will demonstrate mastery of the following:

- Using arrays
    - Creating arrays
        - From a set of objects
        - From another array
    - Processing arrays
        - Finding an array element with a specific characteristic
        - Counting the number of array elements with a specific characteristic
        - Invoking a method on each array element
        - Finding the string representation of an array

Background

This assignment asks you to implement a set of methods that handle arrays.

Description

In this assignment, you will finish the implementation of EmployeeArray, a class with static methods for creating, analyzing, and manipulating arrays of Employee objects. iCarnegie provides a test driver and the class Employee.

Class Employee

A complete implementation of this class is included in the student archive *student-files.zip*. Stop *now* and review its documentation:

- *Employee.html*. Documentation for class Employee

Class EmployeeArray

A partial implementation of this class is included in the student archive *student-files.zip*. You should complete the implementation of every method in this class.

Class TestEmployeeArray

This class is a test driver for class EmployeeArray. It contains test cases for every method in the class. A complete implementation is included in the student archive *student-files.zip*. You should use this class to test your implementation of EmployeeArray.

Files

The following files are needed to complete this assignment:

- *student-files.zip*. Download this file. This archive contains the following:
    - *Employee.java*. A complete implementation
    - *EmployeeArray.java*. Use this template to complete your implementation.

- *TestEmployeeArray.java*. A complete implementation

Tasks

Implement all methods in class EmployeeArray. Follow Sun's code conventions. The following steps will guide you through this assignment. Work incrementally and test each increment. Save often.

1. **Extract** the files by issuing the following command at the command prompt:

   ```
   C:\>unzip student-files.zip
   ```

2. **Test** each method as soon as you finish writing it by issuing the following command at the command prompt:

   ```
   C:\>java TestEmployeeArray
   ```

3. **Implement** the method makeArray.

- *public static Employee[] makeArray(Employee first, Employee second, Employee third)*. This method takes three Employee objects and returns an Employee array with three elements. The first element of the array contains a reference to the first argument; the second element contains a reference to the second argument; and the third element contains a reference to the third argument.

For example, consider the following objects:

```
Employee[102,Mary Jones,60000.0]
Employee[101,Joe Smith,20000.0]
Employee[103,Richard Wong,40000.0]
```

If these objects are passed to makeArray in the indicated order, makeArray will return the following array:

```
{Employee[102,Mary Jones,60000.0],
 Employee[101,Joe Smith,20000.0],
 Employee[103,Richard Wong,40000.0]}
```

Note: Employee[*ID*,*name*,*salary*] is the string representation of an Employee object.

4. **Implement** the method copyArray. Use indexes to implement this method.

- *public static Employee[] copyArray(Employee[] array)*. This method takes an Employee array and returns a *new* array with the same elements in the same order.

For example, consider the following array:

```
{Employee[102,Mary Jones,60000.0],
 Employee[101,Joe Smith,20000.0],
 Employee[103,Richard Wong,40000.0]}
```

If copyArray is passed this array, it will return the following array:

```
{Employee[102,Mary Jones,60000.0],
 Employee[101,Joe Smith,20000.0],
 Employee[103,Richard Wong,40000.0]}
```

5. **Implement** the method getEmployee. Use a for-each loop to implement this method.

- *public static Employee getEmployee(Employee[] array, int id)*. This method takes two arguments, an Employee array and an employee ID, and returns the Employee object with the specified ID. This method returns null if there are no employees in the specified array with the specified ID.

For example, consider the following array:

```
{Employee[102,Mary Jones,60000.0],
 Employee[101,Joe Smith,20000.0],
 Employee[103,Richard Wong,40000.0]}
```

If getEmployee is passed this array and the integer 103, it will return the Employee object for Richard Wong. If getEmployee is passed this array and the integer 222, it will return null.

6. **Implement** the method countHigherSalaries. Use a for-each loop to implement this method.

- *public static int countHigherSalaries(Employee[] array, double amount)*. This method takes two arguments, an Employee array and a dollar amount, and returns the number of employees in the specified array whose salary is higher than the specified dollar amount.

For example, consider the following array:

```
{Employee[102,Mary Jones,60000.0],
 Employee[101,Joe Smith,20000.0],
 Employee[103,Richard Wong,40000.0]}
```

If countHigherSalaries is passed this array and the double 50000.0, it will return 1. If countHigherSalaries is passed this array and the double 20000.0, it will return 2.

7. **Implement** the method sumSalaries. Use a for-each loop to implement this method.

- *public static double sumSalaries(Employee[] array)*. This method takes an Employee array and returns the sum of the salaries of the employees in the specified array.

For example, consider the following array:

```
{Employee[102,Mary Jones,60000.0],
 Employee[101,Joe Smith,20000.0],
 Employee[103,Richard Wong,40000.0]}
```

If sumSalaries is passed this array, it will return 120000.0.

8. **Implement** the method getHighestSalary. Use indexes to implement this method.

- *public static double getHighestSalary(Employee[] array)*. This method takes an Employee array and returns the highest salary in the specified array. To simplify your code, you can assume that the specified array contains one or more elements.

For example, consider the following array:

```
{Employee[102,Mary Jones,60000.0],
 Employee[101,Joe Smith,20000.0],
 Employee[103,Richard Wong,40000.0]}
```

If this array is passed to getHighestSalary, it will return 60000.0.

9. **Implement** the method increaseAll. Use a for-each loop to

implement this method.

- *public static void increaseAll(Employee[] array, double amount)*. This method takes two arguments, an Employee array and a dollar amount, and increases the salary of every employee in the specified array by the specified amount.

For example, consider the following array:

```
{Employee[102,Mary Jones,60000.0],
 Employee[101,Joe Smith,20000.0],
 Employee[103,Richard Wong,40000.0]}
```

If increaseAll is passed this array and the integer 1000, the array will be modified as follows:

```
{Employee[102,Mary Jones,61000.0],
 Employee[101,Joe Smith,21000.0],
 Employee[103,Richard Wong,41000.0]}
```

10. **Implement** the method displayAll. Use indexes to implement this method.

- *public static String displayAll(Employee[] array)*. This method takes an Employee array and returns a string representation of the specified array. To simplify your code, you can assume that every element in the specified array contains a valid reference to an Employee object.

Use the method toString in the class Employee to obtain the string representation of each object in the array. A new line character (\n) should separate the string representations of each Employee object.

For example, consider the following array:

```
{Employee[102,Mary Jones,61000.0],
 Employee[103,Richard Wong,41000.0]}
```

If this array is passed to displayAll, the following string will be returned:

"Employee[102,Mary Jones,61000.0]\nEmployee[103,Richard Wong,41000.0]"

Note that the string does *not* end with a new line character (\n).

Submission

Upon completion, submit **only** the following.

1. EmployeeArray.java

Go to top of question.

File to submit: Upload File Forward File Refresh Ready for Grading

Go to top of assessment.