```
/*
import java.io.*;
import java.util.*;*/

abstract class UCHealthEmployee {
  String name;

  UCHealthEmployee() {}
  UCHealthEmployee (String nm) { name = nm; }
  abstract double computePay();
  void display () {}
  void setHours(double hrs) {}
  void setPatients(double patients) {}
  void setSalary(double salary) { System.out.println("NO!"); }
}

class HourlyEmployee extends UCHealthEmployee {
  double rate;
  double hours;

  HourlyEmployee(String nm) { super(nm); }
  HourlyEmployee(String nm, double r) { super(nm);
                                rate = r; }
  void setRate(double r)    { rate  = r; }
  void setHours(double hrs) { hours = hrs; }
  double computePay()       { return rate*hours; }
}

class Nurse extends HourlyEmployee {
  Nurse (String nm, double w) { super(nm, w); }
  void display() {
    System.out.println("Name: "+name+"\tHours: "+hours +"\tRate: "+rate);
  }
}

class PhysicianAssistant extends HourlyEmployee {
  double perPatientRate;
  double PatientsSeen;

  PhysicianAssistant (String nm, double p) {
    super(nm);
    perPatientRate = p;
  }
```

```
  void setPatientRate(double ptrate) { perPatientRate = ptrate; }

  void setPatients(double patients) { PatientsSeen = patients; }

  double computePay() { return perPatientRate*PatientsSeen; }

  void display() {
    System.out.println("Name: "+name+"\tperPatientRate: "+perPatientRate +"\tPatientSeen:
"+PatientsSeen);
  }
}

class Manager extends UCHealthEmployee {
  double monthlysalary;

  Manager () { super(""); }

  Manager(String nm, double w)  { super(nm);
                          monthlysalary = w; }

  void setSalary(double salary) { monthlysalary = salary; }

  double computePay()          { return monthlysalary; }

  void display() {
    System.out.println("Name: "+name+"\tMonthly Salary: "+ monthlysalary);
  }
}

interface ManagerInterface {
  double managerComputePay();
  void managerDisplay();
}

class PhysicianAssistantManager extends PhysicianAssistant implements ManagerInterface {
  double monthlysalary;

  PhysicianAssistantManager(String nm, double w) { super (nm, w); }

  public double managerComputePay() {
    return monthlysalary;
  }
```

```
/*Pay is computed by adding the result of computingPay as though the employee is of type
PhysicianAssistant, to the result of computingPay as though the employee is of type Manager.*/
  double computePay() {
    System.out.println("PhysicianAssistantManager: " + name + " " +
super.computePay()+managerComputePay());
    return super.computePay() + managerComputePay();
  }

  void setSalary (double s) { monthlysalary = s; }

  public void managerDisplay() {
    System.out.println("Name: "+name+"\tMonthly Salary: "+monthlysalary);
  }

  void display() {
    super.display();
    managerDisplay();
  }
}

//////////////UCHealthEmployeeSummaryClass//////////////////////////

import java.util.ArrayList;

class UCHealthEmployeeSummary{

        private ArrayList<UCHealthEmployee> employeeList = new ArrayList<>();

        UCHealthEmployee find(String nm){
                Integer count = 0;

                while (count < employeeList.size()){
                        UCHealthEmployee curr = employeeList.get(count);

                        if(curr.name == nm){
                                return curr;
                        }
                        count++;
                }
                return null;
        }
```

```
	void setHours(String nm, double hrs){
		find(nm).setHours(hrs);
	}

	void setSalary(String nm, double salary){
		find(nm).setSalary(salary);
	}

	void setPatients(String nm, double patients){
		find(nm).setPatients(patients);
	}

	double payroll(){
		Double payroll = (double) 0;
		Integer count = 0;

		while (count < employeeList.size()){
			UCHealthEmployee curr = employeeList.get(count);
			payroll = payroll + curr.computePay();
			count++;
		}

		return payroll;
	}

	void display(){
		Integer count = 0;

		while (count < employeeList.size()){
			UCHealthEmployee curr = employeeList.get(count);
			curr.display();
			count++;
		}
	}

	public void enqueue(UCHealthEmployee  new_employee) {
		employeeList.add(new_employee);
	}

}

//////////////////////////////////////////////////////////////
```

```
public class UCHealthSalary {
  public static void main(String argv[]) {
    /***
    UCHealthEmployeeSummary emp = new UCHealthEmployeeSummary();
    emp.enqueue(new PhysicianAssistantManager("Laura", 1000));
    emp.enqueue(new PhysicianAssistantManager("Edward", 1000));
    emp.enqueue(new PhysicianAssistantManager("Sarah", 1000));
    emp.enqueue(new PhysicianAssistant("Tim", 3.5));
    emp.enqueue(new PhysicianAssistant("Joan", 4.5));
    emp.enqueue(new PhysicianAssistant("Amy", 2.5));
    emp.enqueue(new Manager("Kim", 10000));
    emp.enqueue(new Manager("Frank", 5000));
    emp.enqueue(new Manager("Beth", 3000));
    emp.enqueue(new Nurse("Kris", 17));
    emp.enqueue(new Nurse("Amber", 15));
    emp.enqueue(new Nurse("Lewis", 13));

    emp.setPatients("Laura", 400);
    emp.setPatients("Edward", 300);
    emp.setPatients("Sarah", 200);
    emp.setSalary("Laura", 1000);
    emp.setSalary("Edward", 2000);
    emp.setSalary("Sarah", 3000);
    emp.setPatients("Tim", 120);
    emp.setPatients("Joan", 100);
    emp.setPatients("Amy", 50);
    emp.setHours("Kris", 35);
    emp.setHours("Amber", 23);
    emp.setHours("Lewis", 3);


    emp.display();
    System.out.println("Payroll: "+emp.payroll());
    ***/
  }
}

//////////////////////////////////////////////OTHER VERSION/////////////////////////////////////////
import java.util.ArrayList;
import java.util.Iterator;


public final class UCHealth {
```

```
abstract class UCHealthEmployee {
        String name;

        UCHealthEmployee() {}
        UCHealthEmployee (String nm) { name = nm; }
        abstract double computePay();
        void display () {}
        void setHours(double hrs) {}
        void setPatients(double patients) {}
        void setSalary(double salary) { System.out.println("NO!"); }
      }

      class HourlyEmployee extends UCHealthEmployee {
        double rate;
        double hours;

        HourlyEmployee(String nm) { super(nm); }
        HourlyEmployee(String nm, double r) { super(nm);
                                    rate = r; }
        void setRate(double r)    { rate  = r; }
        void setHours(double hrs) { hours = hrs; }
        double computePay()       { return rate*hours; }
      }

      class Nurse extends HourlyEmployee {
        Nurse (String nm, double w) { super(nm, w); }
        void display() {
          System.out.println("Name: "+name+"\tHours: "+hours +"\tRate: "+rate);
        }
      }

      class PhysicianAssistant extends HourlyEmployee {
        double perPatientRate;
        double PatientsSeen;

        PhysicianAssistant (String nm, double p) {
          super(nm);
          perPatientRate = p;
        }

        void setPatientRate(double ptrate) { perPatientRate = ptrate; }

        void setPatients(double patients) { PatientsSeen = patients; }
```

```
    double computePay() { return perPatientRate*PatientsSeen; }

    void display() {
      System.out.println("Name: "+name+"\tperPatientRate: "+perPatientRate
+"\tPatientSeen: "+PatientsSeen);
    }
  }

  class Manager extends UCHealthEmployee {
    double monthlysalary;

    Manager () { super(""); }

    Manager(String nm, double w)  { super(nm);
                                    monthlysalary = w; }

    void setSalary(double salary) { monthlysalary = salary; }

    double computePay()           { return monthlysalary; }

    void display() {
      System.out.println("Name: "+name+"\tMonthly Salary: "+ monthlysalary);
    }
  }

  interface ManagerInterface {
    double managerComputePay();
    void managerDisplay();
  }

  class PhysicianAssistantManager extends PhysicianAssistant implements
ManagerInterface {
    double monthlysalary;

    PhysicianAssistantManager(String nm, double w) { super (nm, w); }

    public double managerComputePay() {
      return monthlysalary;
    }
```

```
                /*Pay is computed by adding the result of computingPay as though the employee
is of type PhysicianAssistant, to the result of computingPay as though the employee is of type
Manager.*/
                  double computePay() {
                    System.out.println("PhysicianAssistantManager: " + name + " " +
super.computePay()+managerComputePay());
                    return super.computePay() + managerComputePay();
                  }

                  void setSalary (double s) { monthlysalary = s; }

                  public void managerDisplay() {
                    System.out.println("Name: "+name+"\tMonthly Salary: "+monthlysalary);
                  }

                  void display() {
                    super.display();
                    managerDisplay();
                  }
                }

                ///////////////UCHealthEmployeeSummary Class//////////////////////////

import java.util.ArrayList;

class UCHealthEmployeeSummary{

        private ArrayList<UCHealthEmployee> employeeList = new ArrayList<>();

        UCHealthEmployee find(String nm){
                Integer count = 0;

                while (count < employeeList.size()){
                        UCHealthEmployee curr = employeeList.get(count);

                        if(curr.name == nm){
                                return curr;
                        }
                        count++;
                }
                return null;
        }
```

```
	void setHours(String nm, double hrs){
		find(nm).setHours(hrs);
	}

	void setSalary(String nm, double salary){
		find(nm).setSalary(salary);
	}

	void setPatients(String nm, double patients){
		find(nm).setPatients(patients);
	}

	double payroll(){
		Double payroll = (double) 0;
		Integer count = 0;

		while (count < employeeList.size()){
			UCHealthEmployee curr = employeeList.get(count);
			payroll = payroll + curr.computePay();
			count++;
		}

		return payroll;
	}

	void display(){
		Integer count = 0;

		while (count < employeeList.size()){
			UCHealthEmployee curr = employeeList.get(count);
			curr.display();
			count++;
		}
	}

	public void enqueue(UCHealthEmployee  new_employee) {
		employeeList.add(new_employee);
	}

}

		//////////////////////////////////////////////////////////////
```

```
		public static void main(String argv[]) {
			UCHealth this_be_outter = new UCHealth();
			UCHealthEmployeeSummary emp = this_be_outter.new
UCHealthEmployeeSummary();
			emp.enqueue(this_be_outter.new PhysicianAssistantManager("Laura",
1000));
			emp.enqueue(this_be_outter.new PhysicianAssistantManager("Edward",
1000));
			emp.enqueue(this_be_outter.new PhysicianAssistantManager("Sarah",
1000));
			emp.enqueue(this_be_outter.new PhysicianAssistant("Tim", 3.5));
			emp.enqueue(this_be_outter.new PhysicianAssistant("Joan", 4.5));
			emp.enqueue(this_be_outter.new PhysicianAssistant("Amy", 2.5));
			emp.enqueue(this_be_outter.new Manager("Kim", 10000));
			emp.enqueue(this_be_outter.new Manager("Frank", 5000));
			emp.enqueue(this_be_outter.new Manager("Beth", 3000));
			emp.enqueue(this_be_outter.new Nurse("Kris", 17));
			emp.enqueue(this_be_outter.new Nurse("Amber", 15));
			emp.enqueue(this_be_outter.new Nurse("Lewis", 13));

			emp.setPatients("Laura", 400);
			emp.setPatients("Edward", 300);
			emp.setPatients("Sarah", 200);
			emp.setSalary("Laura", 1000);
			emp.setSalary("Edward", 2000);
			emp.setSalary("Sarah", 3000);
			emp.setPatients("Tim", 120);
			emp.setPatients("Joan", 100);
			emp.setPatients("Amy", 50);
			emp.setHours("Kris", 35);
			emp.setHours("Amber", 23);
			emp.setHours("Lewis", 3);


			emp.display();
			System.out.println("Payroll: "+emp.payroll());
		}

}
```