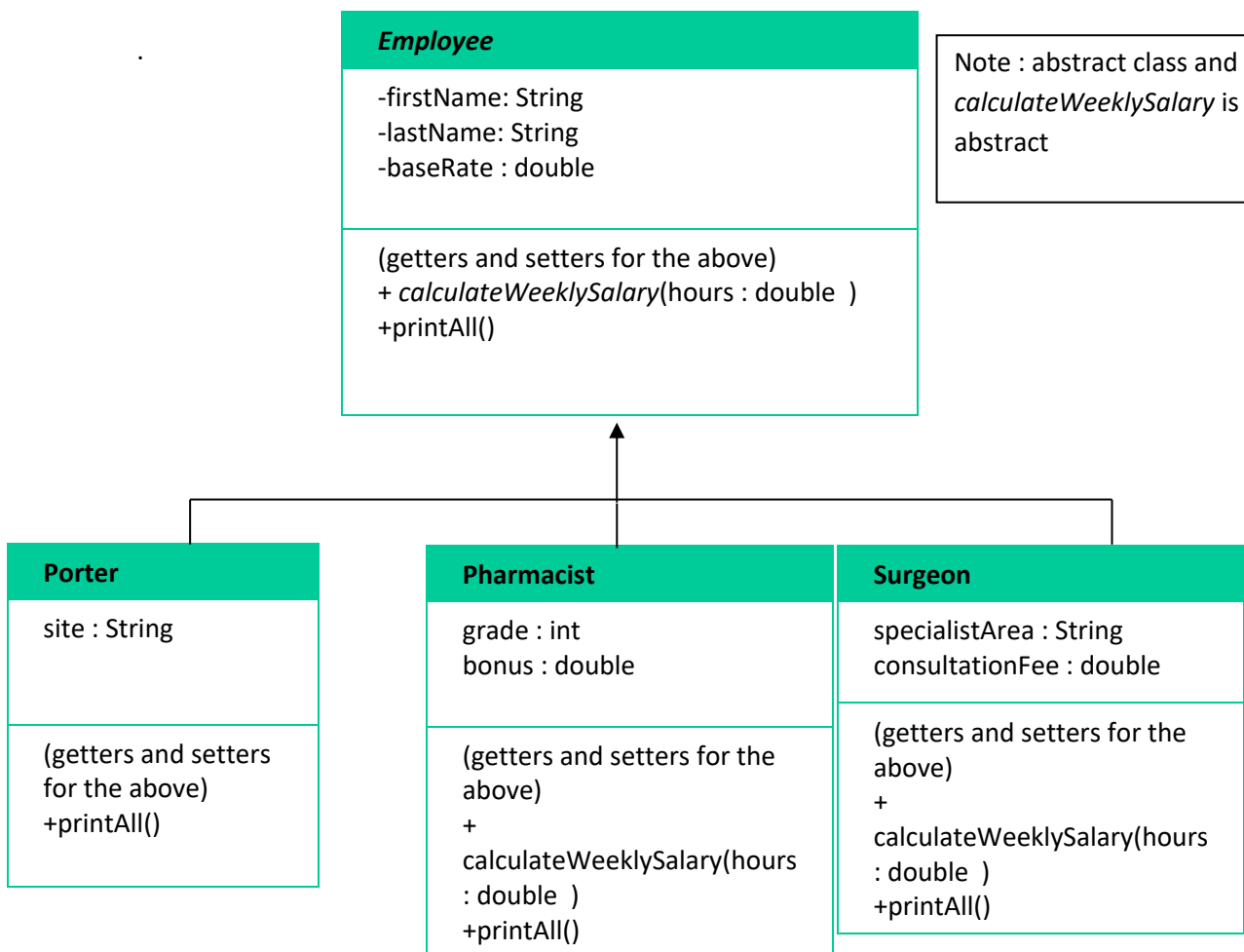


Hospital Payroll system

1. Design

Create the following classes to represent a Hospital payroll system.



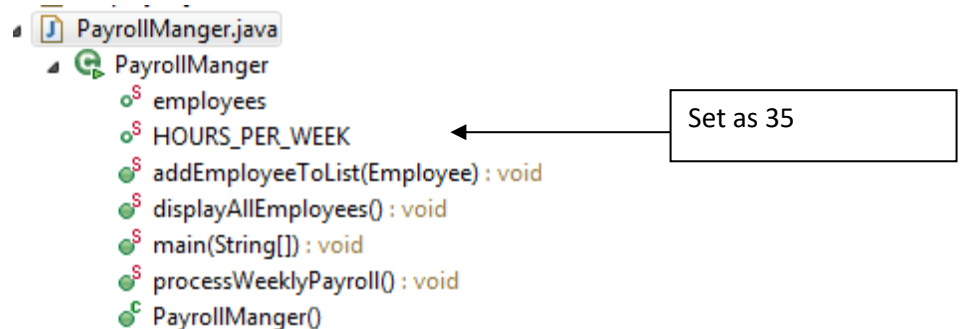
The implementation of the *calculateWeeklySalary* method for each employee type should be as defined below.

| Employee type | Weekly Salary |
|---------------|--|
| Porter | Hours worked per week * base rate |
| Pharmacist | Hours worked per week * base rate + bonus |
| Surgeon | Hours worked per week * base rate + Consultation Fee |

(Note : All employees work 35 hours in a week.)

2. Implementation

Create a class **PayrollManger.java** with the following constant (HOURS_PER_WEEK), static variable (employees) and methods. Note all methods are declared static.



Within **PayrollManger.java** (main method) firstly create and populate the list of employees as an Array of type Employee – limited to **six** employees.

```
// Static variable to hold employees
public static Employee[] employees = new Employee[6];
```

You should use **polymorphic processing** to manage the pay run processing (**processWeeklyPayroll()**) and display all functionality for the employees (**displayAllEmployees()**). Use the test data listed in section 4.

For example when creating the employee types use something like this..

```
Employee porter1 = new Porter("Ivana", "Patient", 10.50, "Royal");
```

When adding an employee to the Array of employees ensure the array has capacity and has free space for it. To do this you should check if the proposed index location in the array is null. If so add it can be added otherwise you can't add at this location.

```
for (int loop=0; loop<employees.length; loop++){
    if (employees[loop] == null){
        // add the employee to the empty array index
    }
}
```

The *processWeeklyPayroll()* should be implemented something like this ...

```
for (Employee e: employees){  
    // check if the array index is not null  
    if (e!=null){  
        e.calculateWeeklySalary(HOURS_PER_WEEK);
```

The displayAllEmployees() method should be implemented something like this ...

```
for (Employee e: employees){  
    // check if the array index is not null  
    if (e!=null){  
        e.printAll();
```

3. Test data :

| Employee type | First Name | Last Name | Base rate | Attributes Specific to employee type |
|---------------|------------|-----------|-----------|---|
| Porter | Ivana | Patient | £10.50 | Site: Royal |
| Porter | Amanda | Pushabed | £10.50 | Site: BCH |
| Surgeon | Jack | Ripper | £55.25 | Specialist Area : Renal Consultation Fee :£650.00 |
| Surgeon | Edward | Lister | £55.25 | Specialist Area : Vascular Consultation Fee :£800.00 |
| Pharmacist | Poppy | Pill | £30.50 | Grade: 7 Bonus : £750.00 |

4. Program output

The output of PayrollManager.java should be...

```
Display all employees
[Type      ] First name      Last name      Rate   Other
[Porter    ] Ivana          Patient        £10.50 Site : Royal
[Porter    ] Amanda         Pushabed       £10.50 Site : BCH
[Surgeon   ] Jack           Ripper         £55.25 Specialist Area : Renal, Consultation Fee :£650.00
[Surgeon   ] Edward         Lister         £55.25 Specialist Area : Vascular, Consultation Fee :£800.00
[Pharmacist] Poppy          Pill           £30.50 Grade : 7, Bonus : £750.00
Total employees in system : 5

Payroll run.....
Ivana      Patient    [Porter      ] : 35.00hrs * £10.50 = £367.50
Amanda     Pushabed   [Porter      ] : 35.00hrs * £10.50 = £367.50
Jack       Ripper     [Surgeon     ] : 35.00hrs * £55.25 + £650.00 = £2583.75
Edward     Lister     [Surgeon     ] : 35.00hrs * £55.25 + £800.00 = £2733.75
Poppy      Pill       [Pharacist   ] : 35.00hrs * £30.50 + £750.00 = £1817.50
Total records processed : 5
```