**Trang bìa**

**Lập trình hướng đối tượng**

**Họ và Tên: Huỳnh Lê Minh Duy**

**STT: 14**

**MSSV: 22662651**

**Lớp học phần: DHKTPM18A**

**BÀI TẬP TUẦN 06-07**

**Bài Làm**

**Nội dung**

**Kế thừa và đa hình**

**Yêu cầu:** Tạo một project có tên theo cú pháp LopHP\_LT\_Tuan07\_STT\_MSSV\_Ten. Làm các bài tập theo yêu cầu sau:

*Toàn bộ bài làm phải chú thích rõ ràng và phải tuân theo Naming Conventions* [*http://java.sun.com/docs/codeconv/html/CodeConventions.doc8.html*](http://java.sun.com/docs/codeconv/html/CodeConventions.doc8.html)

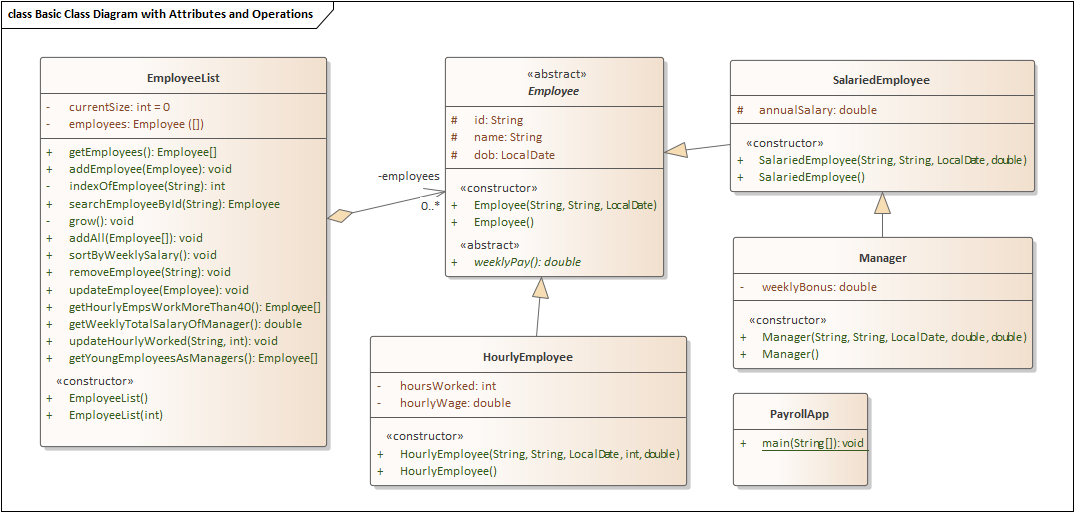
*Cố gắng sử dụng các phương thức của lớp Arrays và Stream https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/util/Arrays.html*

**Bài toán** Hiện thực xử lý bảng lương cho các loại nhân viên khác nhau.

* Nhân viên làm việc theo giờ được trả lương theo giờ, nhưng nếu họ làm việc hơn 40 giờ mỗi tuần, phần vượt quá sẽ được trả một rưỡi.
* Nhân viên làm công ăn lương được trả lương bất kể họ làm việc bao nhiêu giờ.
* Người quản lý là những nhân viên được trả lương, được trả lương gồm: Lương và thưởng.

Chương trình quản lý thông tin các nhân viên và tính toán tiền lương cần phải trả hằng tuần cho một tập hợp nhân viên *(dùng mảng)*.

**1/** Hiện thực mô hình lớp sau



* 1. Với các lớp thực thể, viết các phương thức:
* Tạo 1 constructor đầy đủ tham số, 1 default constructor *(khởi tạo các giá trị mặc định)*
* Các phương thức setters / getters.
* Viết phương thức toString biểu diễn dữ liệu như mẫu in ra bên dưới. Trong đó, kiểu ngày định dạng dd/MM/yyyy; Phân cách hàng nghìn lấy 2 số lẻ và đơn vị tiền tệ là $ cho kiểu dữ liệu là tiền tệ.
  1. Lớp EmployeeList, viết các phương thức
* Default constructor: Tạo mảng các nhân viên với kích thước mặc định là 10
* Copy constructor: Tạo một mảng các nhân viên với số lượng phần tử cho trước
* Phương thức trả về danh sách toàn bộ các nhân viên

+ getEmployees (): Employee

* Phương thức thêm 1 nhân viên vào danh sách, ném lỗi nếu bị trùng mã số nhân viên

Nếu mảng đầy, tăng kích thước mảng lên 1.5

+ addEmployee (emp: Employee): void

* Phương thức trả về index của nhân viên khi biết mã số

- indexOfEmployee (id: String): int

* Phương thức tìm kiếm nhân viên khi biết mã số. Trả về null, nếu không tìm thấy

+ searchEmployeeById (id: String): Employee

* Phương thức tăng kích thước mảng lên 1.5

- grow (): void

* Phương thức thêm một danh sách các nhân viên mới vào danh sách có trước, vẫn đảm bảo không bị trùng mã số nhân viên.

+ addAll (emps: Employee[]): void

* Phương thức sắp xếp danh sách nhân viên theo lương tuần tăng dần

+ sortByWeeklySalary (): void

* Phương thức xóa 1 nhân viên khi biết mã số, ném lỗi nếu không tồn tại mã số nhân viên

+ removeEmployee (id: String): void

* Phương thức cập nhật thông tin của nhân viên khi biết mã số, ném lỗi nếu không tồn tại mã số nhân viên *(dùng chung 1 phương thức cho cả 3 loại nhân viên)*

+ updateEmployee (newInfor: Employee): void

* Trả về danh sách các nhân viên là nhân viên theo giờ và làm việc hơn 40 giờ một tuần

+ getHourlyEmpsWorkMoreThan40 (): Employee

* Trả về tổng lương hàng tuần của tất cả các nhân viên là người quản lý

+ getWeeklyTotalSalaryOfManager (): double

* Cập nhật giờ làm việc của một nhân viên theo giờ khi biết mã số. Ném lỗi nếu giờ làm việc mới nhỏ hơn 0, hoặc nếu mã nhân viên không tồn tại, hoặc nhân viên không phải là nhân viên theo giờ

+ updateHourlyWorked (id: String, newHour: int): void

* Trả về danh sách nhân viên trẻ làm quản lý *(có tuổi < 30, tuổi = năm hiện tại – năm sinh)*

+ getYoungEmployeesAsManagers (): Employee

*Tất cả các Exception type được sử dụng là IllegalArgumentException và message phù hợp*

**2/** Viết lớp và tạo các Unit test để kiểm nghiệm

**3/** Viết lớp chứa phương thức main:

* Tạo sẳn 5 nhân viên của 3 loại
* Tạo menu các lựa chọn, cho phép người dùng lựa chọn chức năng thực hiện, dữ liệu được nhập từ bàn phím.

