Programming and Logic I
Project 3 - RunEmployee
Due: May 2, 2021

Files to Submit:

<yourname>RunEmployee.java
<yourname>Employee.java

Currently, you have a RunEmployee class that reads employee data from a file, creates an Employee object, and writes formatted output to standard output. In this project, you will create an additional field in your Employee class along with an accessor and mutator for the field and a method to display the Employee data. In RunEmployee, you will create an array of Employee objects, and loop through the array to display the employee data.

Employee class

Add a data field to your employee class called `email`. This field should be a String.

Methods

- Create an **accessor** and **mutator** method for the new email field

- Add a display() method that **takes no parameters and does not return a value**. This method should display the contents of an Employee object. (Code that displays employee data is currently in RunEmployee within the while loop. Remove that code from RunEmployee and add it to this method. It will be very similar with perhaps a few tweaks. The output can look the same. Since the code for determining the employee's position is also within the loop, that will be moved to the display method as well.)

RunEmployee class

Create an array of type Employee. We need to give the array a size. Although we may not know the number of employees in the file, we are going to fudge it a little here and give the array a size of 12. There are 12 employees in the file. Normally, we might not know this.

This class currently opens a file for reading, reads the company name, and then iterates through the file reading each employee's data and writing employee data to standard output. The loop ends when there is no more data in the file. It currently creates the Employee object and calls a 5-argument constructor. After you create the Employee

object, add the employee's email address to the object by calling the email mutator (set) method you created in the Employee class.

After the Employee object has been created and updated with the email address, add the Employee object to the array. You will need an index to add the Employee object to the array. To do this, create an int variable that is set to 0 before the loop begins and incremented within the loop.

Remove all the output statements. The Employee object now has a display method that will be called after the loop completes.

After the loop completes, you will have an array of type Employee that will be filled with 12 Employee objects. Create a for loop that iterates 12 times. Inside this loop, use the array variable and loop control variable (as index to the Employee array) to call each Employee object's display method. The output can look the same, the difference is the Employee object is calling its own method to display its data.

Notes:

- You are creating ONE array. The type of the array is Employee.
- You are NOT creating an ArrayList
- The main method is still printing the name of the company.

Sample Run:

```
> run ConwayRunEmployee3

Hudson Valley Corporation
-----------------------------------------
Employee:       Jerry Seinfeld
Position:       Associate
Salary:         $100,000.00
Email Address:  j-seinfeld@hv.com
Bonus:          false
-----------------------------------------
Employee:       Elaine Benes
Position:       Associate
Salary:         $100,000.00
Email Address:  e-benes@hv.com
Bonus:          false
-----------------------------------------
Employee:       George Costanza
Position:       Assistant
Salary:         $50,000.00
Email Address:  g-costanza@hv.com
Bonus:          false
-----------------------------------------
```

```
Employee:       Cosmo Kramer
Position:       Assistant
Salary:         $50,000.00
Email Address:  c-kramer@hv.com
Bonus:          true
-----------------------------------------
Employee:       Jacopo Peterman
Position:       Executive
Salary:         $150,000.00
Email Address:  j-peterman@hv.com
Bonus:          true
-----------------------------------------
Employee:       David Putty
Position:       Assistant
Salary:         $50,000.00
Email Address:  d-putty@hv.com
Bonus:          true
-----------------------------------------
Employee:       Justin Pitt
Position:       Executive
Salary:         $200,000.00
Email Address:  j-pitt@hv.com
Bonus:          true
-----------------------------------------
```

```
Employee:       Russell Dalrymple
Position:       Executive
Salary:         $150,000.00
Email Address:  r-dalrymple@hv.com
Bonus:          false
-----------------------------------------
Employee:       Tim Whatley
Position:       Intern
Salary:         $25,000.00
Email Address:  t-whatley@hv.com
Bonus:          false
-----------------------------------------
Employee:       Joe Davola
Position:       Intern
Salary:         $25,000.00
Email Address:  j-davola@hv.com
Bonus:          false
-----------------------------------------
Employee:       Kenny Bania
Position:       Assistant
Salary:         $50,000.00
Email Address:  k-bania@hv.com
Bonus:          true
-----------------------------------------
Employee:       Susan Ross
Position:       Executive
Salary:         $150,000.00
Email Address:  s-ross@hv.com
Bonus:          true
```