**Text

Description automatically generatedTRƯỜNG CAO ĐẲNG THỰC HÀNH FPT POLYTECHNC HỒ CHÍ MINH**

**BÁO CÁO ASSIGNMENT**

**Môn:** Lập trình Java 1 **MOB1014**

**Chuyên ngành:** Thiết kế web (Back-end)

**Họ tên:** Ngô Quốc Đạt

**MSSV:** PD05994 – **Lớp:** WE17304

**Giáo viên HD:** Nguyễn Trí Định



**Hồ Chí Minh**, Ngày 17 tháng 02 năm 2022

1. Class **NhanVien**

*package* Assignment;  
  
*import* java.util.Scanner;  
  
*public abstract class* NhanVien {  
 *private* String maNv;  
 *private* String hoTen;  
 *private double* luong;  
  
 *public* NhanVien() {  
 }  
  
 *public* NhanVien(String maNv, String hoTen, *double* luong) {  
 *this*.maNv = maNv;  
 *this*.hoTen = hoTen;  
 *this*.luong = luong;  
 }  
  
 *public* String getMaNv() {  
 *return* maNv;  
 }  
  
 *public void* setMaNv(String maNv) {  
 *this*.maNv = maNv;  
 }  
  
 *public* String getHoTen() {  
 *return* hoTen;  
 }  
  
 *public void* setHoTen(String hoTen) {  
 *this*.hoTen = hoTen;  
 }  
  
 *public double* getLuong() {  
 *return* luong;  
 }  
  
 *public void* setLuong(*double* luong) {  
 *this*.luong = luong;  
 }  
  
 *public abstract double* getThuNhap();  
  
 *public double* getThueTN() {  
 *double* thue = 0;  
 *if* (getThuNhap() > 15000000) {  
 thue = getThuNhap() \* 0.12;  
 } *else if* (getThuNhap() >= 8000000 && getThuNhap() <= 15000000) {  
 thue = getThuNhap() \* 0.1;  
 } *else* {  
 thue = getThuNhap();  
 }  
  
 *return* thue;  
 }  
  
 *public void* nhapThongTin() {  
 Scanner sc = *new* Scanner(System.in);  
 System.out.print("Nhập mã nhân viên: ");  
 *this*.maNv = sc.nextLine();  
 System.out.print("Nhập họ tên: ");  
 *this*.hoTen = sc.nextLine();  
 System.out.print("Nhập lương: ");  
 *this*.luong = sc.nextDouble();  
 }  
  
 *public void* xuatThongTin() {  
 System.out.print("Mã NV: " + *this*.maNv + " - Họ tên: " + *this*.hoTen + " - Lương: " + *this*.luong + " - Thu nhập: " + getThuNhap() + " - Thuế: " + getThueTN());  
 }  
}

2. Class **HanhChinh**

*package* Assignment;  
  
*public class* HanhChinh *extends* NhanVien {  
 *public* HanhChinh() {  
 }  
  
 *public* HanhChinh(String maNv, String hoTen, *double* luong) {  
 *super*(maNv, hoTen, luong);  
 }  
  
 @Override  
 *public double* getThuNhap() {  
 *return super*.getLuong();  
 }  
}

3. Class **TiepThi**

*package* Assignment;  
  
*import* java.util.Scanner;  
  
*public class* TiepThi *extends* NhanVien {  
 *private double* doanhSo;  
 *private double* hoaHong;  
  
 *public* TiepThi() {  
 }  
  
 *public* TiepThi(String maNv, String hoTen, *double* luong, *double* doanhSo, *double* hoaHong) {  
 *super*(maNv, hoTen, luong);  
 *this*.doanhSo = doanhSo;  
 *this*.hoaHong = hoaHong;  
 }  
  
 *public double* getDoanhSo() {  
 *return* doanhSo;  
 }  
  
 *public void* setDoanhSo(*double* doanhSo) {  
 *this*.doanhSo = doanhSo;  
 }  
  
 *public double* getHoaHong() {  
 *return* hoaHong;  
 }  
  
 *public void* setHoaHong(*double* hoaHong) {  
 *this*.hoaHong = hoaHong;  
 }  
  
 @Override  
 *public double* getThuNhap() {  
 *return super*.getLuong() + *this*.doanhSo + *this*.hoaHong;  
 }  
  
 @Override  
 *public void* nhapThongTin() {  
 *super*.nhapThongTin();  
 Scanner sc = *new* Scanner(System.in);  
 System.out.print("Nhập doanh số: ");  
 *this*.doanhSo = sc.nextDouble();  
 System.out.print("Nhập hoa hồng: ");  
 *this*.hoaHong = sc.nextDouble();  
 }  
  
 @Override  
 *public void* xuatThongTin() {  
 *super*.xuatThongTin();  
 System.out.print(" - Doanh số: " + *this*.doanhSo + " - Hoa hồng: " + *this*.hoaHong);  
 }  
}

4. Class **TruongPhong**

*package* Assignment;  
  
*import* java.util.Scanner;  
  
*public class* TruongPhong *extends* NhanVien {  
 *private double* luongTrachNhiem;  
  
 *public* TruongPhong() {  
 }  
  
 *public* TruongPhong(String maNv, String hoTen, *double* luong, *double* luongTrachNhiem) {  
 *super*(maNv, hoTen, luong);  
 *this*.luongTrachNhiem = luongTrachNhiem;  
 }  
  
 *public double* getLuongTrachNhiem() {  
 *return* luongTrachNhiem;  
 }  
  
 *public void* setLuongTrachNhiem(*double* luongTrachNhiem) {  
 *this*.luongTrachNhiem = luongTrachNhiem;  
 }  
  
 @Override  
 *public double* getThuNhap() {  
 *return* getLuong() + *this*.luongTrachNhiem;  
 }  
  
 @Override  
 *public void* nhapThongTin() {  
 *super*.nhapThongTin();  
 Scanner sc = *new* Scanner(System.in);  
 System.out.print("Nhập lương trách nhiệm: ");  
 *this*.luongTrachNhiem = sc.nextDouble();  
 }  
  
 @Override  
 *public void* xuatThongTin() {  
 *super*.xuatThongTin();  
 System.out.print(" - Lương trách nhiệm: " + *this*.luongTrachNhiem);  
 }  
}

5. Class **Main**

*package* Assignment;  
  
*import* java.util.Scanner;  
  
*public class* Main {  
 *public static void* main(String[] args) {  
 *menu*();  
 }  
  
 *public static void* menu() {  
 Scanner scanner = *new* Scanner(System.in);  
 QuanLy quanLy = *new* QuanLy();  
  
 *int* choice;  
  
 *do* {  
 System.out.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_MENU\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");  
 System.out.println("1. Nhập danh sách nhân viên");  
 System.out.println("2. Xuất danh sách nhân viên");  
 System.out.println("3. Tìm kiếm nhân viên theo mã");  
 System.out.println("4. Xuất nhân viên theo mã");  
 System.out.println("5. Cập nhật thông tin nhân viên");  
 System.out.println("6. Tìm kiếm nhân viên theo lương");  
 System.out.println("7. Sắp xếp danh sách nhân viên theo họ tên");  
 System.out.println("8. Sắp xếp danh sách nhân viên theo thu nhập");  
 System.out.println("9. Xuất 5 nhân viên có thu nhập cao nhất công ty");  
 System.out.println("0. Thoát chương trình");  
 System.out.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");  
  
 System.out.print("Nhập lựa chọn của bạn: ");  
 choice = scanner.nextInt();  
 System.out.println();  
  
 *switch* (choice) {  
 *case* 1 -> {  
 System.out.println("Nhập danh sách nhân viên");  
 quanLy.nhapDanhSach();  
 }  
 *case* 2 -> {  
 System.out.println("Xuất danh sách nhân viên");  
 quanLy.xuatDanhSach();  
 }  
 *case* 3 -> {  
 System.out.println("Tìm kiếm nhân viên theo mã");  
 quanLy.timNhanVien();  
 }  
 *case* 4 -> {  
 System.out.println("Xoá nhân viên theo mã");  
 quanLy.xoaNhanVien();  
 }  
 *case* 5 -> {  
 System.out.println("Cập nhật thông tin nhân viên");  
 quanLy.capNhatNhanVien();  
 }  
 *case* 6 -> {  
 System.out.println("Tìm kiếm nhân viên theo lương");  
 quanLy.timTheoKhoangLuong();  
 }  
 *case* 7 -> {  
 System.out.println("Sắp xếp danh sách nhân viên theo họ tên");  
 quanLy.sapXepTheoTen();  
 }  
 *case* 8 -> {  
 System.out.println("Sắp xếp danh sách nhân viên theo thu nhập");  
 quanLy.sapXepTheoThuNhap();  
 }  
 *case* 9 -> {  
 System.out.println("Xuất 5 nhân viên có thu nhập cao nhất công ty");  
 quanLy.xuatTop5ThuNhap();  
 }  
 *case* 0 -> {  
 System.out.println("Thoát chương trình");  
 System.*exit*(0);  
 }  
 *default* -> System.out.println("Nhập sai lựa chọn");  
 }  
  
 System.out.println();  
 System.out.print("Quay lại trang menu? (Y/N): ");  
 String choice1 = scanner.next();  
 *if* (choice1.equalsIgnoreCase("N")) {  
 *break*;  
 }  
 } *while* (*true*);  
 }  
}

6. Class **QuanLy**

*package* Assignment;  
  
*import* java.util.ArrayList;  
*import* java.util.Scanner;  
  
*public class* QuanLy {  
 Scanner sc = *new* Scanner(System.in);  
 ArrayList<NhanVien> listNhanVien = *new* ArrayList<>();  
  
 *public void* nhapDanhSach() {  
 *int* choice;  
  
 *do* {  
 System.out.println("---CHỌN KIỂU NHÂN VIÊN CẦN NHẬP---");  
 System.out.println("1. Nhân viên hành chính");  
 System.out.println("2. Nhân viên tiếp thị");  
 System.out.println("3. Trưởng phòng");  
 System.out.println("0. Thoát");  
 System.out.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");  
  
 System.out.print("Nhập lựa chọn của bạn: ");  
 choice = sc.nextInt();  
 System.out.println();  
  
 *switch* (choice) {  
 *case* 1 -> {  
 HanhChinh hc = *new* HanhChinh();  
 hc.nhapThongTin();  
 listNhanVien.add(hc);  
 }  
 *case* 2 -> {  
 TiepThi tt = *new* TiepThi();  
 tt.nhapThongTin();  
 listNhanVien.add(tt);  
 }  
 *case* 3 -> {  
 TruongPhong tp = *new* TruongPhong();  
 tp.nhapThongTin();  
 listNhanVien.add(tp);  
 }  
 *case* 0 -> {  
 System.out.println("Đã thoát chức năng nhập danh sách!");  
 }  
 *default* -> System.out.println("Nhập sai lựa chọn");  
 }  
  
 *if* (choice != 0) {  
 System.out.println();  
 System.out.print("Bạn có muốn tiếp tục nhập danh sách? (Y/N): ");  
 String choice1 = sc.next();  
 *if* (choice1.equalsIgnoreCase("N")) {  
 *break*;  
 }  
 }  
 } *while* (choice != 0);  
 }  
  
 *public void* xuatDanhSach() {  
 *if* (listNhanVien.size() == 0) {  
 System.out.println("Không có nhân viên nào trong hệ thống!");  
 } *else* {  
 *for* (NhanVien nv : listNhanVien) {  
 *if* (listNhanVien.size() == 0) {  
 System.out.println("Không tìm thấy nhân viên nào trong hệ thống!");  
 } *else* {  
 nv.xuatThongTin();  
 System.out.println();  
 }  
 }  
 }  
 }  
  
 *public void* timNhanVien() {  
 System.out.print("Nhập mã nhân viên cần tìm: ");  
 String maNV = sc.next();  
 *boolean* flag = *false*;  
  
 *for* (NhanVien nv : listNhanVien) {  
 *if* (nv.getMaNv().equalsIgnoreCase(maNV)) {  
 System.out.println("\nĐã tìm thấy nhân viên có mã " + maNV);  
 nv.xuatThongTin();  
 System.out.println();  
 flag = *true*;  
 *break*;  
 }  
 }  
  
 *if* (!flag) {  
 System.out.println("Không tìm thấy nhân viên có mã " + maNV);  
 }  
 }  
  
 *public void* xoaNhanVien() {  
 System.out.print("Nhập mã nhân viên cần xóa: ");  
 String maNV = sc.next();  
 *boolean* flag = *false*;  
  
 *for* (NhanVien nv : listNhanVien) {  
 *if* (nv.getMaNv().equalsIgnoreCase(maNV)) {  
 listNhanVien.remove(nv);  
 System.out.println("Đã xóa nhân viên có mã " + maNV);  
 flag = *true*;  
 *break*;  
 }  
 }  
  
 *if* (!flag) {  
 System.out.println("Không tìm thấy nhân viên có mã " + maNV);  
 }  
 }  
  
 *public void* capNhatNhanVien() {  
 System.out.print("Nhập mã nhân viên cần cập nhật: ");  
 String maNV = sc.next();  
 *boolean* flag = *false*;  
  
 *for* (NhanVien nv : listNhanVien) {  
 *if* (nv.getMaNv().equalsIgnoreCase(maNV)) {  
 System.out.println("\nĐã tìm thấy nhân viên có mã " + maNV);  
 nv.nhapThongTin();  
 System.out.println("Đã cập nhật thông tin nhân viên có mã " + maNV);  
 flag = *true*;  
 *break*;  
 }  
 }  
  
 *if* (!flag) {  
 System.out.println("Không tìm thấy nhân viên có mã " + maNV);  
 }  
 }  
  
 *public void* timTheoKhoangLuong() {  
 System.out.print("Nhập khoảng lương bắt đầu: ");  
 *double* khoangLuongBatDau = sc.nextInt();  
 System.out.print("Nhập khoảng lương kết thúc: ");  
 *double* khoangLuongKetThuc = sc.nextInt();  
 *boolean* flag = *false*;  
  
 *for* (NhanVien nv : listNhanVien) {  
 *if* (nv.getThuNhap() >= khoangLuongBatDau && nv.getThuNhap() <= khoangLuongKetThuc) {  
 nv.xuatThongTin();  
 System.out.println();  
 flag = *true*;  
 }  
 }  
  
 *if* (!flag) {  
 System.out.println("Không tìm thấy nhân viên có thu nhập trong khoảng " + khoangLuongBatDau + " - " + khoangLuongKetThuc);  
 }  
 }  
  
 *public void* sapXepTheoTen() {  
 listNhanVien.sort((NhanVien nv1, NhanVien nv2) -> nv1.getHoTen().compareTo(nv2.getHoTen()));  
 xuatDanhSach();  
 }  
  
 *public void* sapXepTheoThuNhap() {  
 listNhanVien.sort((NhanVien nv1, NhanVien nv2) -> (*int*) (nv1.getThuNhap() - nv2.getThuNhap()));  
 xuatDanhSach();  
 }  
  
 *public void* xuatTop5ThuNhap() {  
 listNhanVien.sort((NhanVien nv1, NhanVien nv2) -> (*int*) (nv2.getThuNhap() - nv1.getThuNhap()));  
  
 *if* (listNhanVien.size() == 0) {  
 System.out.println("Không có nhân viên nào trong hệ thống!");  
 } *else* {  
 *for* (NhanVien nv : listNhanVien) {  
 *if* (listNhanVien.indexOf(nv) < 5) {  
 nv.xuatThongTin();  
 System.out.println();  
 } *else* {  
 *break*;  
 }  
 }  
 }  
 }  
}

**CHẠY CHƯƠNG TRÌNH**

Text

Description automatically generated

Kết quả khi chạy chương trình

Text

Description automatically generated

Kết quả chức năng **Nhập danh sách NV**

**Text

Description automatically generated**

Kết quả chức năng **Xuất danh sách NV**

**Text

Description automatically generated**

Kết quả chức năng **Tìm kiếm NV theo mã**

Graphical user interface, text

Description automatically generated with medium confidence

Kết quả chức năng **Xoá NV theo mã**

Text

Description automatically generated with low confidence

Danh sách NV sau khi xoá NV

Text

Description automatically generated

Kết quả sau khi cập nhật NV

A picture containing text

Description automatically generated

Danh sách nhân viên sau khi cập nhật

Graphical user interface, text

Description automatically generated with medium confidence

Kết quả chức năng tìm kiếm NV theo lương

A picture containing text

Description automatically generated

Kết quả chức năng sắp xếp danh sách NV theo tên

A picture containing graphical user interface

Description automatically generated

Kết quả danh sách NV sắp xếp theo thu nhập

Graphical user interface

Description automatically generated with medium confidence

Kết quả chức năng top 5 nhân viên thu nhập cao nhất

**LỜI CẢM ƠN**

Cho em gửi lời cảm ơn đến thầy Nguyễn Trí Định đã giảng dạy tại trường FPT Polytechnic, người đã truyền dạy cho em nhiều kiến thức hay và hữu ích để em có thể hoàn thành bài assignment này.