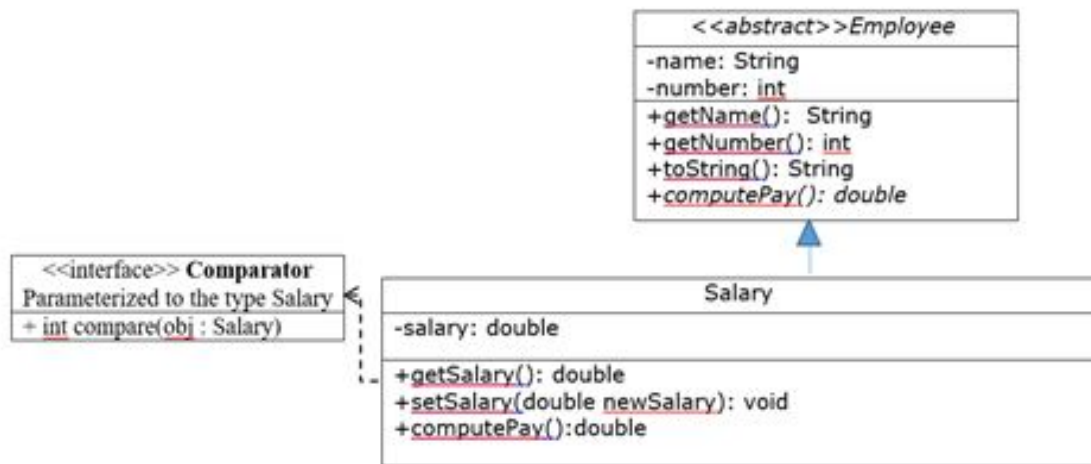


1. Create a Student class with 2 fields *Name*, *cgpa*. Create an arraylist of *n* students, write a comparator that sorts them in the order of decreasing *cgpa*. When 2 or more students have the same *cgpa*, sort those students alphabetically by name. Write a class that implements the comparable interface for comparing two students. **Note:** Two students can have the same name.
2. Consider the Employee class diagram shown below.



- a. Employee Class: It is an abstract class with name and number as private member variables. The constructor of this class assigns the name and number of the employee. It has getter methods for name and number. It overrides the `toString()` to print the name and number of the employee object. It has an abstract method `computePay()`.
 - b. Salary Class: This class inherits the Employee class and has salary as an additional private member. The constructor of this class sets the salary of an employee. The constructor sets the private members of the super class as well. The abstract method 'computePay' is overridden to compute the weekly pay of an Employee when the annual income is given as the input.
 - c. Test class: A linked list of salary type is created. The employee records are added to the list. An anonymous instance of type Comparator is created and it overrides the `compare()` method to sort employee list in descending order according to the employee's salary.
3. Write a Generic method "sort" that takes a generic array as an input and returns the sorted array. From the main method pass an Integer array, String array and Double array to test the generic method.
 4. Read the file 'Text.txt' which contains data about a specific topic. Split the string read from the file and count the frequency of occurrence of all the unique words in it. Sort the word frequency pair and exclude if the words are "a", "an", "the", "is", "was", "it". Find the top three keywords describing the text.

5. Update the Employee class given in question no. 2 to include a reference to the Department class. Let the Dname and Dno be the fields of the Department class. Display the list of each department with the details of the employee getting the highest salary and the total salary of all the employee in the department.