Periodically, a company processes the retirement of some of its employees. In this question, you will write methods to help the company determine whether an employee is eligible to retire and to process the retirement of all eligible employees.

The Employee class is declared as follows.

public class Employee

{//instance variables, constructors, and other methods are not shown.

/\*\* @return the age (in years) of this employee

\*/

public int getAge()

{//implementation not shown}

/\*\* @return the number of years this employee has worked

\*/

public int getYears()

{//implementation not shown}

/\*\* @return the salary of this employee in dollars

\*/

public double getSalary()

{//implementation not shown}

/\*\* @return the unique employee ID number of this employee

\*/

public int getID()

{//implementation not shown}

}

The Company class is declared as follows.

public class Company

{

private Employee[] empList;

// sorted in ascending order by employee ID

private int retireAge; //minimum age to retire

private int retireYears; //minimum years on job to retire

private double retireSalary; //minimum salary to retire

private double salaryBudget; //total salary of all employees

/\*\* @param emp is an Employee object

\* @return true if emp is eligible to retire; false otherwise

\*/

private boolean employeeIsEligible(Employee emp)

{/\* to be implemented in part (a) \*/ }

/\*\***Postcondition**: all retirement-eligible employees have been removed from empList; empList

\*has been resized to reflect retirements; empList remains sorted by employee ID; salaryBudget

\*has been updated to reflect remaining employees.

\*/

public void processRetirements()

{/\* to be implemented in part (b) \*/ }

(a) An employee is eligible for retirement if (s)he meets at least two of the following requirements:

1. The employee is at least retireAge years old.
2. The employee has worked for at least retireYears.
3. The employee’s salary is at least retireSalary.

Write the Company method employeeIsEligible. Method employeeIsEligible returns a boolean value that indicates whether Employee emp is eligible for retirement using the rules described above. Complete method employeeIsEligible below.

/\*\* @param emp is an Employee object

\* @return true if emp is eligible to retire; false otherwise

\*/

private boolean employeeIsEligible(Employee emp)

(b) Write the Company method processRetirements. Method processRetirements removes all retirement-eligible employees from the empList array, resizes (shrinks) empList as appropriate (maintaining its order by employee ID), and decreases salaryBudget to reflect the salary of the remaining employees. In writing processRetirements, you may call employeeIsEligible, specified in part (a). Assume that employeeIsEligible works as specified, regardless of what you wrote in part (a). Complete method processRetirements below.

/\*\***Postcondition**: all retirement-eligible employees have been removed from empList; empList

\*has been resized to reflect retirements; empList remains sorted by employee ID; salaryBudget

\*has been updated to reflect remaining employees.

\*/

public void processRetirements()