**BẢO MẬT CƠ SỞ DỮ LIỆU**

**Lab04 – Cá nhân**

**Mã hóa dữ liệu từ client trước khi lưu xuống CSDL**

**Giải mã dữ liệu ở client sau khi truy vấn dữ liệu từ CSDL**

Sinh viên:

**Đỗ Trọng Nghĩa - 18120477**

C:\Users\tdqua_000\Dropbox\SS-Slides\DeCuong-CDIO\Template CDIO v4.2\Templates\Hinh anh\LogoTruong.png

Khoa Công nghệ Thông tin

Đại học Khoa học Tự nhiên TP HCM

[I. Tạo cơ sở dữ liệu 3](#_Toc74441946)

[I.1. Mô hình CSDL 3](#_Toc74441947)

[I.2. Các Stored Procedure chính 4](#_Toc74441948)

[I.2.1. SP\_INS\_ ENCRYPT \_SINHVIEN 4](#_Toc74441949)

[I.2.2. SP\_INS\_ ENCRYPT \_NHANVIEN 4](#_Toc74441950)

[I.2.3. SP\_SEL\_ ENCRYPT \_NHANVIEN 5](#_Toc74441951)

[I.2.4. SP\_LOG\_IN 5](#_Toc74441952)

[II. Mã hóa dữ liệu ở phía Client 6](#_Toc74441953)

[II.1. Class AES256 6](#_Toc74441954)

[II.1.1. Thiết lâp các thuộc tính 6](#_Toc74441955)

[II.1.2. Mã hóa 7](#_Toc74441956)

[II.1.3. Giải mã 8](#_Toc74441957)

[II.2. Class MD5 9](#_Toc74441958)

[II.3. Class SHA1 10](#_Toc74441959)

[III. Sử dụng công cụ SQL Profile để theo dõi các màn hình 11](#_Toc74441960)

[III.1. Theo dõi màn hình đăng nhập 11](#_Toc74441961)

[III.1.1. Đối với tài khoản trong bảng SINHVIEN 11](#_Toc74441962)

[III.1.2. Đối với tài khoản trong bảng NHANVIEN 13](#_Toc74441963)

[III.2. Theo dõi màn hình load danh sách nhân viên 15](#_Toc74441964)

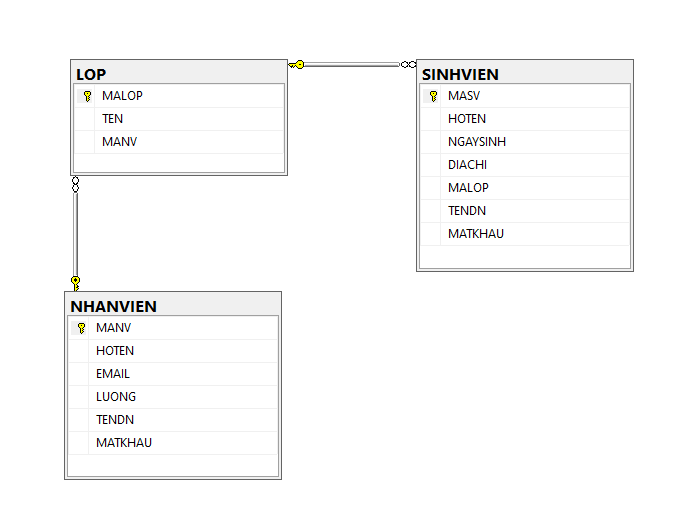
[III.3. Theo dõi màn hình thêm nhân viên 17](#_Toc74441965)

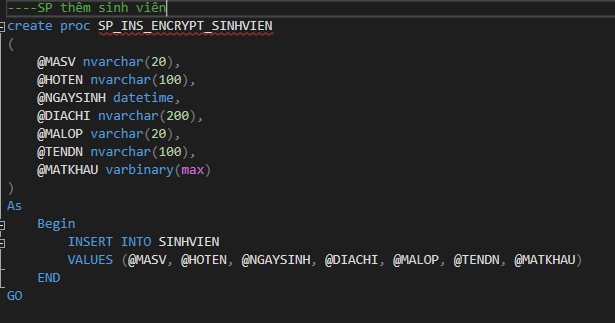
[III.4. Theo dõi thêm các thao tác xóa và chỉnh sửa nhân viên 19](#_Toc74441966)

[III.4.1. Chỉnh sửa 19](#_Toc74441967)

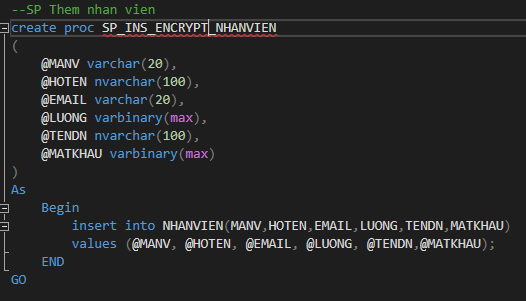
[III.4.2. Xóa nhân viên 21](#_Toc74441968)

1. Tạo cơ sở dữ liệu
   1. Mô hình CSDL

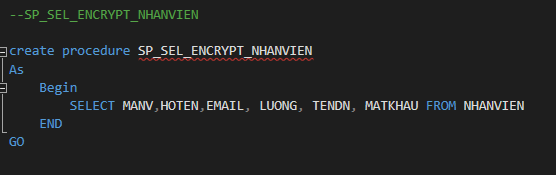
* Thiết lập CSDL có mô hình như sau
  1. Các Stored Procedure chính
     1. SP\_INS\_ ENCRYPT \_SINHVIEN



* + 1. SP\_INS\_ ENCRYPT \_NHANVIEN



* + 1. SP\_SEL\_ ENCRYPT \_NHANVIEN

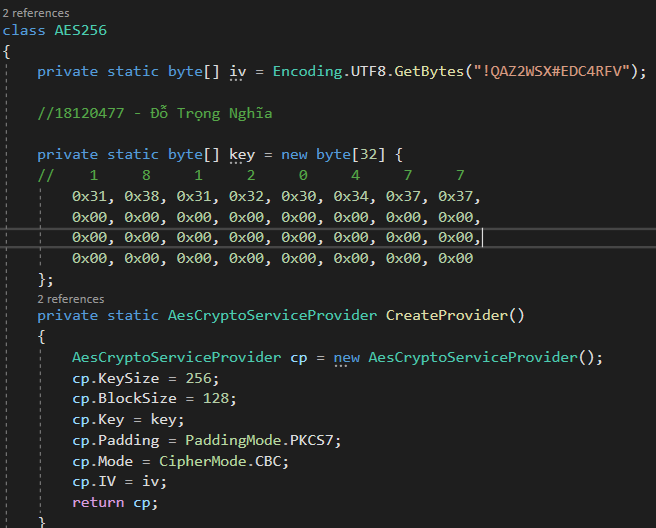


* + 1. SP\_LOG\_IN
* Với mỗi tên đăng nhập, nhận mật khẩu mã hóa dưới 2 dạng MD5 và SHA1 (cách mã hóa mật khẩu của sinh viên và nhân viên). Tiến hành select, cột đầu tiên là 1 – Nếu có tồn tại tài khoản, 0 – Nếu không tồn tại tài khoản. Cột thứ hai là 0 – Nếu là tài khoản sinh viên, mã nhân viên – Nếu là của nhân viên đó đăng nhập.

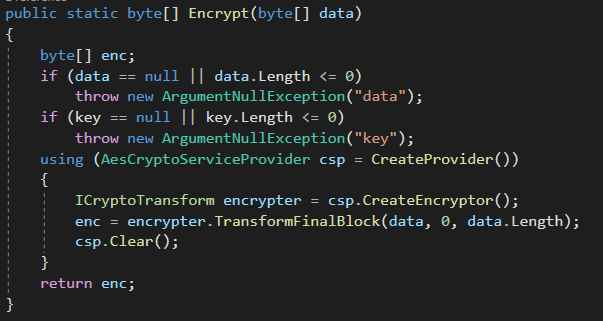


1. Mã hóa dữ liệu ở phía Client
   1. Class AES256
      1. Thiết lâp các thuộc tính

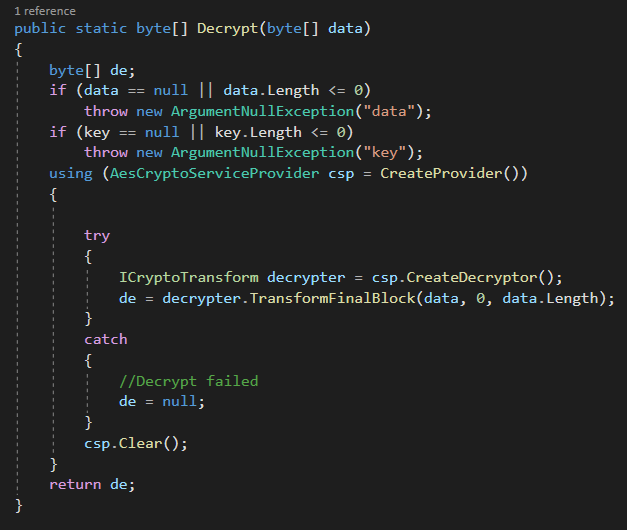
* Sử dụng thư viện **AesCryptoServiceProvider** và tạo phương thức CreateProvider() để set Keysize (256 bit), BlockSize, IV, Padding, Mode. Trong đó Key là mã số sinh viên “18120477” (chuỗi dưới dạng byte). Vì vậy, khi tạo data (trong script TaoData.sql) tuân theo các quy tắc mã hóa này để dễ dàng giải mã.



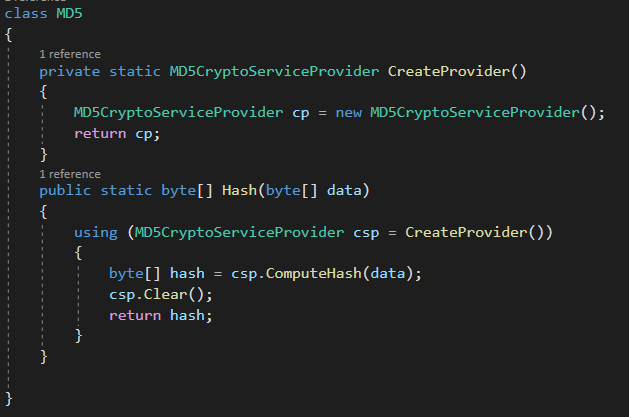
* + 1. Mã hóa
* Các thuộc tính dựa vào phương thức CreateProvider()



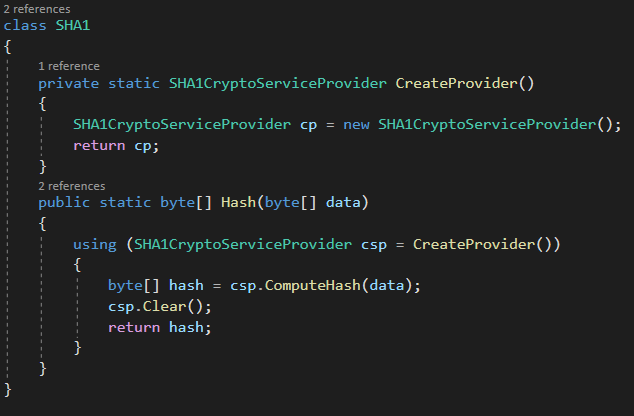
* + 1. Giải mã
* Các thuộc tính dựa vào phương thức CreateProvider()



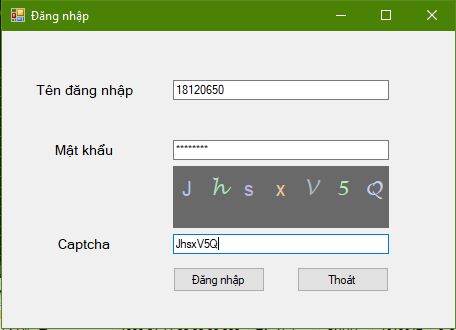
* 1. Class MD5
* Dùng để mã hóa mật khẩu của sinh viên

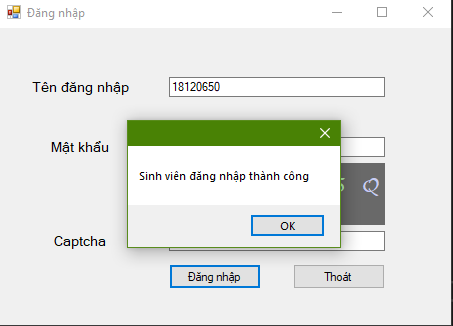


* 1. Class SHA1
* Dùng để mã hóa mật khẩu nhân viên

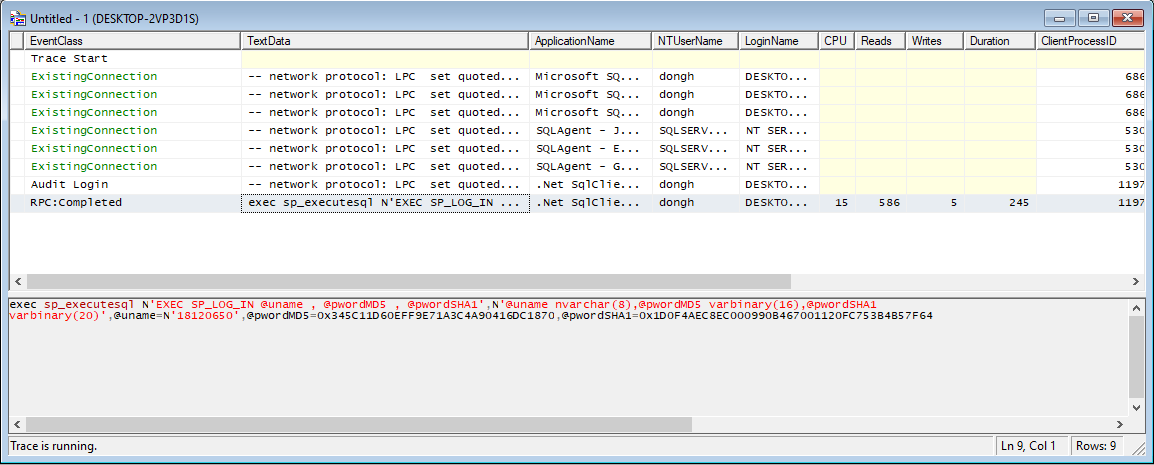


1. Sử dụng công cụ SQL Profile để theo dõi các màn hình
   1. Theo dõi màn hình đăng nhập
      1. Đối với tài khoản trong bảng SINHVIEN

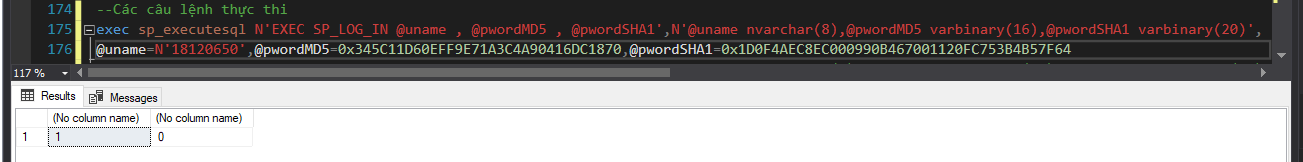




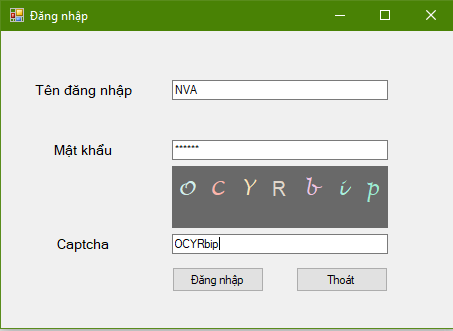
* Kết quả ở SQL Profile

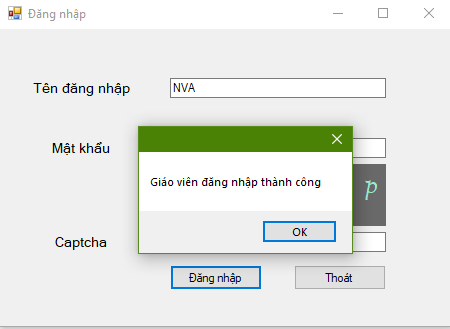


* Thực thi câu lệnh trên ở SQL Server

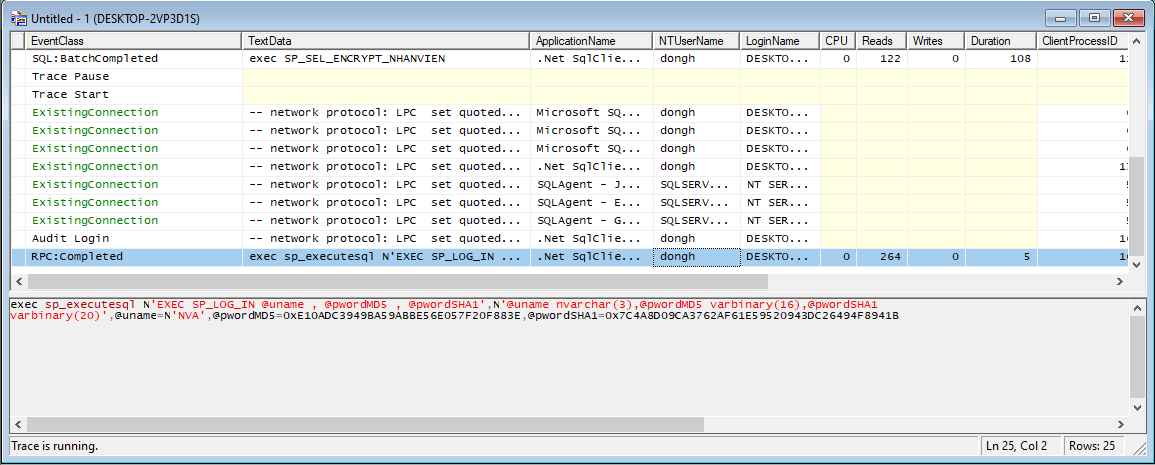


* Nhận xét: Kết quả trả ra của stored procedure này giống với mong đợi đối với người dùng thuộc bảng SINHVIEN (1 – Cột đầu thỏa điều kiện có tồn tại tài khoản, 0 – Cột thứ hai thỏa điều kiện tài khoản là của sinh viên). Các thuộc tính cần giữ bí mật như mật khẩu đã bị mã hóa ở phía Client rồi truyền xuống dưới CSDL. Vì vậy nên an toàn đối với người trung gian và kẻ tấn công (vì chỉ có chuỗi mật khẩu bị mã hóa). Người quản lí dữ liệu cũng không thể biết được bí mật của người sử dụng
  + 1. Đối với tài khoản trong bảng NHANVIEN

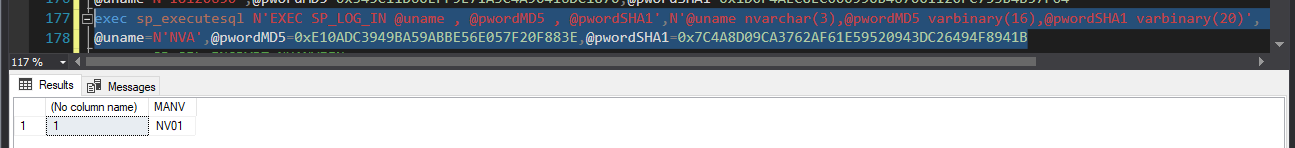




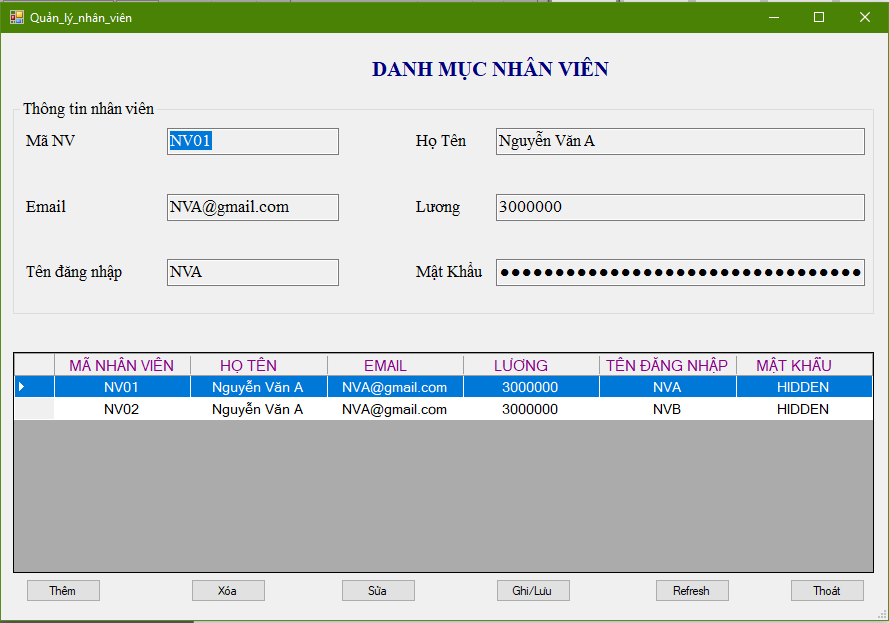
* Kết quả ở SQL Profile



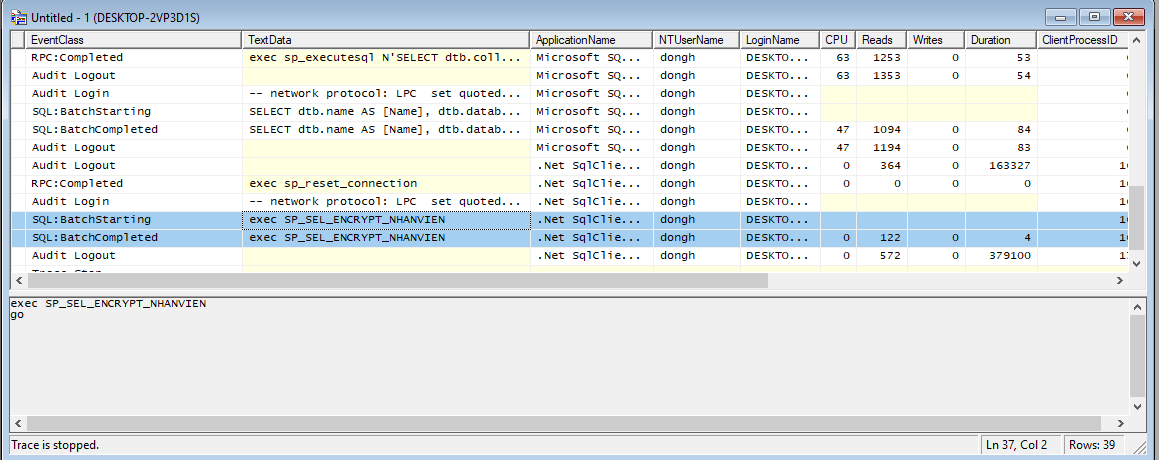
* Thực thi câu lệnh trên ở SQL Server



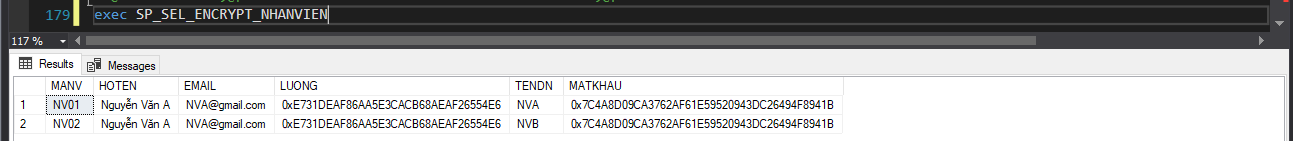
* Nhận xét: Cũng giống với SINHVIEN kết quả trả ra của stored procedure này giống với mong đợi đối với người dùng thuộc bảng NHANVIEN (1 – Cột đầu thỏa điều kiện có tồn tại tài khoản, NV01 – Cột thứ hai thỏa điều kiện là mã số của nhân viên đăng nhâp). Các thông tin cần thiết đều đã bị mã hóa ở phía Client trước khi gửi đến CSDL thông qua Stored Procedure, đảm bảo an toàn bảo mât cho người sử dụng. Người quản lí dữ liệu cũng không thể biết được bí mật của người sử dụng.
  1. Theo dõi màn hình load danh sách nhân viên
* Sau khi đăng nhập thành công bằng một trong hai tài khoàn trên sẽ hiện lên thì sẽ hiện màn hình quản lí nhân viên
* Vì mật khẩu được mã hóa bằng SHA1, không decrypt được nên textbox mật khẩu chỉ là chuỗi hash.



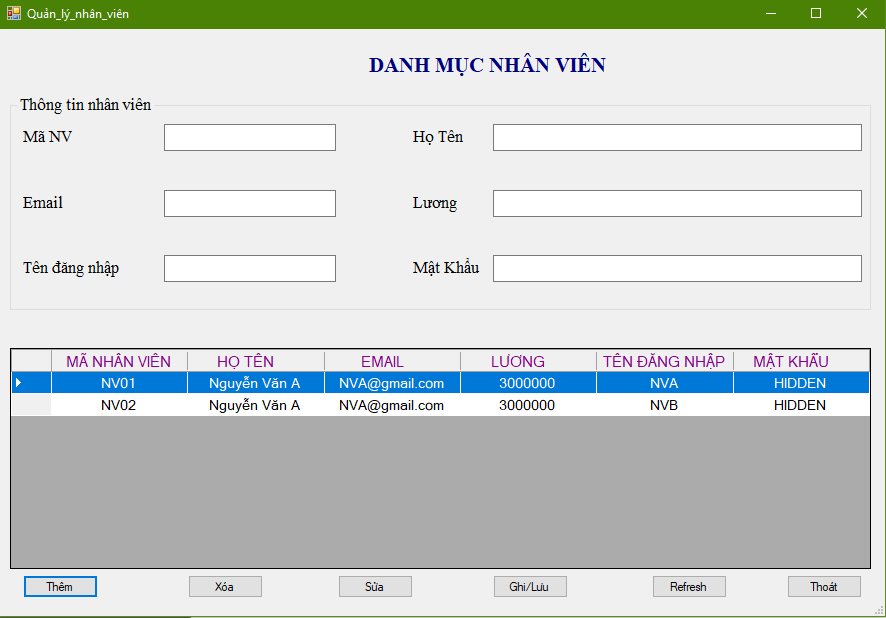
* Theo dõi kết quả ở SQL Profile



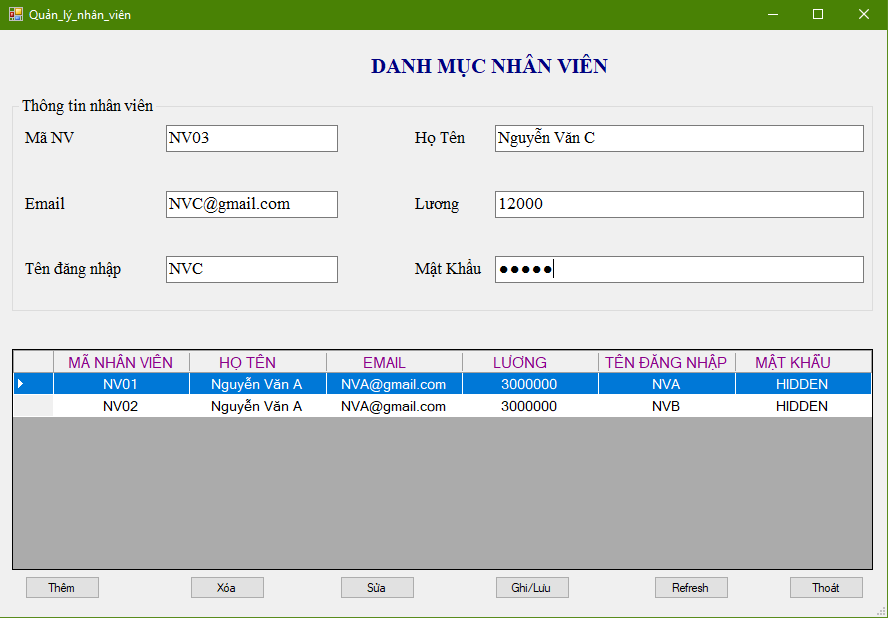
* Thực thi câu lệnh trên ở SQL Server



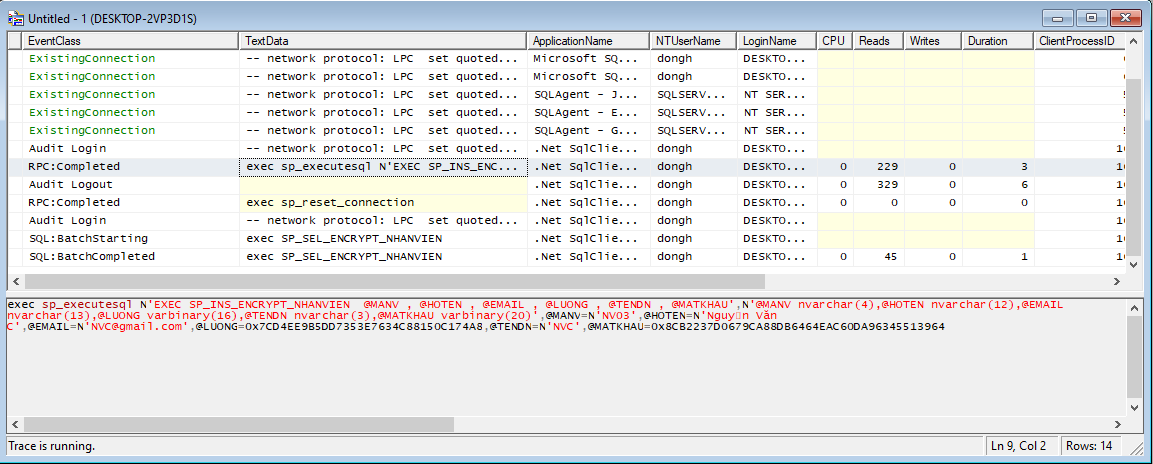
* Nhận xét: Thuộc tính LUONG của các nhân viên đã được mã hóa và chỉ có người dùng bên phía Client mới có thể giải mã. Đảm bảo bí mật vì chỉ những người có khóa mới có thể xem được dữ liệu.
  1. Theo dõi màn hình thêm nhân viên
* Sau khi tải xong danh sách, ta chọn chức năng “Thêm” để mở khóa các textbox

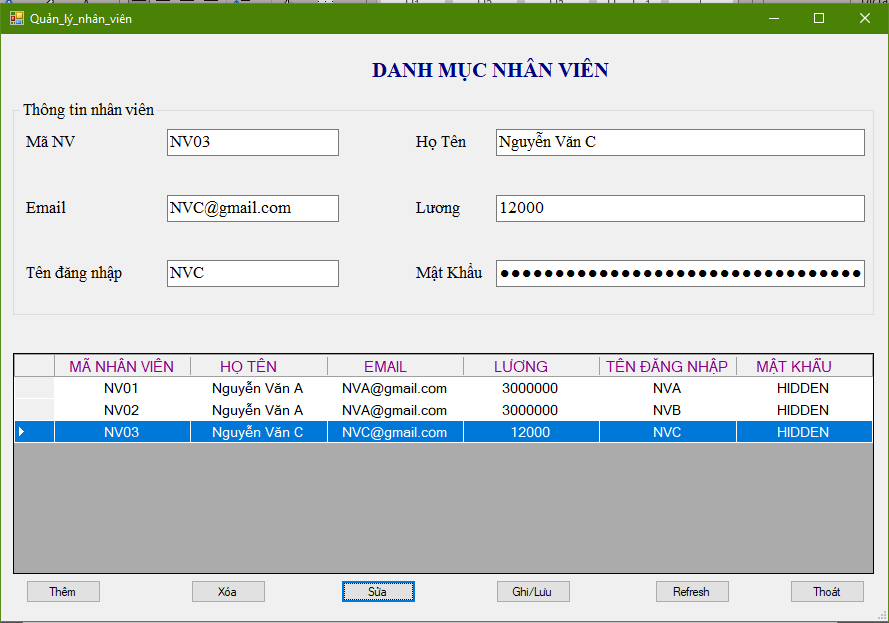


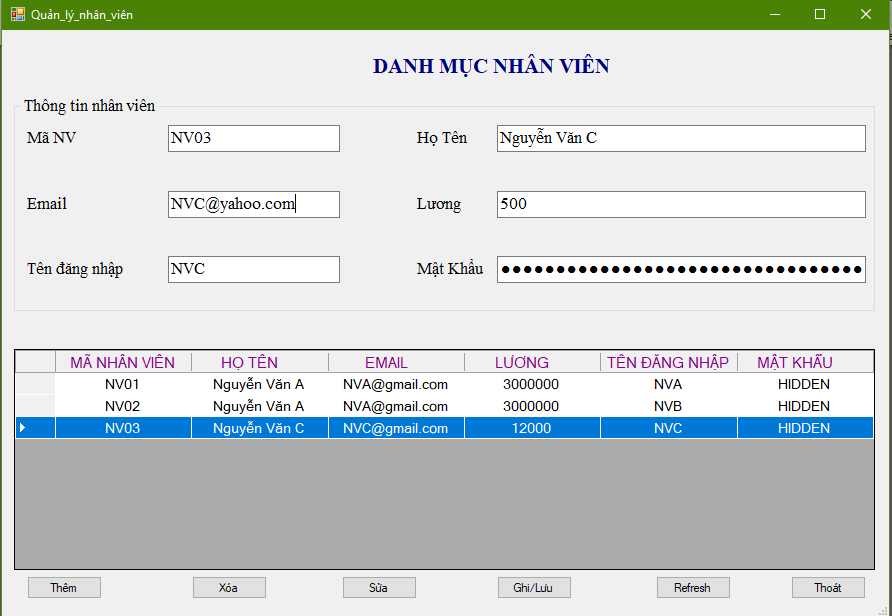
* Nhập đầy đủ các trường (trong đó password là 12345). Sau đó chọn Ghi/Lưu



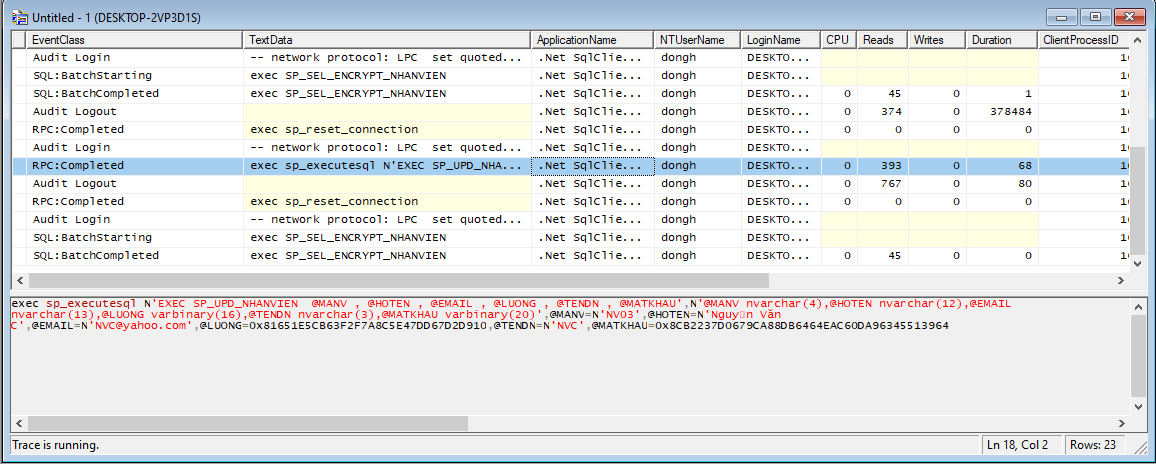
* Kết quả ở SQL Profile



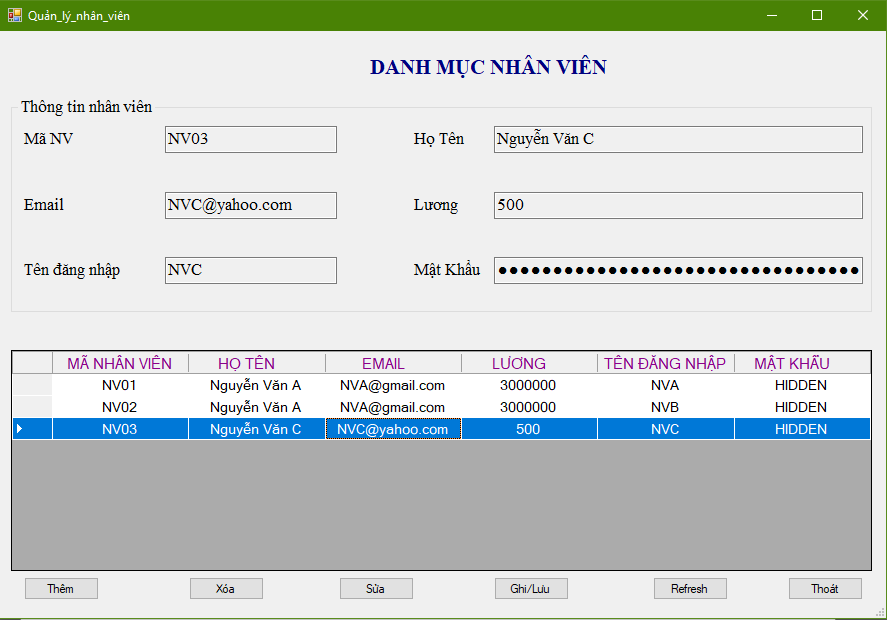
* Cũng giống như các tác vụ ở trên, việc mã hóa và giải mã ở phía Client đảm bảo tính án toàn cao hơn so với việc mã hóa và giải mã ở phía Server. Khi ta lưu trữ và truyền dữ liệu đến server thì có rất nhiều ngưởi có thể theo dõi, lấy dữ liệu. Vì vậy khi truyền các dữ liệu đã được mã hóa từ Client sẽ an toàn hơn.
  1. Theo dõi thêm các thao tác xóa và chỉnh sửa nhân viên
     1. Chỉnh sửa
* Chọn “Sửa” để sửa các trường cơ bản (trừ Mã NV)
* Chọn “Ghi/Lưu”



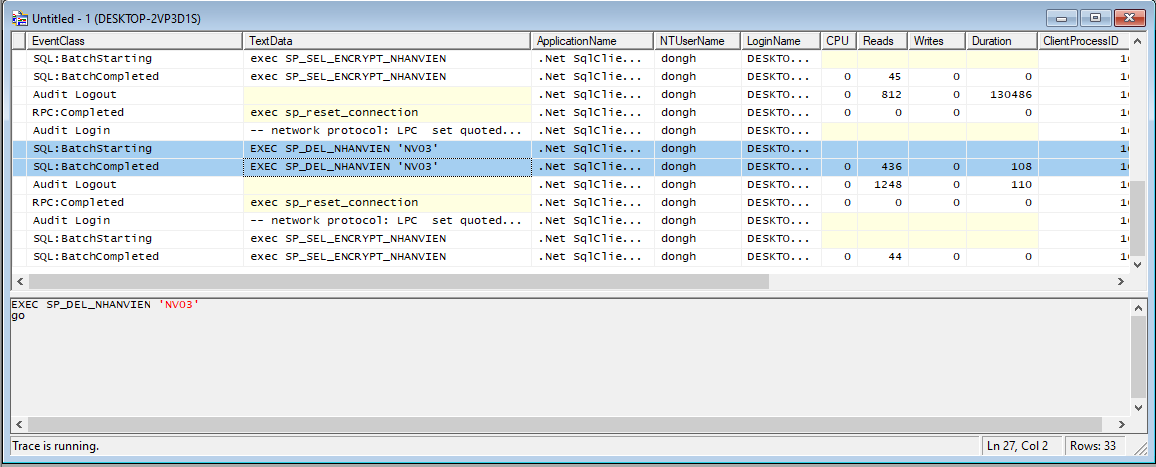
* Theo dõi kết quả ở SQL Profile



* + 1. Xóa nhân viên
* Chọn nhân viên muốn xóa và chọn “Xóa”



* Theo dõi kết quả ở SQL Profile



* Cũng giống như các câu lệnh ở trên, tất cả các thông tin đã được mã hóa ở Client trước khi truyền xuống dưới CSDL thông qua việc truyền vào Stored Procedure.