**ÔN TẬP PHẦN THỰC HÀNH WPF**

**Đề bài**

Xây dựng phần mềm quản lý Nhân viên theo mô tả dưới đây:

Công ty A muốn xây dựng phần mềm quản lý ***nhân viên bán hàng, nhân viên giao nhận***. Mỗi nhân viên sẽ bao gồm các thông tin như: *Mã nhân viên (string), Họ tên (string), Giới tính (string), Số điện thoại (string), Ngày vào làm (datetime)*. Nhân viên bán hàng sẽ có thêm thuộc tính *Doanh số (double),* nhân viên giao nhận sẽ có thêm thuộc tính *Phụ cấp nhiên liệu (double)*.

Lương cơ bản của nhân viên là 7 triệu đồng, đối với nhân viên bán hàng sẽ được cộng thêm 10% doanh số, đối với nhân viên giao nhận sẽ được cộng thêm phụ cấp nhiên liệu.

Sinh viên hãy xây dựng mô hình Lớp hướng đối tượng để quản lý nhân viên và viết chương trình phần mềm với các chức năng yêu cầu như hình dưới đây:





1. **~~Giữ tính cân đối~~** ~~của các điều khiển khi phóng to, thu nhỏ màn hình;~~ **~~hỏi xác nhận~~** ~~khi đóng chương trình.~~
2. **~~Khi mới mở chương trình~~**~~: giới tính Nam mặc định được chọn, ngày vào làm mặc định là ngày hiện tại, loại nhân viên bán hàng mặc định được chọn.~~
3. **~~RadioButton loại NV~~**~~: Khi chọn nhân viên bán hàng thì TextBox nhập doanh số hiển thị, khi chọn nhân viên giao nhận thì TextBox nhập tiền phụ cấp nhiên liệu hiển thị.~~
4. **~~Nút Thêm~~**~~: Xóa dữ liệu ở các TextBox nhập liệu, các giá trị mặc định được gán lại như lúc mới mở chương trình, đưa con trỏ lên TextBox nhập mã nhân viên.~~
5. **Nút Lưu**: Kiểm tra mọi thông tin có hợp lệ hay không (mã nhân viên, họ tên không được để trống; ngày vào làm không được lớn hơn ngày hiện tại, …), nếu hợp lệ thì cho phép lưu vào ListView danh sách nhân viên, tô nền vàng dữ liệu nếu nhân viên có thâm niên trên 5 năm (thâm niên được tính dựa vào năm vào làm), chọn dòng dữ liệu vừa mới được thêm vào trên ListView.
6. **ListView danh sách nhân viên**: khi chọn 1 dòng trên ListView => hiển thị đầy đủ dữ liệu cho các điều khiển nhập liệu như lúc nhập vào ban đầu. *Chưa hiện được nhân viên Bán hàng*
7. **~~Nút Xóa~~**~~: Xóa nhân viên được chọn trong ListView, hỏi xác nhận trước khi xóa, chọn dòng dữ liệu liền kề sau dòng được xóa trên ListView, trường hợp xóa hết dữ liệu trong ListView thì hiển thị lại các thông tin mặc định như lúc mới mở chương trình.~~
8. **~~Nút Sửa~~**~~: Cho phép chỉnh sửa dữ liệu trong ListView (chọn 1 dòng trong ListView => hiển thị dữ liệu lên các điều khiển tương ứng cho phép chỉnh sửa => nhấn nút Sửa cập nhật lại dữ liệu trong ListView và vẫn giữ dòng chọn hiện hành). Lưu ý: cho phép sửa doanh số hoặc phụ cấp nhiên liệu ứng với nhân viên bán hàng hoặc nhân viên giao nhận.~~
9. **~~Nút Sắp xếp~~**~~: Sắp xếp dữ liệu trong ListView danh sách nhân viên theo thâm niên giảm dần nếu cùng thâm niên thì sắp xếp theo họ tên tăng dần.~~
10. **~~Nút Thống kê~~**~~: Hiển thị Messagebox cho biết Công ty hiện có bao nhiêu nhân viên bán hàng, bao nhiêu nhân viên giao nhận, tổng lương chi ứng với từng loại nhân viên.~~
11. // Calculate the statistics
12. int soNhanVienBanHang = 0;
13. int soNhanVienGiaoNhan = 0;
14. decimal tongLuongBanHang = 0;
15. decimal tongLuongGiaoNhan = 0;
16. foreach (EmployeeItem employee in lvDS.Items)
17. {
18. if (employee.TypeId == "Bán Hàng")
19. {
20. soNhanVienBanHang++;
21. tongLuongBanHang += 7000000 + Convert.ToDecimal(employee.DoanhSo) \* 0.1m;
22. }
23. else if (employee.TypeId == "Giao nhận")
24. {
25. soNhanVienGiaoNhan++;
26. tongLuongBanHang += 7000000 + Convert.ToDecimal(employee.DoanhSo);
27. }
28. }
29. // Display the statistics
30. string message = string.Format("Công ty hiện có {0} nhân viên bán hàng và {1} nhân viên giao nhận.\nTổng lương cho nhân viên bán hàng: {2:C}\nTổng lương cho nhân viên giao nhận: {3:C}", soNhanVienBanHang, soNhanVienGiaoNhan, tongLuongBanHang, tongLuongGiaoNhan);
31. MessageBox.Show(message, "Thống kê");