TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP HỒ CHÍ MINH

KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN   
Mã Hóa Ứng Dụng

Đề tài:

Mã hóa dữ liệu cho cơ sở dữ liệu “Quản lý sinh viên đơn giản”

GV hướng dẫn : Huỳnh Thanh Tâm

Sinh viên thực hiện: Nhóm 7

1. Nguyễn Huy Bảo – MSSV : 2033220316
2. Lý Gia Khang– MSSV : 2033221999

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | MSSV | Họ tên | Công việc được giao | Đánh giá |
| 1 | 2033220316 | Nguyễn Huy Bảo | Lab 1+2+word | Done(100%) |
| 2 | 2033221999 | Lý Gia Khang | Lab 1+2+word | Done(100%) |

Mục Lục

[Lời Mở Đầu 5](#_Toc169266863)

[Đề tài: **Cơ sở dữ liệu “Quản lý sinh viên đơn giản”** 5](#_Toc169266864)

[Cơ sở dữ liệu “Quản lý sinh viên đơn giản” LAB1 5](#_Toc169266865)

[Các Table 5](#_Toc169266866)

[Script tạo BANGDIEM 5](#_Toc169266867)

[Script tạo HOCPHAN 6](#_Toc169266868)

[Script tạo LOP 6](#_Toc169266869)

[Script tạo NHANVIEN 6](#_Toc169266870)

[Script tạo SINHVIEN 7](#_Toc169266871)

[Các dữ liệu được thêm vào 7](#_Toc169266872)

[Các Stored (Xóa) 9](#_Toc169266873)

[Các Stored (Thêm) 11](#_Toc169266874)

[Các Stored (Chọn) 14](#_Toc169266875)

[Các Stored (Cập Nhật) 18](#_Toc169266876)

[Stored (Đăng Nhập) 20](#_Toc169266877)

[Giao diện phần mềm quản lí CSDL “sinh viên” (LAB 1) 22](#_Toc169266878)

[TH1: Đăng nhập thành công chuyển đến các form khác 22](#_Toc169266879)

[Giao diện quản lý nhân viên 24](#_Toc169266880)

[Giao diện quản lý điểm sinh viên 28](#_Toc169266881)

[TH2: Không được sửa điểm sinh viên không thuộc quyền quản lý nhân viên 31](#_Toc169266882)

[Cơ sở dữ liệu “Quản lý sinh viên đơn giản” LAB2 32](#_Toc169266883)

[Các Table 32](#_Toc169266884)

[Script tạo HOCPHAN 32](#_Toc169266885)

[Script tạo LOP 33](#_Toc169266886)

[Script tạo SINHVIEN 33](#_Toc169266887)

[Các dữ liệu được thêm vào 33](#_Toc169266888)

[Các Stored (Xóa) 34](#_Toc169266889)

[Các Stored (Thêm) 36](#_Toc169266890)

[Các Stored (Chọn) 38](#_Toc169266891)

[Các Stored (Cập Nhật) 40](#_Toc169266892)

[Stored ( Đăng Nhập) 43](#_Toc169266893)

[Các hàm mã hóa liên quan 44](#_Toc169266894)

[Internal class RSA512 44](#_Toc169266895)

[RSACryptoServiceProvider rsa 44](#_Toc169266896)

[Private RSAParameters \_privateKey 44](#_Toc169266897)

[Hàm GetPrivateKey() 44](#_Toc169266898)

[Hàm GetPublicKey() 44](#_Toc169266899)

[Hàm so sánh RSA key 45](#_Toc169266900)

[Hàm so sánh 2 mảng byte (tương tự như so sánh RSA key) 45](#_Toc169266901)

[Hàm chuyển đổi XML về RSAParameters 45](#_Toc169266902)

[Hàm ghi key vào XML 46](#_Toc169266903)

[Hàm mã hóa RSA 47](#_Toc169266904)

[Hàm giải mã RSA 47](#_Toc169266905)

[Giao diện phần mềm quản lí CSDL “sinh viên” (LAB 2) 48](#_Toc169266906)

[TH1: Đăng nhập thành công chuyển đến các form khác 49](#_Toc169266907)

[TH2: Đăng nhập thất bại thông báo cho người dùng 51](#_Toc169266908)

[Giao diện quản lý nhân viên 51](#_Toc169266909)

[Giao diện quản lý lớp học 60](#_Toc169266910)

[Giao diện quản lý sinh viên 66](#_Toc169266911)

[Giao diện quản lý điểm sinh viên 74](#_Toc169266912)

[PHẦN KẾT 82](#_Toc169266913)

[Kết Luận 82](#_Toc169266914)

[Tài liệu tham khảo 83](#_Toc169266915)

[[DES] Thuật toán mã hóa và giải mã DES ~ VLSI TECHNOLOGY (nguyenquanicd.blogspot.com) 83](#_Toc169266916)

[[C#] Mã hóa RSA trong lập trình Csharp (laptrinhvb.net) 83](#_Toc169266917)

Lời Mở Đầu

Đề tài: **Cơ sở dữ liệu “Quản lý sinh viên đơn giản”**

- Mã hóa dữ liệu sử dụng các thuật toán mã hóa công khai

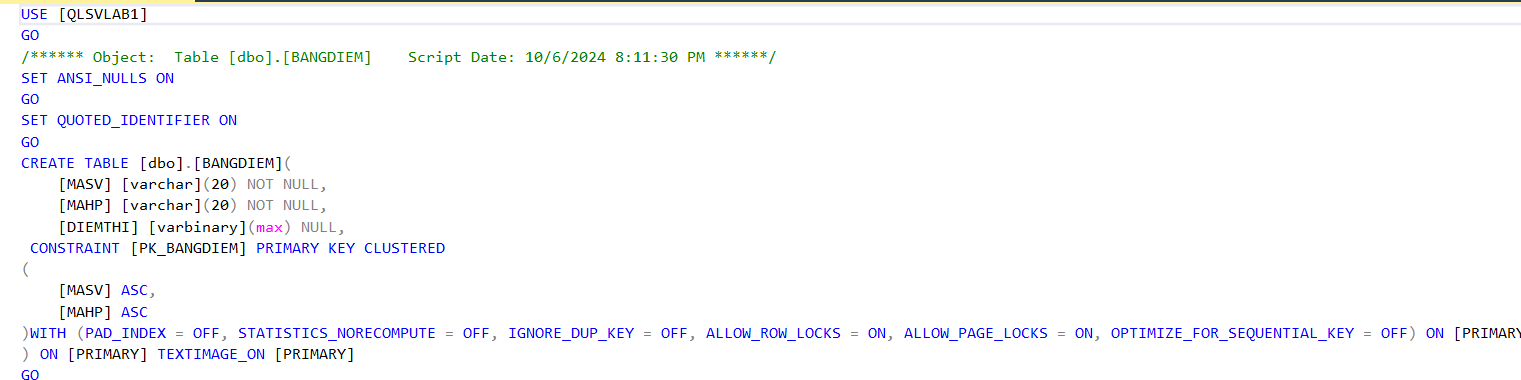
- Mã hóa dữ liệu từ client trước khi lưu xuống CSDL

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

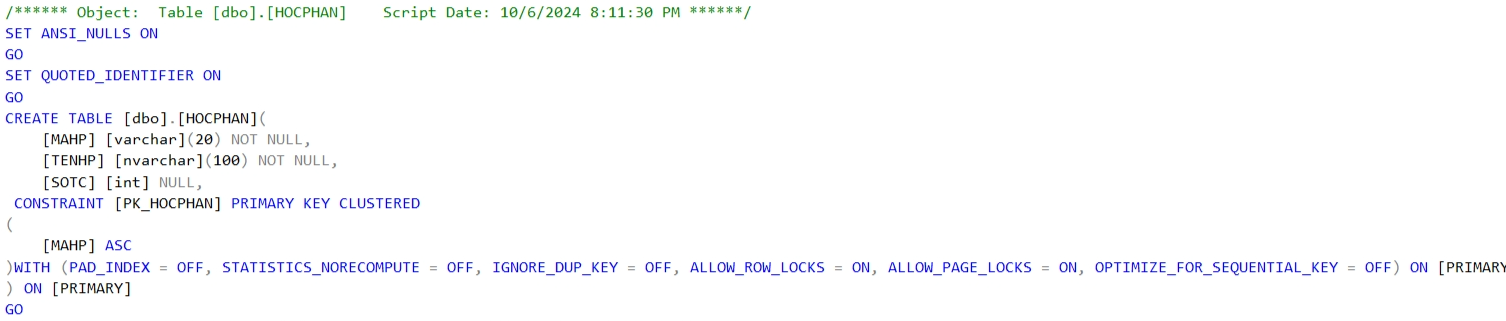
Cơ sở dữ liệu “Quản lý sinh viên đơn giản” LAB1

Các Table

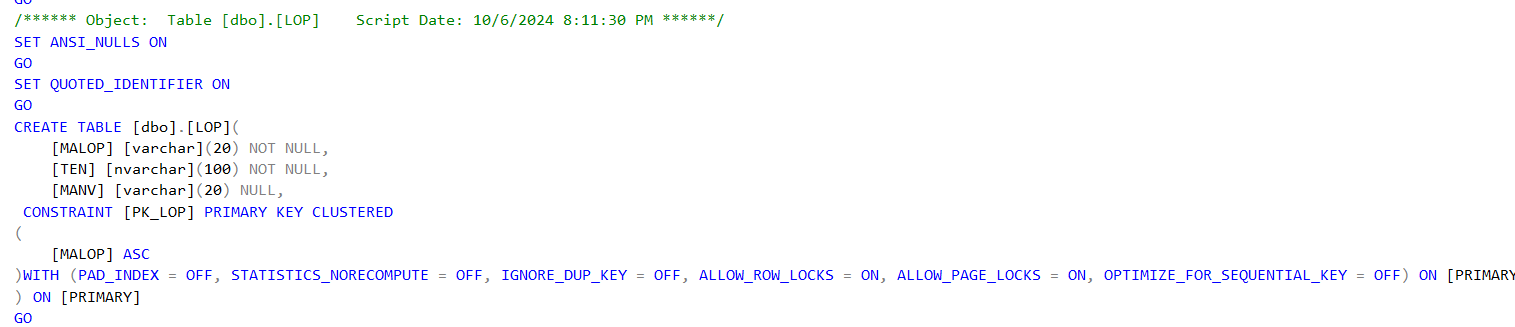
Script tạo BANGDIEM



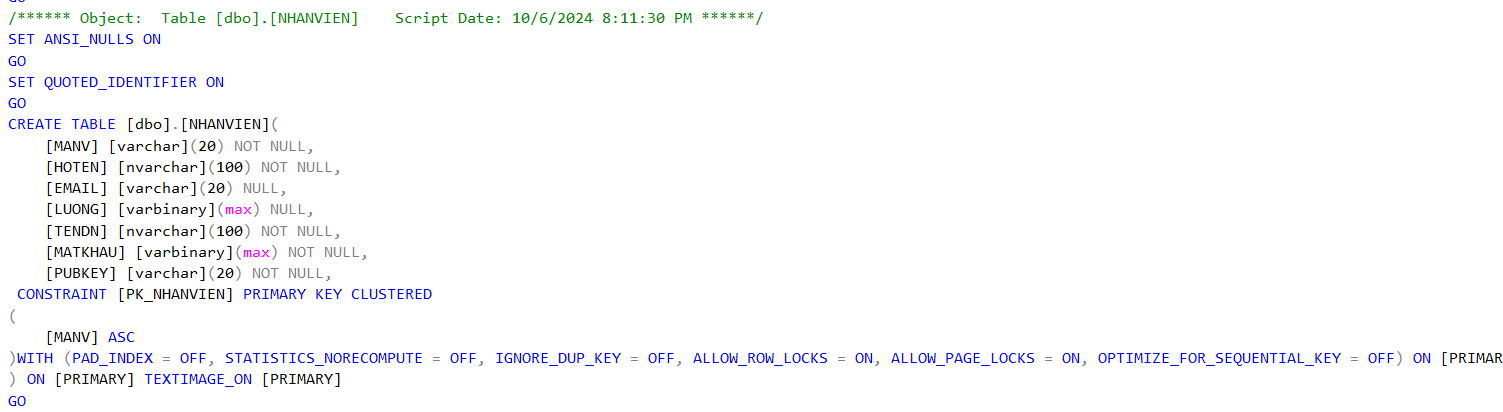
Script tạo HOCPHAN



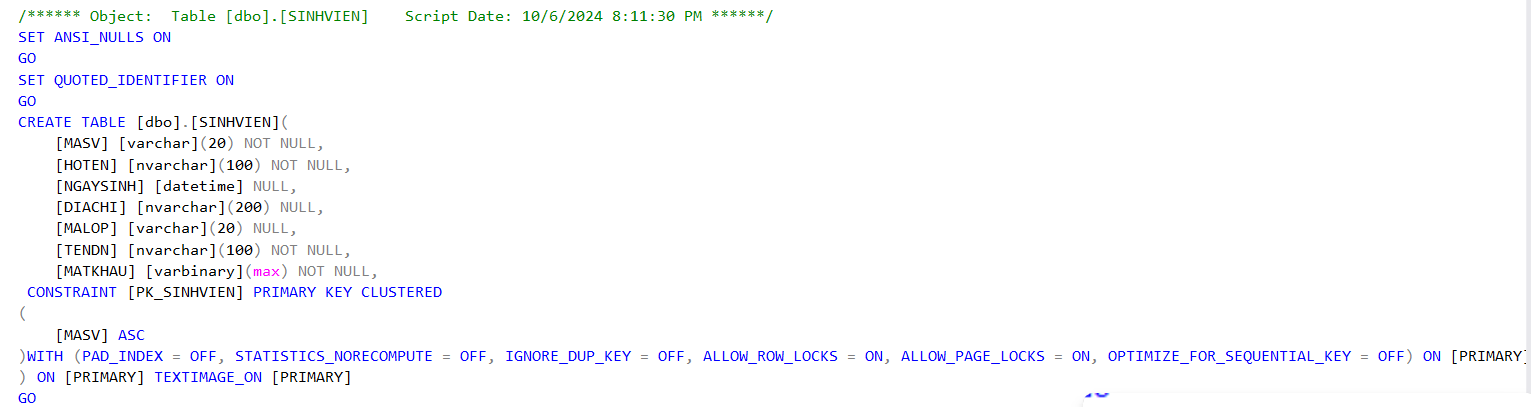
Script tạo LOP



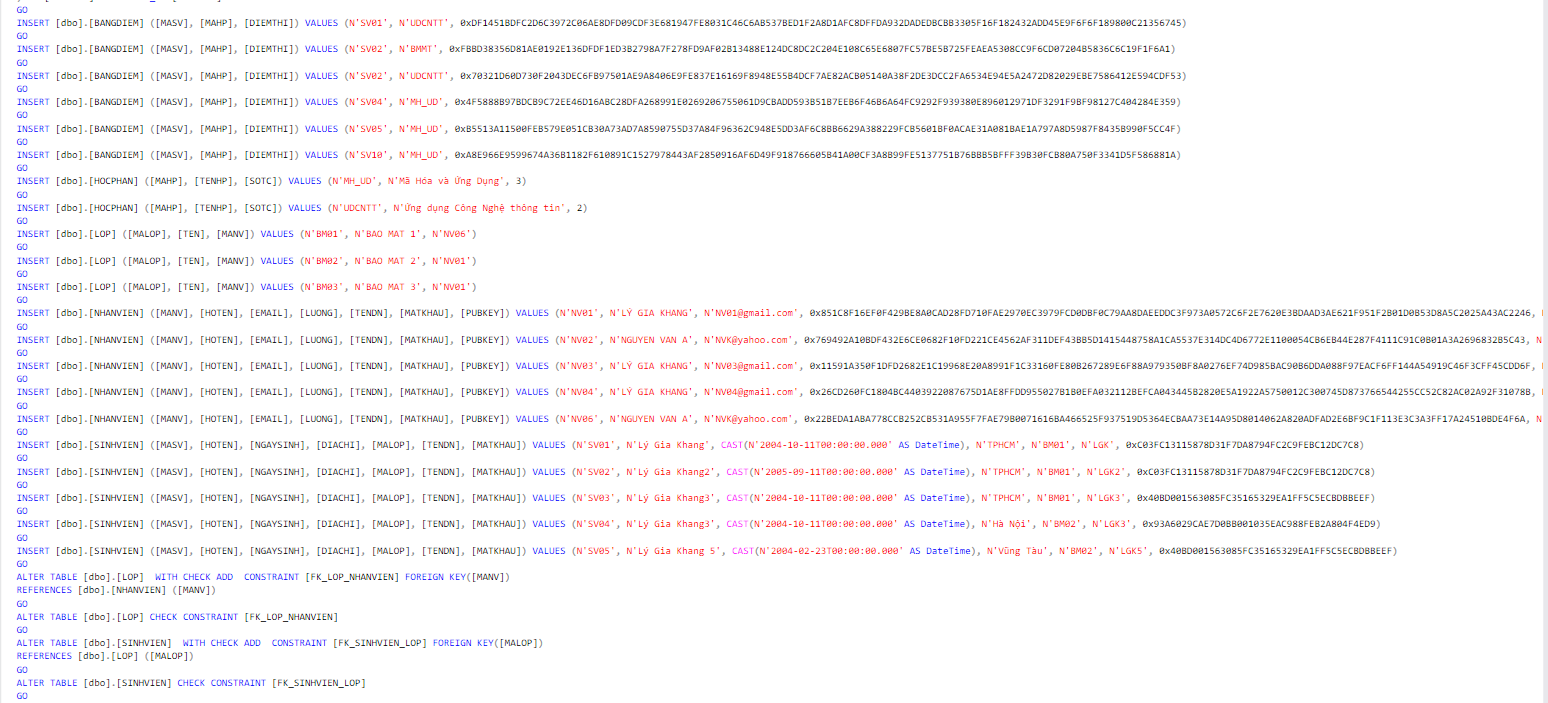
Script tạo NHANVIEN



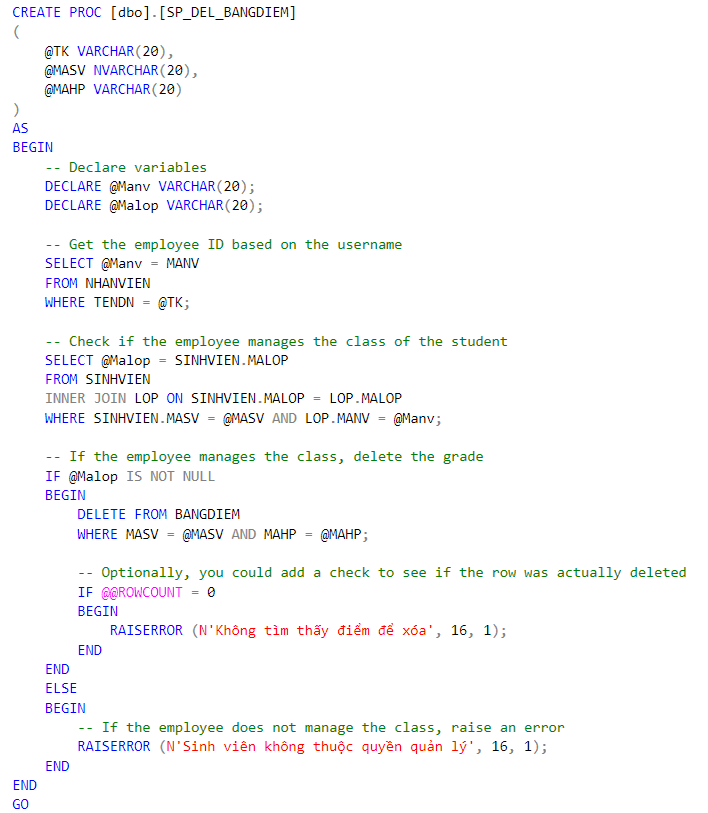
Script tạo SINHVIEN



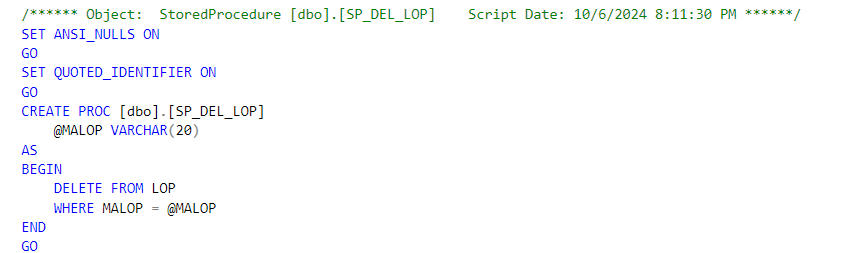
Các dữ liệu được thêm vào



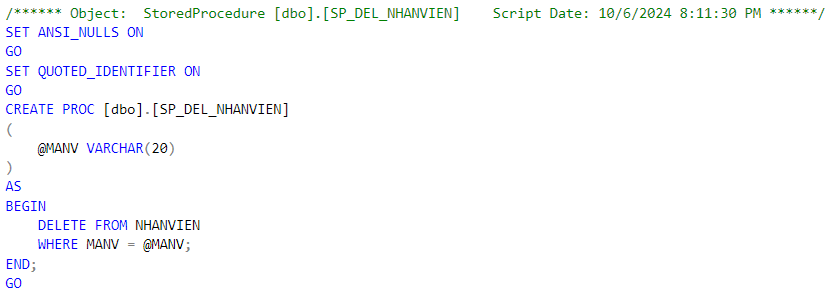
Các Stored (Xóa)



Stored DEL\_BANGDIEM dùng để xóa dữ liệu Điểm



Stored DEL\_LOP dùng để xóa dữ liệu LOP

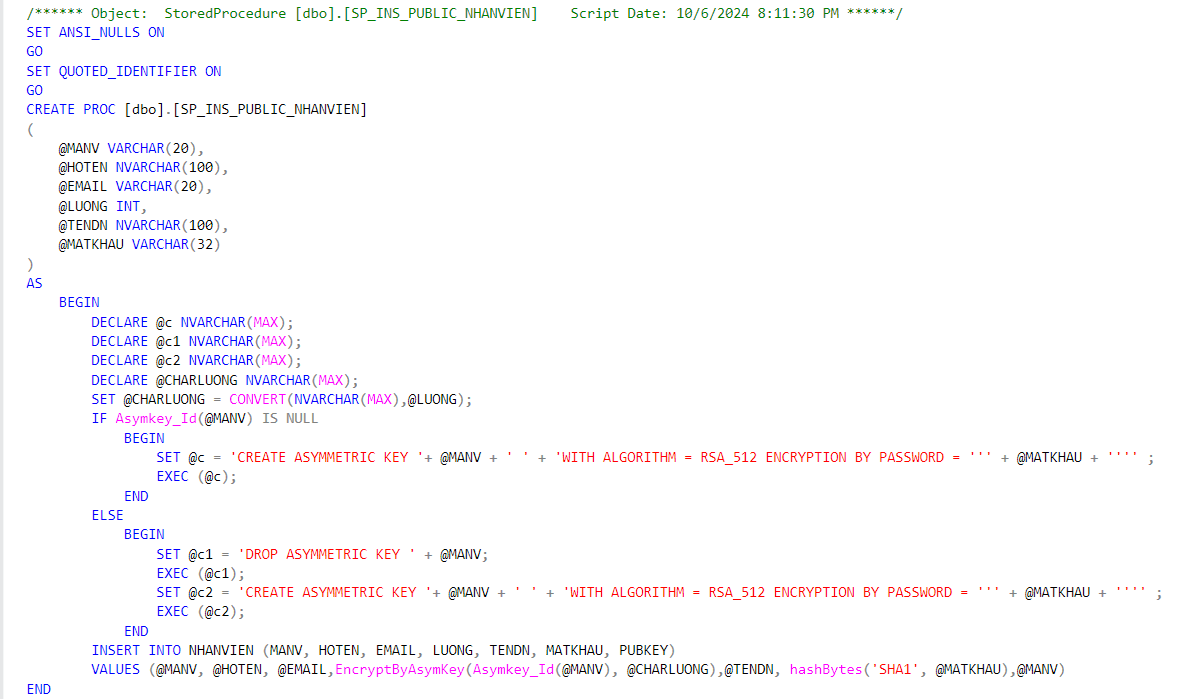


Stored DEL\_SINHVIEN dùng để xóa dữ liệu SINHVIEN

Các Stored (Thêm)



Stored INS\_BANGDIEM dùng để thêm điểm



**Tham Số**:

@MANV VARCHAR(20): Mã nhân viên.

@HOTEN NVARCHAR(100): Họ tên nhân viên.

@EMAIL VARCHAR(20): Email nhân viên.

@LUONG INT: Lương của nhân viên.

@TENDN NVARCHAR(100): Tên đăng nhập của nhân viên.

@MATKHAU VARCHAR(32): Mật khẩu của nhân viên.

**Khai Báo Biến**:

@c, @c1, @c2 (NVARCHAR(MAX)): Chuỗi dùng để lưu các câu lệnh SQL động.

@CHARLUONG NVARCHAR(MAX): Chuỗi dùng để lưu lương đã được chuyển đổi.

**Chuyển Đổi Lương**:

Lương @LUONG dạng số nguyên được chuyển thành chuỗi và lưu trong @CHARLUONG.

**Kiểm Tra Khóa Bất Đối Xứng**:

Thủ tục kiểm tra nếu một khóa bất đối xứng với tên @MANV đã tồn tại bằng cách sử dụng Asymkey\_Id(@MANV).

**Tạo Hoặc Tạo Lại Khóa Bất Đối Xứng**:

Nếu khóa chưa tồn tại, thủ tục sẽ tạo một khóa bất đối xứng mới sử dụng thuật toán RSA\_512 với mật khẩu cung cấp @MATKHAU.

Nếu khóa đã tồn tại, thủ tục sẽ xóa khóa hiện tại và tạo lại nó với cùng các thông số.

**Thêm Bản Ghi Nhân Viên**:

Thêm một bản ghi mới vào bảng NHANVIEN với các tham số được cung cấp:

MANV, HOTEN, EMAIL: Trực tiếp từ các tham số đầu vào.

LUONG: Lương được mã hóa bằng khóa bất đối xứng.

TENDN: Tên đăng nhập từ tham số đầu vào.

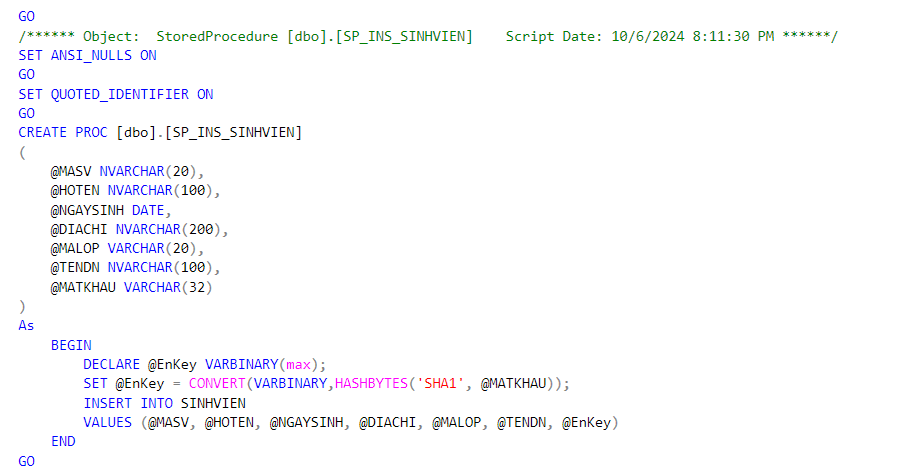
MATKHAU: Mật khẩu được băm bằng SHA1.

PUBKEY: Mã nhân viên được sử dụng làm khóa công khai.

Tóm Tắt Các Hoạt Động Chính:

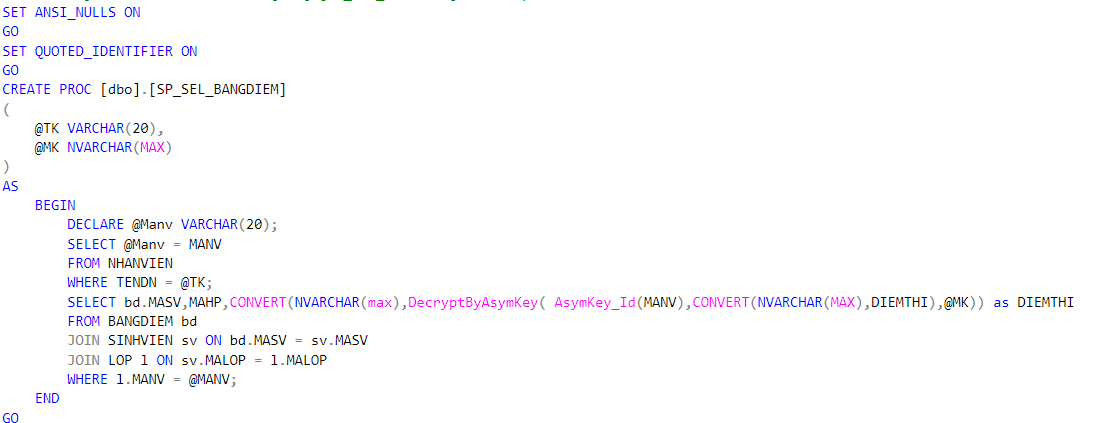
**Thực Thi SQL Động**: Thủ tục này tạo ra các câu lệnh SQL động để quản lý khóa bất đối xứng.

**Mã Hóa và Băm**: Sử dụng mã hóa bất đối xứng cho lương và băm cho mật khẩu để đảm bảo an ninh dữ liệu.



Stored INS\_SINHVIEN sử dụng SHA1 để mã khóa @MATKHAU đăng nhập và thêm SINHVIEN.

Các Stored (Chọn)



Stored SEL\_BANGDIEM dùng để chọn DIEM từ CSDL

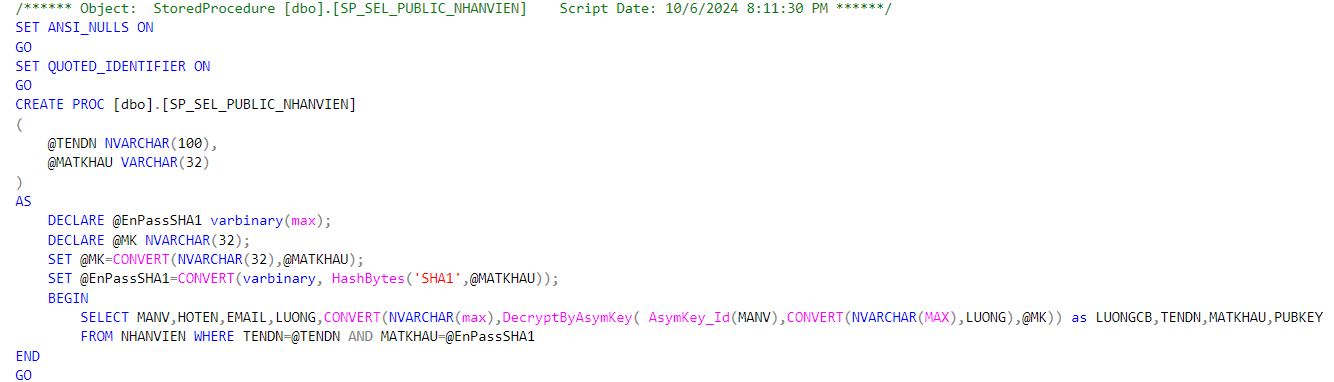
1. **Tham Số**:
   * @TK VARCHAR(20): Tài khoản (tên đăng nhập) của nhân viên.
   * @MK NVARCHAR(MAX): Mật khẩu của nhân viên.
2. **Khai Báo Biến**:
   * @Manv VARCHAR(20): Mã nhân viên.
3. **Lấy Mã Nhân Viên**:
   * Sử dụng câu lệnh SELECT để lấy MANV từ bảng NHANVIEN dựa trên tên đăng nhập @TK.
4. **Lấy Thông Tin Bảng Điểm**:
   * Sử dụng câu lệnh SELECT để lấy thông tin bảng điểm của sinh viên:
     + MASV: Mã sinh viên.
     + MAHP: Mã học phần.
     + DIEMTHI: Điểm thi của sinh viên, được giải mã bằng khóa bất đối xứng.
   * Kết nối bảng BANGDIEM với bảng SINHVIEN và bảng LOP để lấy thông tin bảng điểm cho sinh viên thuộc lớp của nhân viên có mã @Manv.

Tóm Tắt Các Hoạt Động Chính:

* **Xác Định Mã Nhân Viên**: Lấy mã nhân viên dựa trên tài khoản đăng nhập.
* **Giải Mã Điểm Thi**: Sử dụng khóa bất đối xứng và mật khẩu để giải mã điểm thi.
* **Kết Nối Bảng**: Kết nối các bảng BANGDIEM, SINHVIEN, và LOP để lấy thông tin chi tiết.

Giải Thích Các Phần Chính:

* **SELECT @Manv = MANV FROM NHANVIEN WHERE TENDN = @TK**: Lấy mã nhân viên dựa trên tên đăng nhập.
* **SELECT bd.MASV, MAHP, CONVERT(NVARCHAR(max), DecryptByAsymKey(AsymKey\_Id(MANV), CONVERT(NVARCHAR(MAX), DIEMTHI), @MK)) as DIEMTHI**:
  + MASV, MAHP: Trực tiếp từ bảng BANGDIEM.
  + DIEMTHI: Điểm thi được giải mã bằng khóa bất đối xứng và mật khẩu.
* **JOIN SINHVIEN sv ON bd.MASV = sv.MASV JOIN LOP l ON sv.MALOP = l.MALOP WHERE l.MANV = @MANV**: Kết nối các bảng để lấy thông tin bảng điểm của sinh viên thuộc lớp của nhân viên.



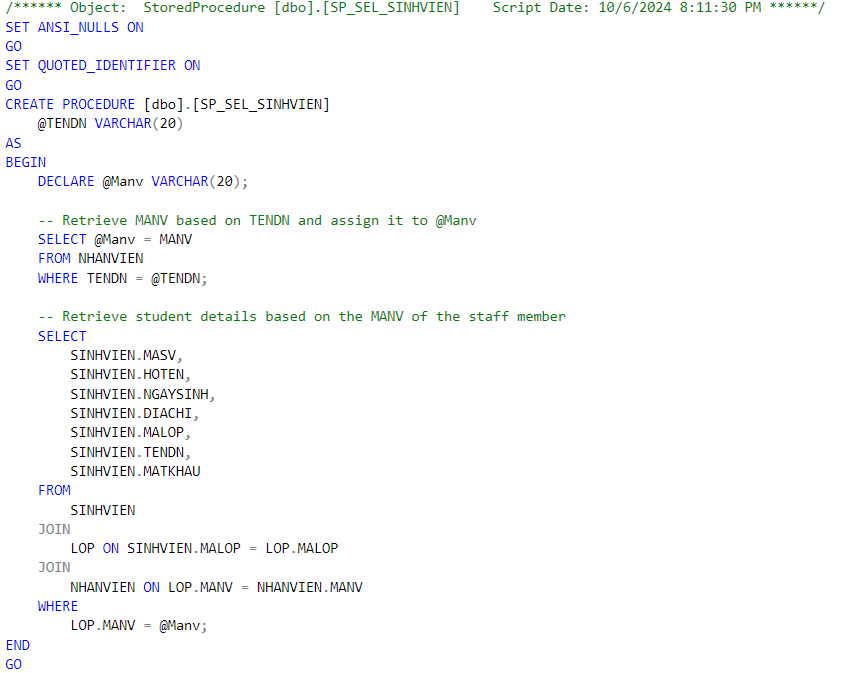
1. **Tham Số**:
   * @TENDN NVARCHAR(100): Tên đăng nhập của nhân viên.
   * @MATKHAU VARCHAR(32): Mật khẩu của nhân viên.
2. **Khai Báo Biến**:
   * @EnPassSHA1 varbinary(max): Mật khẩu được băm bằng SHA1.
   * @MK NVARCHAR(32): Mật khẩu dạng chuỗi ký tự.
3. **Chuyển Đổi Mật Khẩu**:
   * @MK: Chuyển đổi mật khẩu @MATKHAU thành chuỗi ký tự NVARCHAR.
   * @EnPassSHA1: Chuyển đổi mật khẩu @MATKHAU thành dạng băm bằng SHA1 và lưu dưới dạng varbinary.
4. **Lấy Thông Tin Nhân Viên**:
   * Sử dụng câu lệnh SELECT để lấy thông tin nhân viên từ bảng NHANVIEN dựa trên tên đăng nhập @TENDN và mật khẩu đã băm @EnPassSHA1.
   * Các thông tin bao gồm:
     + MANV: Mã nhân viên.
     + HOTEN: Họ tên nhân viên.
     + EMAIL: Email nhân viên.
     + LUONG: Lương được mã hóa.
     + LUONGCB: Lương cơ bản, được giải mã bằng khóa bất đối xứng và mật khẩu.
     + TENDN: Tên đăng nhập.
     + MATKHAU: Mật khẩu đã băm.
     + PUBKEY: Khóa công khai.

Tóm Tắt Các Hoạt Động Chính:

* **Chuyển Đổi và Băm Mật Khẩu**: Chuyển đổi mật khẩu thành dạng chuỗi ký tự NVARCHAR và băm bằng SHA1.
* **Giải Mã Lương**: Sử dụng khóa bất đối xứng và mật khẩu để giải mã lương.
* **Lấy Thông Tin Nhân Viên**: Lấy thông tin chi tiết của nhân viên từ bảng NHANVIEN.

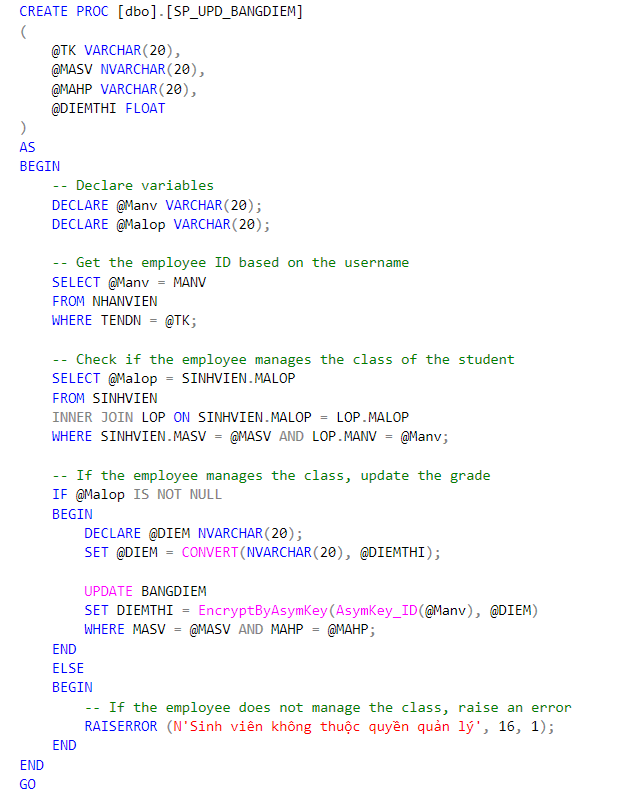
Giải Thích Các Phần Chính:

* **SET @MK=CONVERT(NVARCHAR(32),@MATKHAU)**: Chuyển đổi mật khẩu thành dạng chuỗi ký tự.
* **SET @EnPassSHA1=CONVERT(varbinary, HashBytes('SHA1',@MATKHAU))**: Băm mật khẩu bằng SHA1 và chuyển đổi thành varbinary.
* **SELECT MANV, HOTEN, EMAIL, LUONG, CONVERT(NVARCHAR(max), DecryptByAsymKey(AsymKey\_Id(MANV), CONVERT(NVARCHAR(MAX), LUONG), @MK)) as LUONGCB, TENDN, MATKHAU, PUBKEY**:
  + Lấy các thông tin chi tiết của nhân viên, bao gồm lương cơ bản đã được giải mã bằng khóa bất đối xứng.

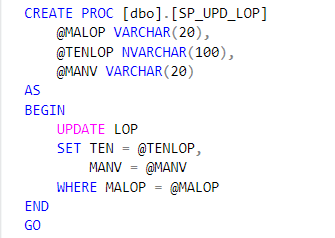


Stored SEL\_SINHVIEN dùng để chọn SINHVIEN

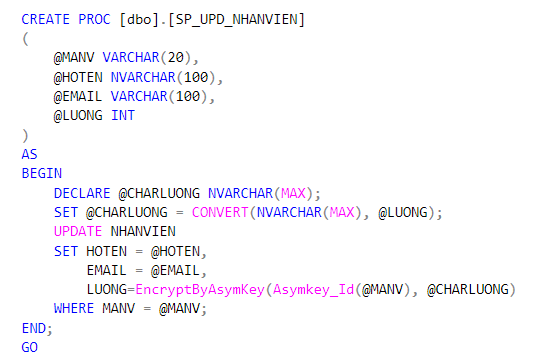
Các Stored (Cập Nhật)



Stored UPD\_BANGDIEM dùng để cập nhật DIEM



Stored UPD\_LOP dùng để cập nhật LOP

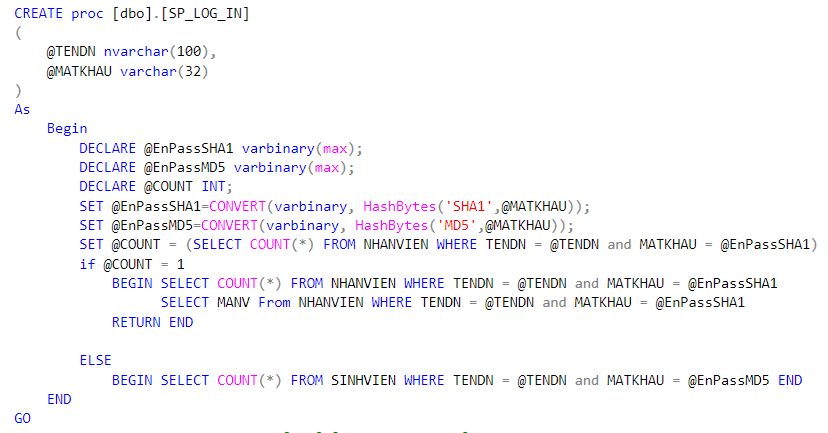


Stored UPD\_NHANVIEN dùng để cập nhật NHANVIEN và giải mã LUONG dựa trên @MANV



Stored dùng để sửa thông tin SINHVIEN

Stored (Đăng Nhập)



Dùng để đăng nhập vào Phần mềm

Giao diện phần mềm quản lí CSDL “sinh viên” (LAB 1)

TH1: Đăng nhập thành công chuyển đến các form khác

A screenshot of a computer screen

Description automatically generated

SQL Profiler

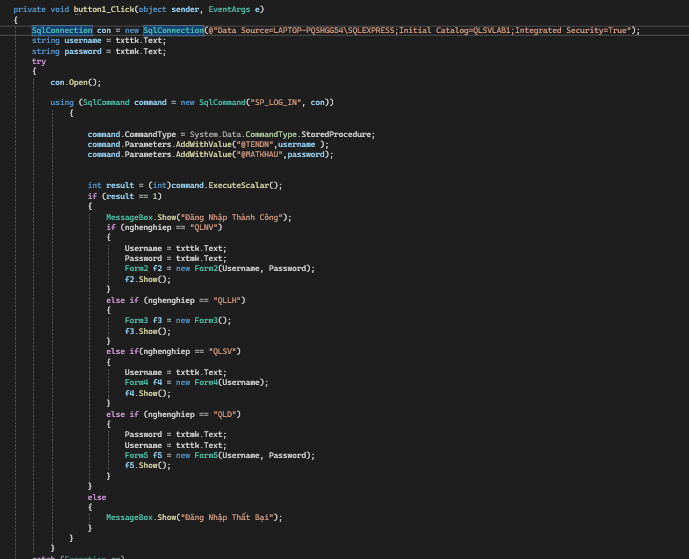


exec SP\_LOG\_IN @TENDN=N'NVK',@MATKHAU=N'1'

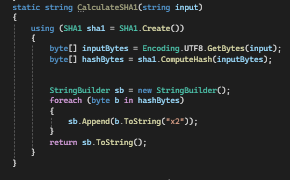


Code

Hàm xử lý đăng nhập



Hàm mã hóa SHA cho mật khẩu



Giao diện quản lý nhân viên

A screenshot of a computer

Description automatically generated

Thêm nhân viên

A screenshot of a computer

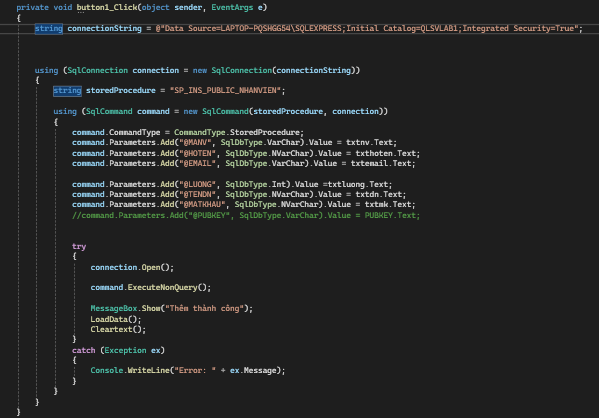
Description automatically generated

SQL Profiler



exec SP\_INS\_PUBLIC\_NHANVIEN @MANV='NV07',@HOTEN=N'NGUYEN VAN A7',@EMAIL='NVK7@yahoo.com',@LUONG=1000000,@TENDN=N'NVK7',@MATKHAU=N'khang2222'

Code



1. Kết nối CSDL
2. Gọi Stored
3. Nhập dữ liệu tương ứng và thêm

Dữ liệu hiển thị phía CSDL



Sửa thông tin nhân viên

A screenshot of a computer

Description automatically generated

Nhấn nút 

A screenshot of a computer

Description automatically generated

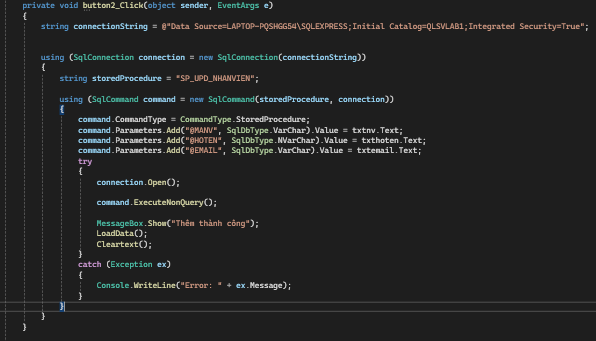
Sửa thành công

SQL Profiler



exec SP\_UPD\_NHANVIEN @MANV='NV07',@HOTEN=N'NGUYEN VAN A7',@EMAIL='NVK7@yahoo.com',@LUONG=123

Code



1. Kết nối đến CSDL
2. Gọi stored cập nhật
3. Nhập dữ liệu tương ứng

Dữ liệu hiển thị phía CSDL



Giao diện quản lý điểm sinh viên

A screenshot of a computer

Description automatically generated

Thêm nhân viên

A screenshot of a computer

Description automatically generated

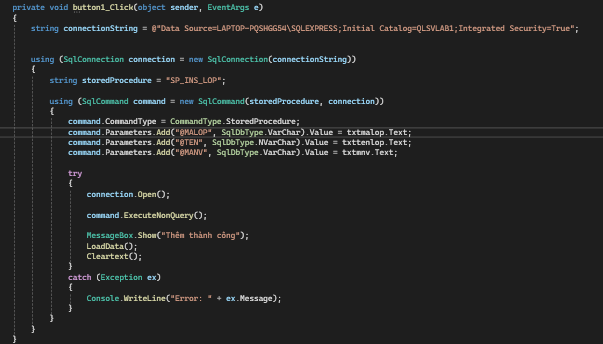
Nhấn nút 

SQL Profiler



exec SP\_INS\_BANGDIEM @TK='NVK',@MASV=N'SV01',@MAHP=' BMMT',@DIEMTHI=9.1999999999999993

Code



1. Kết nối đến CSDL
2. Gọi stored cập nhật
3. Nhập dữ liệu tương ứng

Dữ liệu hiển thị phía CSDL



Sửa nhân viên

A screenshot of a computer

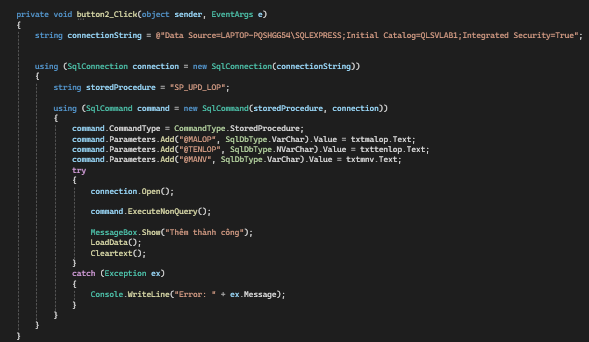
Description automatically generated

Nhấn nút 

SQL Profiler



Code



1. Kết nối đến CSDL
2. Gọi stored cập nhật
3. Nhập dữ liệu tương ứng

Dữ liệu hiển thị phía CSDL



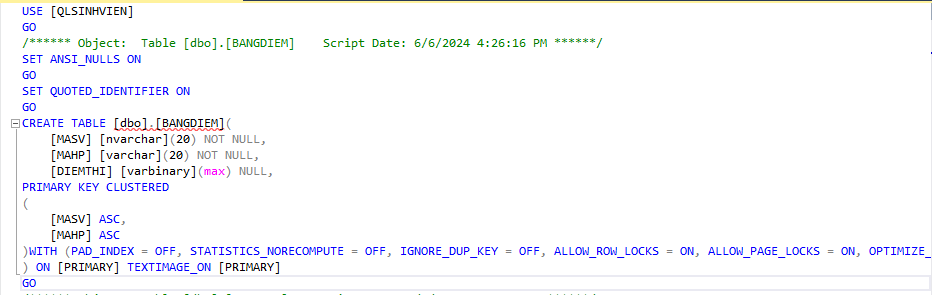
TH2: Không được sửa điểm sinh viên không thuộc quyền quản lý nhân viên

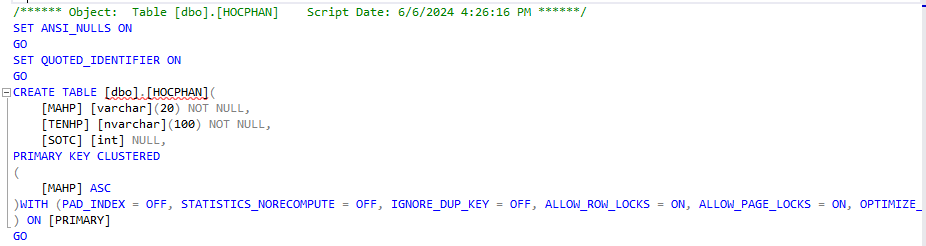
A screenshot of a computer

Description automatically generated

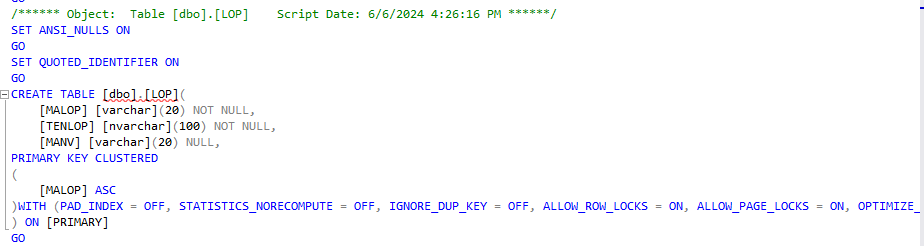
Cơ sở dữ liệu “Quản lý sinh viên đơn giản” LAB2

Các Table

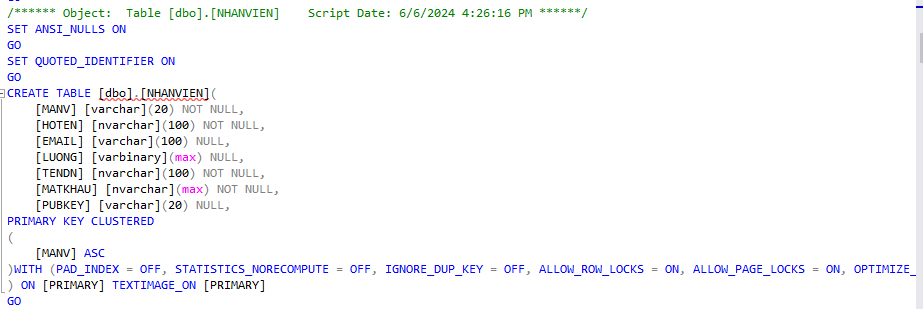


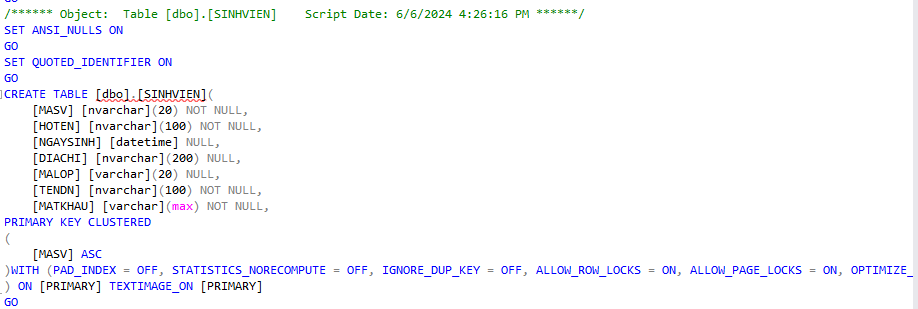
Script tạo BANGDIEM

Script tạo HOCPHAN



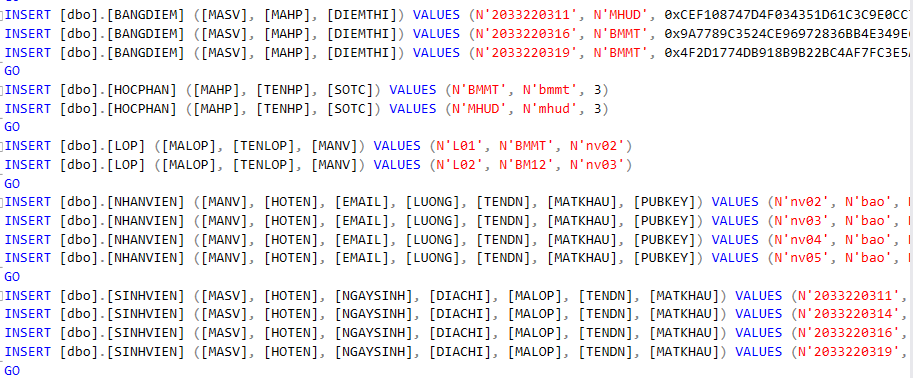
Script tạo LOP



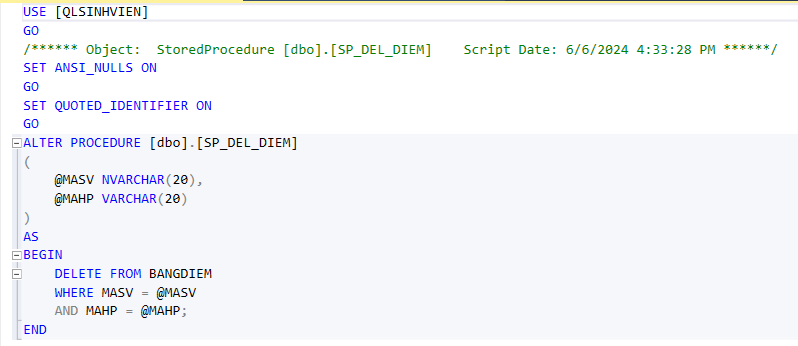
Script tạo NHANVIEN

Script tạo SINHVIEN

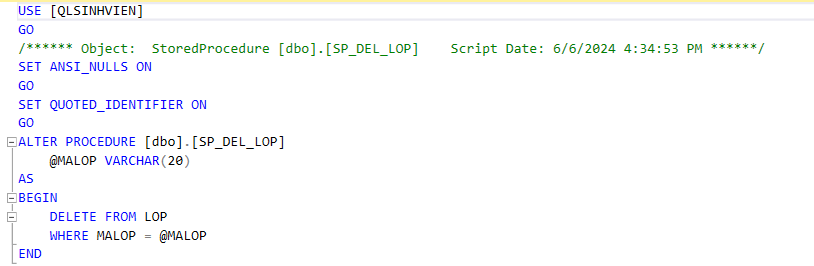
Các dữ liệu được thêm vào



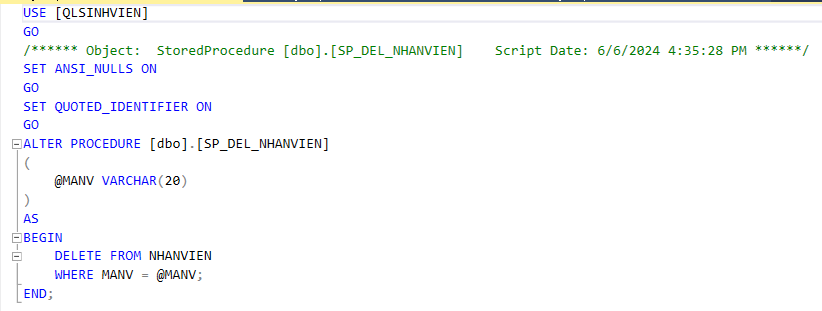
Các Stored (Xóa)



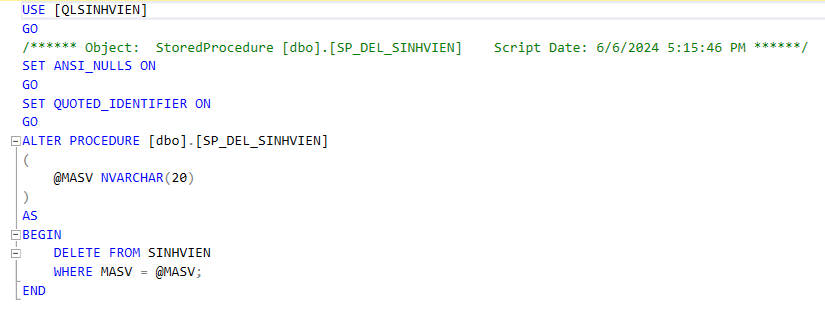
Stored DEL\_DIEM dùng để xóa dữ liệu Điểm



Stored DEL\_DIEM dùng để xóa dữ liệu LOP

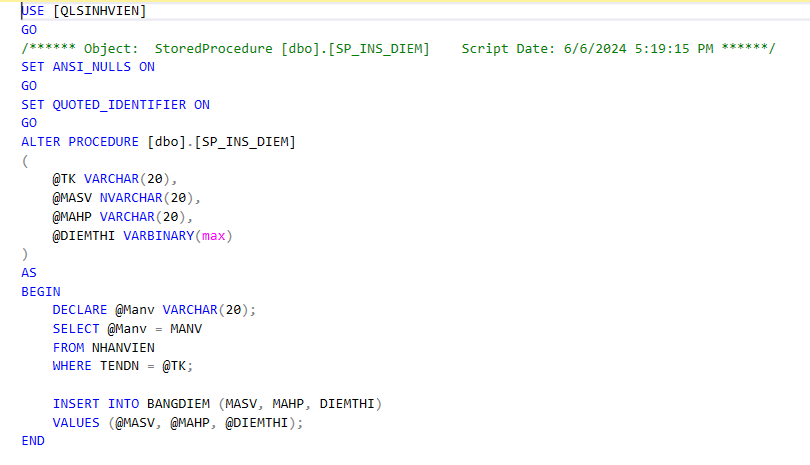


Stored DEL\_DIEM dùng để xóa dữ liệu NHANVIEN

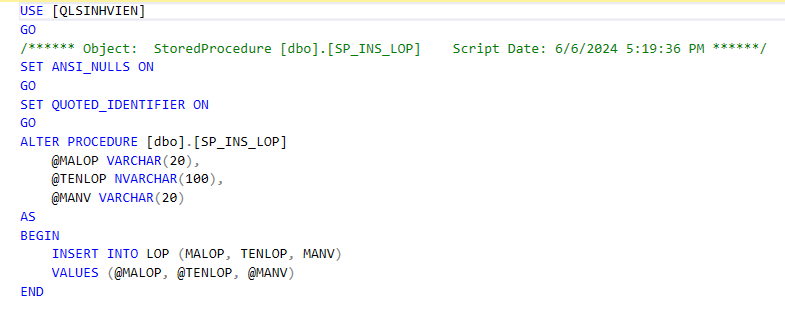


Stored DEL\_DIEM dùng để xóa dữ liệu SVIEN

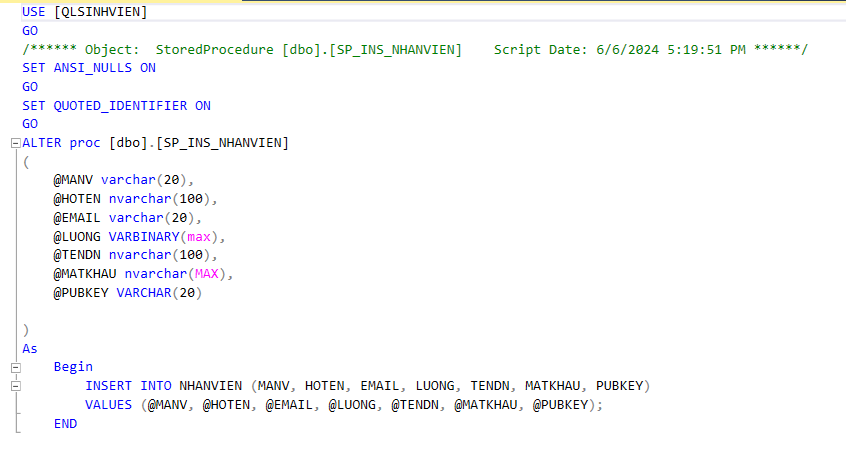
Các Stored (Thêm)



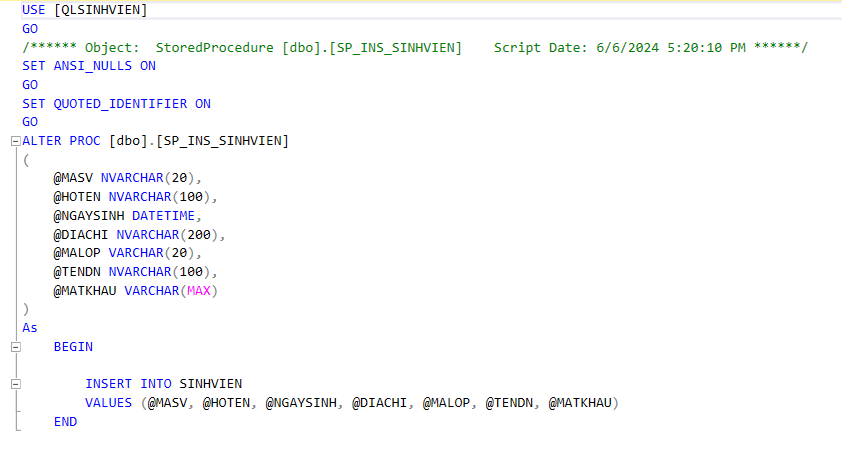
Stored INS\_DIEM dùng để thêm dữ liệu DIEM



Stored INS\_DIEM dùng để thêm dữ liệu LOP

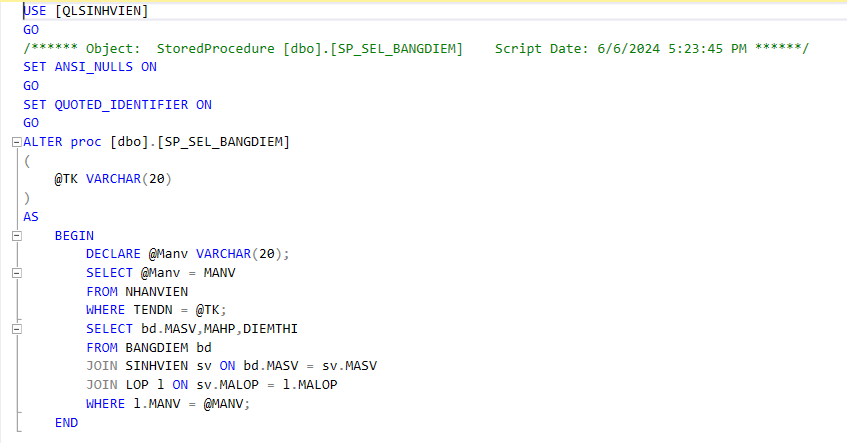


Stored INS\_DIEM dùng để thêm dữ liệu NHANVIEN

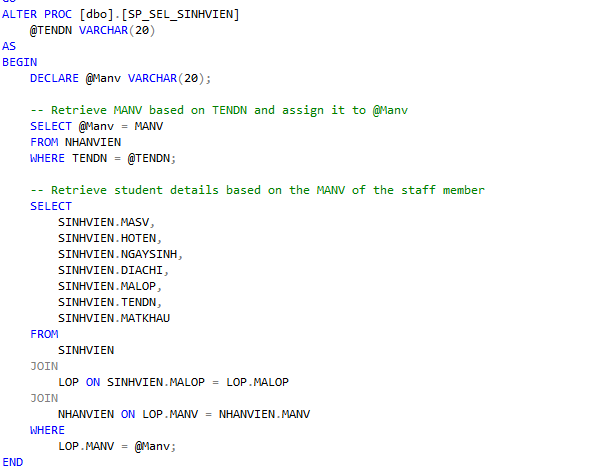


Stored INS\_DIEM dùng để thêm dữ liệu SINHVIEN

Các Stored (Chọn)

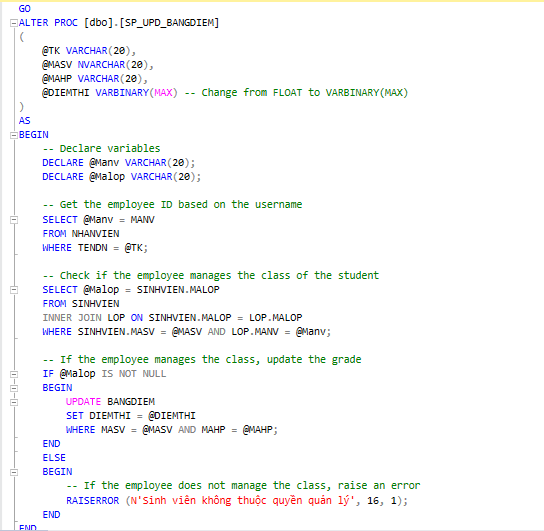


Stored SEL\_BANGDIEM dùng để xem dữ liệu SINHVIEN

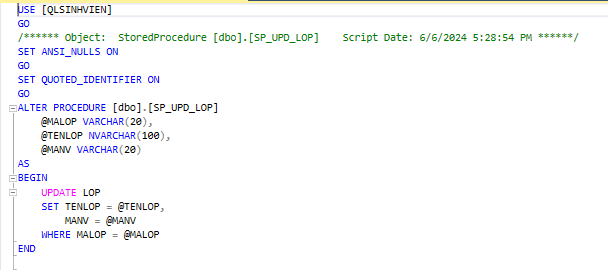


Stored SEL\_SINHVIEN dùng để xem dữ liệu SINHVIEN

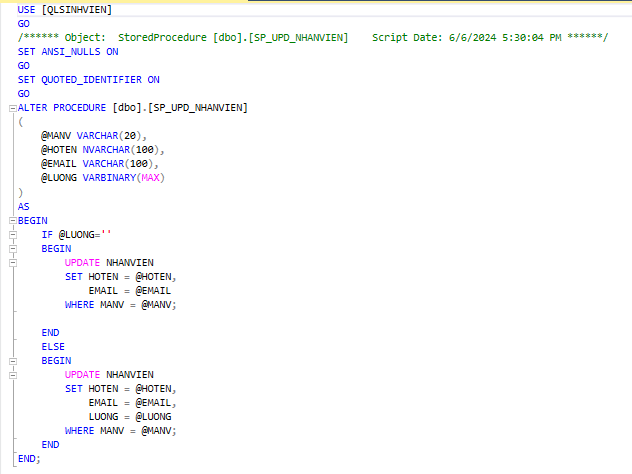
Các Stored (Cập Nhật)



Stored UPD\_BANGDIEM dùng để cập nhật dữ liệu DIEM của SINHVIEN (Chỉ được chỉnh sửa điểm của sinh viên mà nhân viên quản lí)



Stored UPD\_LOP dùng để cập nhật dữ liệu của LOP



Stored UPD\_NHANVIEN dùng để cập nhật dữ liệu của NHANVIEN



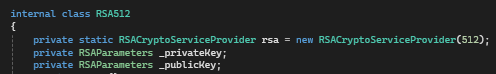
Stored UPD\_SINHVIEN dùng để cập nhật dữ liệu của SINHVIEN

Stored ( Đăng Nhập)



Stored LOG\_IN dùng để đăng nhập vào CSDL

Các hàm mã hóa liên quan



Internal class RSA512

* Khai báo một lớp có tên RSA512 với phạm vi truy cập nội bộ, nghĩa là nó chỉ có thể truy cập được trong cùng một assembly.

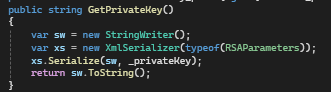
RSACryptoServiceProvider rsa

* Tạo một đối tượng tĩnh của lớp RSACryptoServiceProvider, được khởi tạo với kích thước khóa là 512 bit. Lớp này cung cấp các phương thức để mã hóa, giải mã và tạo khóa sử dụng thuật toán RSA.

Private RSAParameters \_privateKey

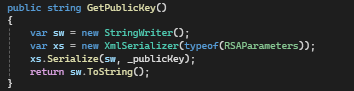
* Constructor khởi tạo trường \_privateKey bằng cách xuất các tham số RSA từ đối tượng RSACryptoServiceProvider.

Hàm GetPrivateKey()



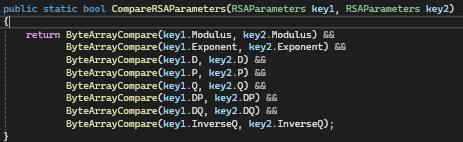
* StringWriter được sử dụng để lưu trữ dữ liệu XML đã được tuần tự hóa.
* XmlSerializer được sử dụng để tuần tự hóa đối tượng RSAParameters thành định dạng XML.
* Các tham số khóa riêng tư được tuần tự hóa và trả về dưới dạng chuỗi XML.

Hàm GetPublicKey()



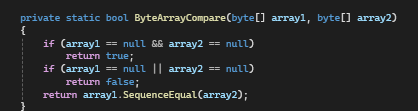
* Tương tự GetPrivateKey cung cấp khóa công khai ở định dạng dễ dàng chia sẻ hoặc lưu trữ.

Hàm so sánh RSA key



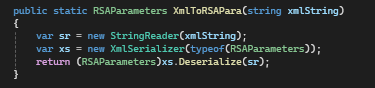
* So sánh hai đối tượng kiểu RSAParameters bằng cách kiểm tra từng thuộc tính của chúng. RSAParameters là một cấu trúc trong .NET Framework được sử dụng để lưu trữ các tham số của thuật toán RSA (Rivest-Shamir-Adleman)
* Phương thức ByteArrayCompare được sử dụng để so sánh các mảng byte của các thuộc tính này. Nếu tất cả các thuộc tính đều bằng nhau, phương thức CompareRSAParameters sẽ trả về true, ngược lại sẽ trả về false.

Hàm so sánh 2 mảng byte (tương tự như so sánh RSA key)



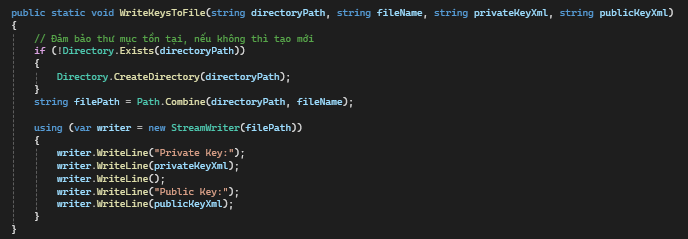
* Hàm ByteArrayCompare được sử dụng để kiểm tra xem hai mảng byte có bằng nhau hay không, bao gồm việc xử lý các trường hợp mảng null và so sánh nội dung của các mảng không null. Hàm này hỗ trợ phương thức CompareRSAParameters trong việc so sánh các thuộc tính của các đối tượng RSAParameters.

Hàm chuyển đổi XML về RSAParameters



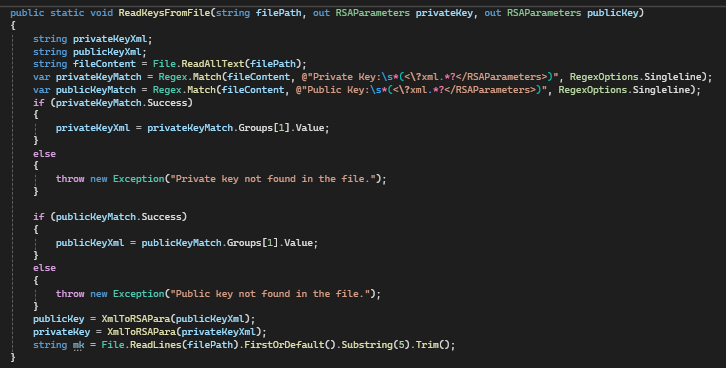
* Hàm XmlToRSAPara nhận một chuỗi XML chứa thông tin về các tham số của một khóa RSA. Nó sử dụng XmlSerializer để chuyển đổi chuỗi XML này thành một đối tượng RSAParameters, sau đó trả về đối tượng.

Hàm ghi key vào XML



* Hàm WriteKeysToFile nhận đầu vào là đường dẫn thư mục (directoryPath), tên tệp tin (fileName), cùng hai chuỗi XML chứa khóa riêng tư và khóa công khai của thuật toán RSA (privateKeyXml và publicKeyXml).
* Ghi thông tin về khóa riêng tư và khóa công khai vào một tệp tin văn bản tại đường dẫn đã chỉ định. Nếu thư mục không tồn tại, nó sẽ tạo mới thư mục đó.
* Sau đó, hàm sẽ ghi vào tệp tin:
* Dòng "Private Key:" theo sau là chuỗi XML của khóa riêng tư.
* Dòng "Public Key:" theo sau là chuỗi XML của khóa công khai.

Hàm đọc KEY từ file XML



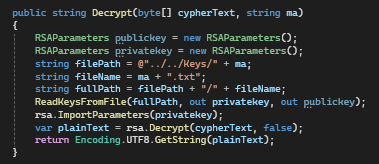
* Đọc khóa từ file XML lấy key công khai và bí mật đã tạo để giải mã và mã hóa
* Sử dụng biểu thức chính quy để tìm chuỗi XML của khóa riêng tư và khóa công khai trong nội dung tệp tin.
* Nếu khóa riêng tư hoặc khóa công khai không được tìm thấy, hàm sẽ ném ra một ngoại lệ.
* Nếu tìm thấy, hàm sẽ chuyển đổi chuỗi XML thành đối tượng RSAParameters sử dụng hàm XmlToRSAPara.
* Cuối cùng, hàm trả về khóa riêng tư và khóa công khai dưới dạng đối tượng RSAParameters, được truyền ra ngoài hàm thông qua tham số out.

Hàm mã hóa RSA



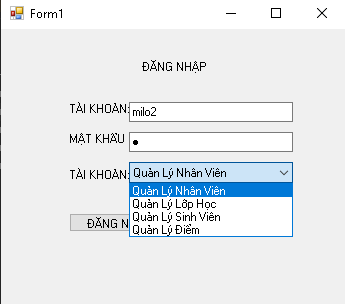
* Hàm RSAEncrypt sử dụng khóa công khai để mã hóa một chuỗi văn bản bằng thuật toán RSA và trả về dữ liệu đã được mã hóa dưới dạng một mảng byte.

Hàm giải mã RSA

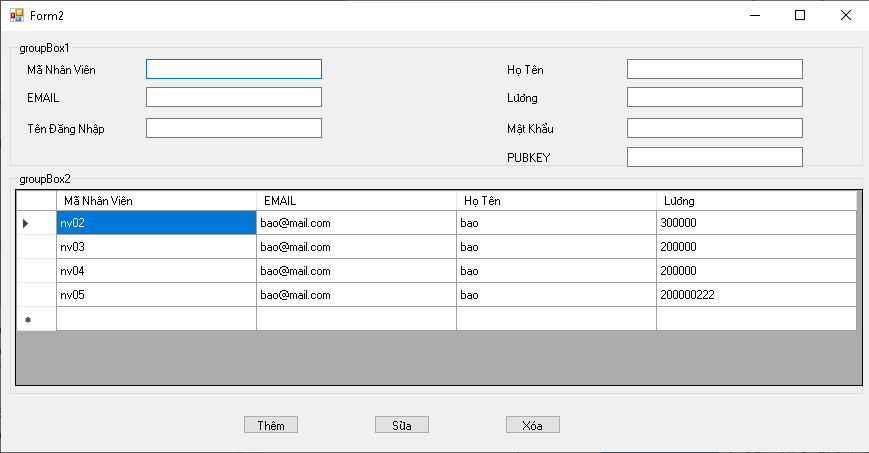
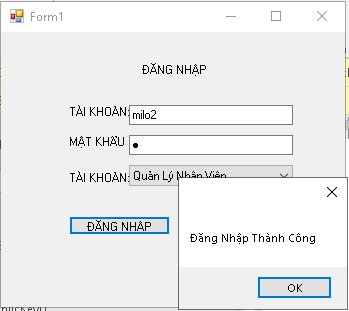


* Hàm Decrypt nhận một mảng byte dữ liệu đã được mã hóa, cùng với một mã để xác định khóa riêng tư tương ứng.
* Sau đó đọc khóa riêng tư và công khai từ file XML tương ứng. Sau đó, nó sử dụng khóa riêng tư để giải mã dữ liệu đã được mã hóa trước đó bằng phương thức Decrypt của đối tượng RSACryptoServiceProvider.
* Kết quả được chuyển đổi từ mảng byte thành chuỗi văn bản bằng mã hóa UTF-8 và trả

Giao diện phần mềm quản lí CSDL “sinh viên” (LAB 2)

Giao diện đăng nhập

TH1: Đăng nhập thành công chuyển đến các form khác



SQL Server Profiler

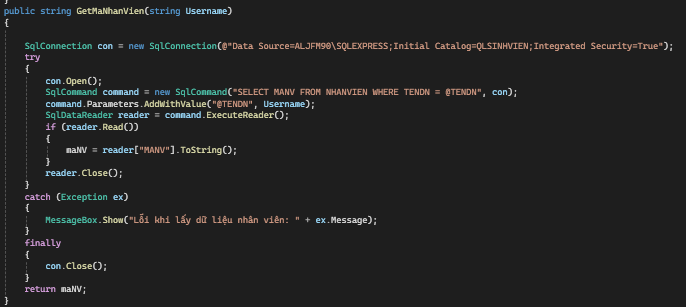




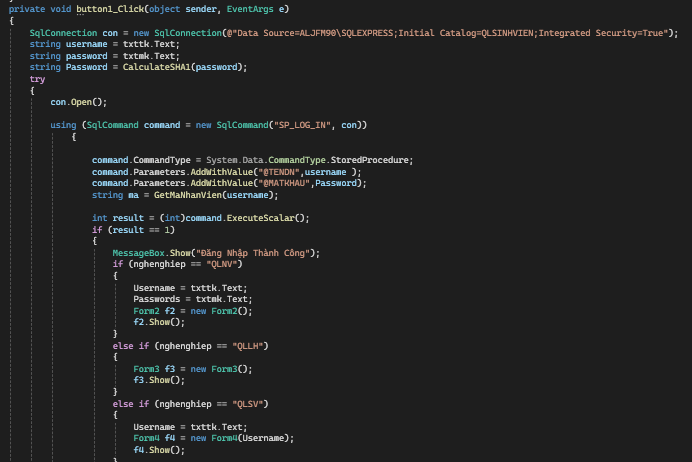
* Đăng nhập thành công sử dụng stored SP\_LOG\_IN để kiểm tra đăng nhập, nếu có dữ liệu đúng thì đăng nhập thành công và mật khẩu được mã hóa bằng SHA

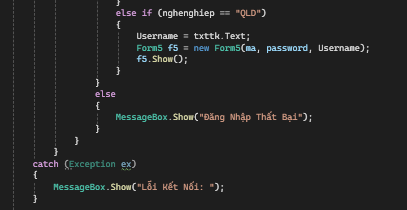
Code

Lấy mã nhân viên



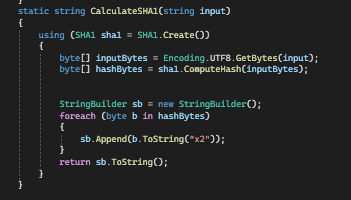
Hàm xử lý đăng nhập



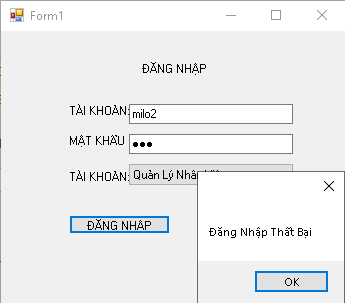


* Xử lý đăng nhập và chuyển các form khác nhau

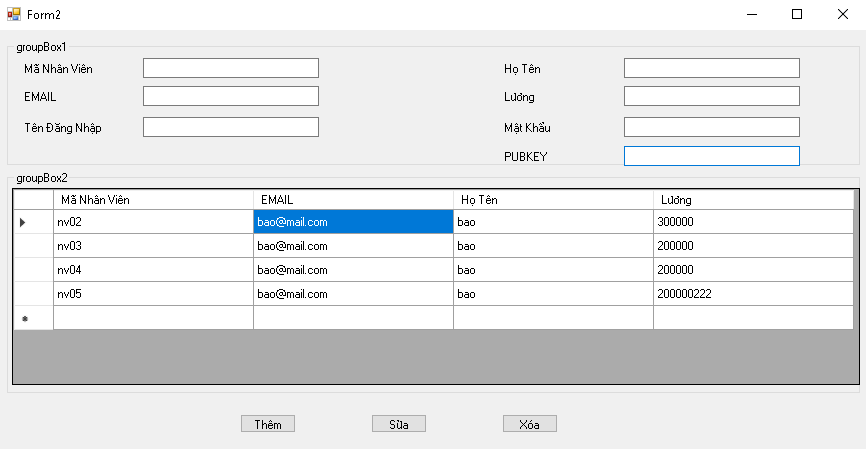
Hàm mã hóa SHA cho mật khẩu

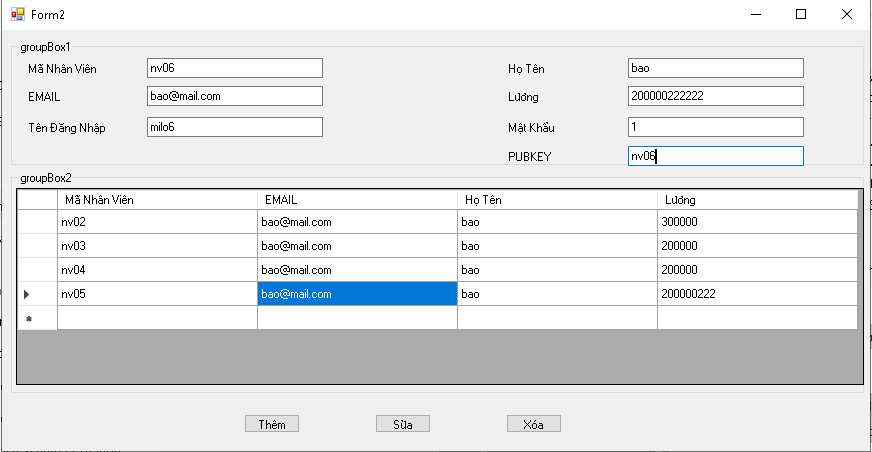


TH2: Đăng nhập thất bại thông báo cho người dùng

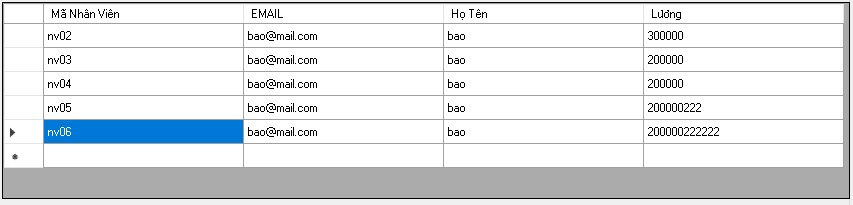


Giao diện quản lý nhân viên



Thêm nhân viên

Ấn nút 



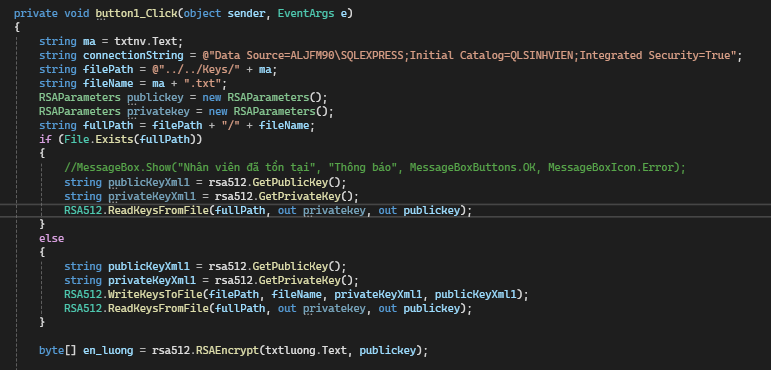
* Thêm thành công

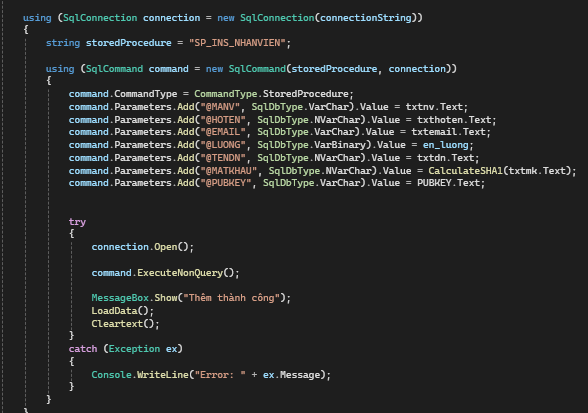
SQL Server Profiler



* exec SP\_INS\_NHANVIEN @MANV='nv06',@HOTEN=N'bao',@EMAIL='bao@mail.com',@LUONG=0x0B539A157A782CEADD50888D8138284A5C9800C78C28F54CDFB78847F23409B9FD0DC38BE3E37DF331E0EA4B9EC0071F91C89681B8C256E6A0C2E1BD1A10D5F1,@TENDN=N'milo6',@MATKHAU=N'356a192b7913b04c54574d18c28d46e6395428ab',@PUBKEY='nv06' (Đã mã hóa cột LUONG và MATKHAU bằng RSA)

Code



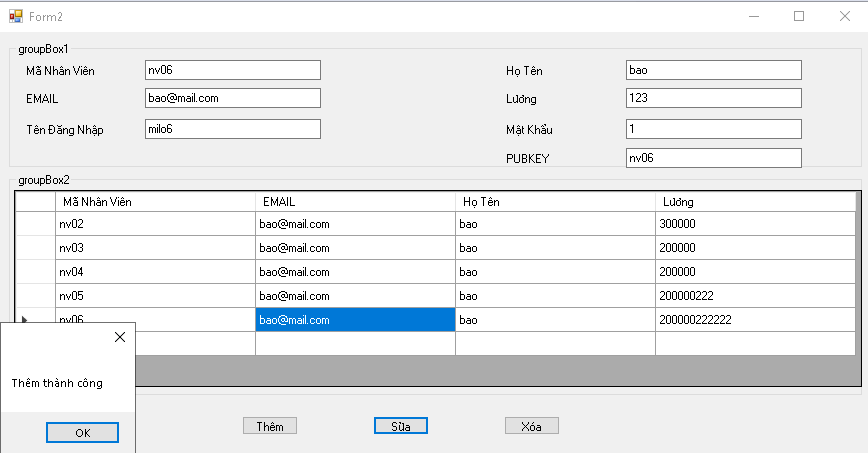


* Lấy mã nhân viên từ ô văn bản txtnv.
* Xây dựng chuỗi kết nối đến cơ sở dữ liệu SQL Server.
* Xây dựng đường dẫn đến tệp chứa khóa RSA dựa trên mã nhân viên.
* Kiểm tra xem tệp chứa khóa đã tồn tại hay chưa.
* Nếu đã tồn tại, đọc khóa từ tệp tin.
* Nếu chưa tồn tại, tạo mới khóa, ghi vào tệp tin và sau đó đọc khóa từ tệp tin.
* Mã hóa dữ liệu nhập từ ô văn bản txtluong sử dụng khóa công khai RSA.
* Mở kết nối đến cơ sở dữ liệu và thực thi một thủ tục lưu trữ để chèn thông tin của nhân viên vào cơ sở dữ liệu.
* Hiển thị thông báo thành công sau khi thêm dữ liệu vào cơ sở dữ liệu.
* Tải lại dữ liệu trong giao diện người dùng và xóa nội dung đã nhập trong các ô văn bản.

Sửa thông tin nhân viên

* Sửa nhân viên 06 ( Lương thành 123)

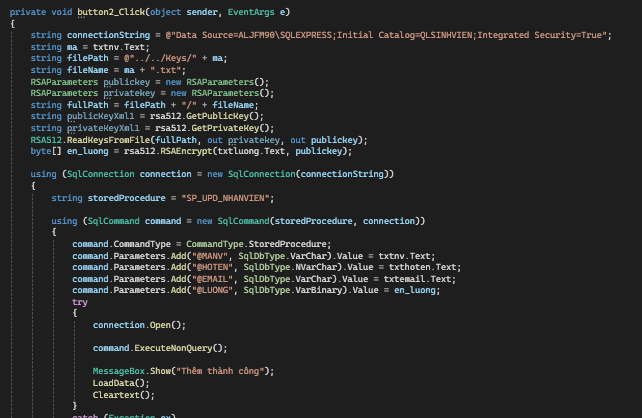




SQL Sever Profiler

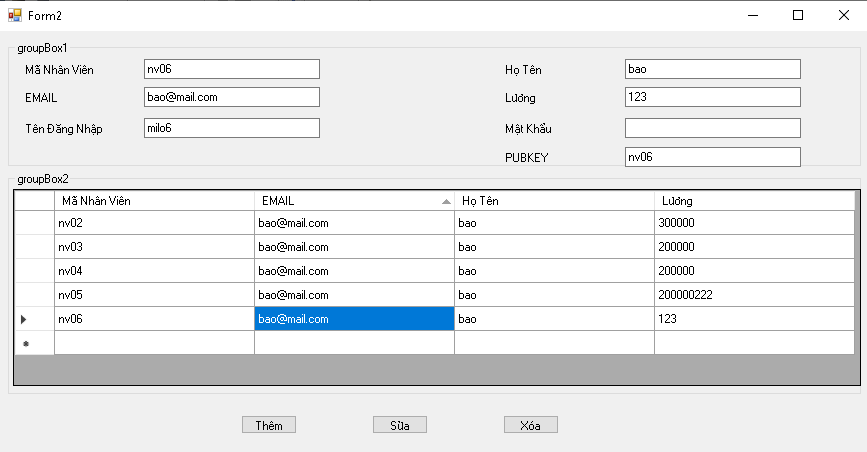


* SP\_UPD\_NHANVIEN @MANV='nv06',@HOTEN=N'bao',@EMAIL='bao@mail.com',@LUONG=0x775B84A547D6CB642528ED32054373CD91112900C33165A93CACF4DEE12AB3251E85366A4CCBAA14180F0898143C2E78443B84595F5F0301A81C6BC7CB4F246E ( Dữ liệu lương đã được mã hóa bằng RSA)

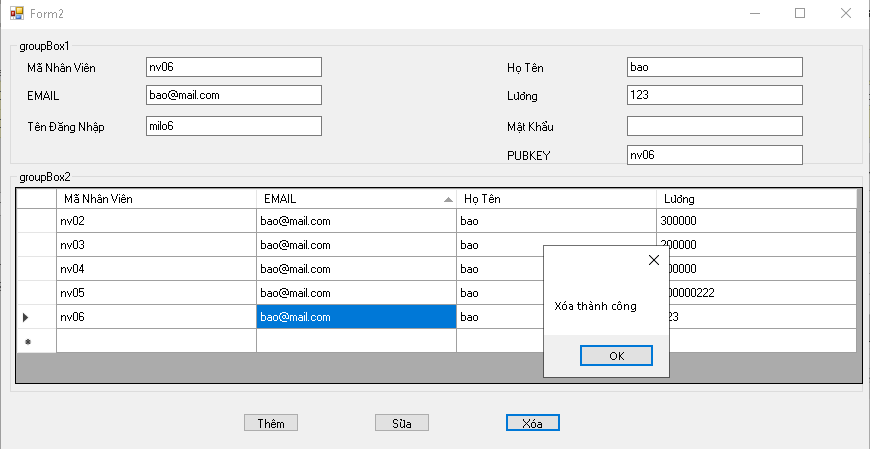
Code

* Xây dựng chuỗi kết nối đến cơ sở dữ liệu SQL Server.
* Lấy mã nhân viên từ ô văn bản txtnv.
* Xây dựng đường dẫn đến tệp chứa khóa RSA dựa trên mã nhân viên.
* Đọc khóa công khai và riêng tư từ tệp tin.
* Mã hóa dữ liệu nhập từ ô văn bản txtluong sử dụng khóa công khai RSA.
* Mở kết nối đến cơ sở dữ liệu và thực thi một thủ tục lưu trữ để cập nhật thông tin của nhân viên trong cơ sở dữ liệu.
* Hiển thị thông báo thành công sau khi cập nhật dữ liệu.
* Tải lại dữ liệu trong giao diện người dùng và xóa nội dung đã nhập trong các ô văn bản

Xóa nhân viên



Chọn nhân viên muốn xóa, nhấn nút 

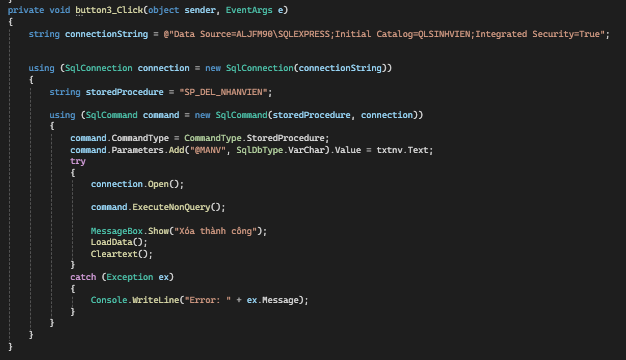


SQL Sever Profiler

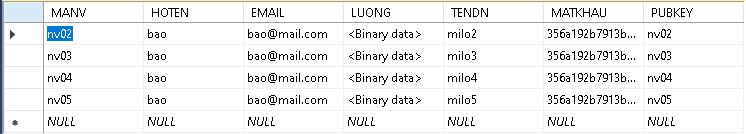




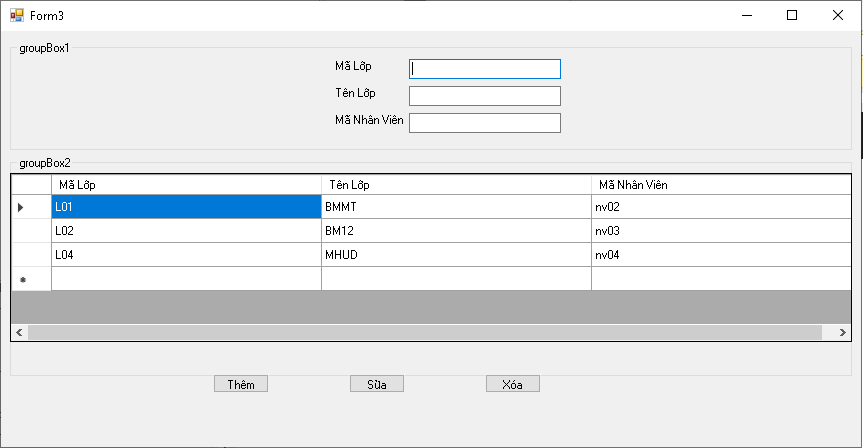
Code



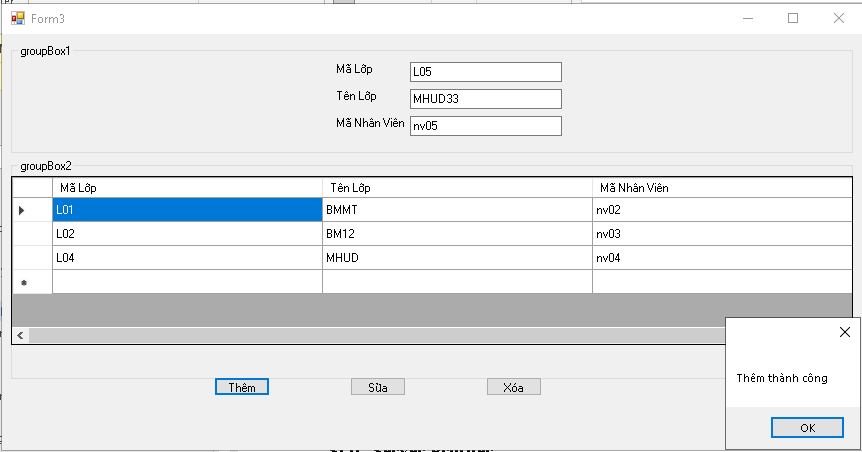
Dữ liệu hiển thị phía CSDL



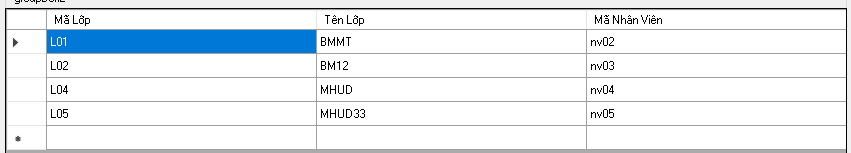
Giao diện quản lý lớp học



Thêm lớp học



* Nhập thông tin lớp và ấn nút 



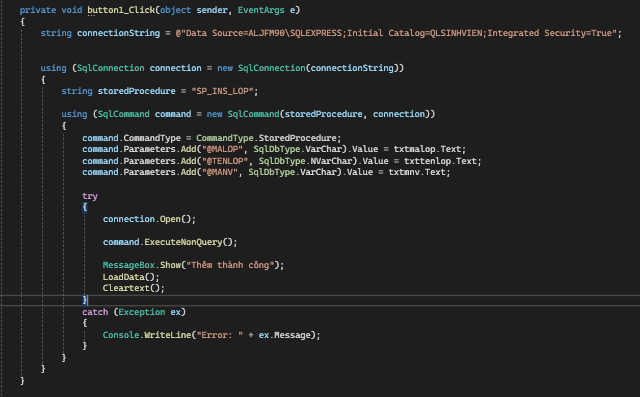
* Thêm thành công

SQL Server Profiler



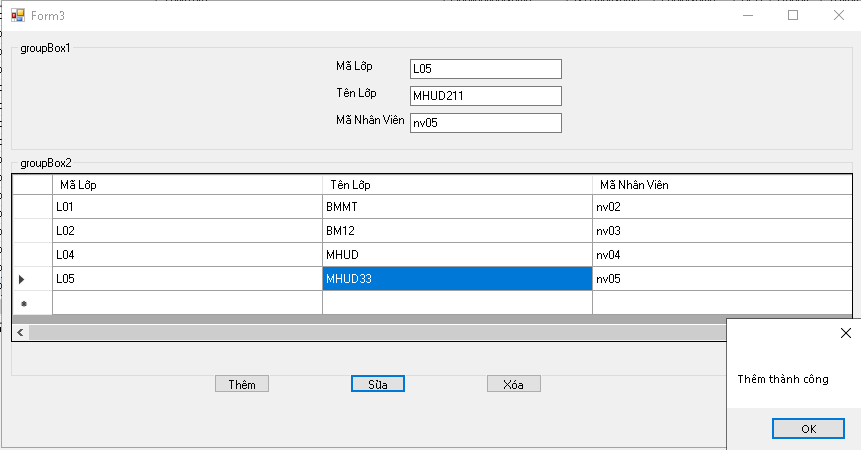


Code

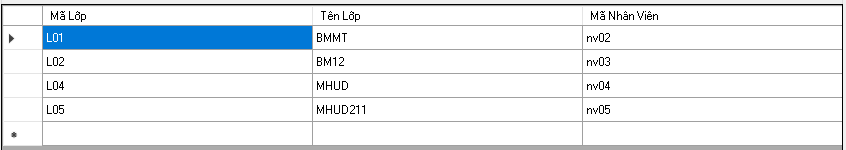


* Xây dựng chuỗi kết nối đến cơ sở dữ liệu SQL Server.
* Mở kết nối đến cơ sở dữ liệu và thực thi một thủ tục lưu trữ để chèn thông tin về một lớp vào cơ sở dữ liệu.
* Hiển thị thông báo thành công sau khi thêm dữ liệu vào cơ sở dữ liệu.
* Tải lại dữ liệu trong giao diện người dùng và xóa nội dung đã nhập trong các ô văn bản.

Sửa thông tin lớp học



* Nhập thông tin lớp học cần cập nhật, nhấn nút 



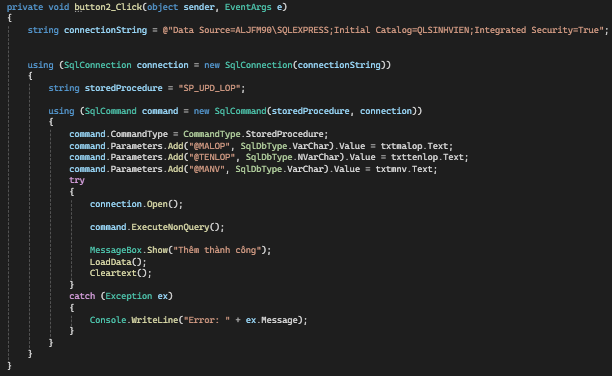
* Sửa thành công.

SQL Server Profiler



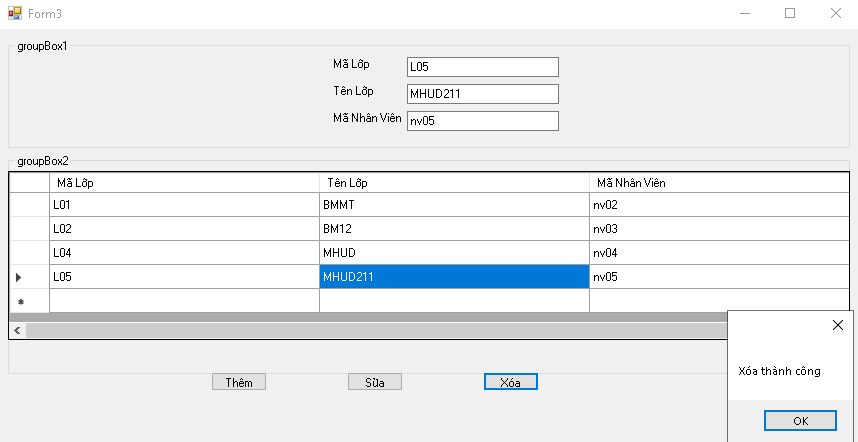


Code

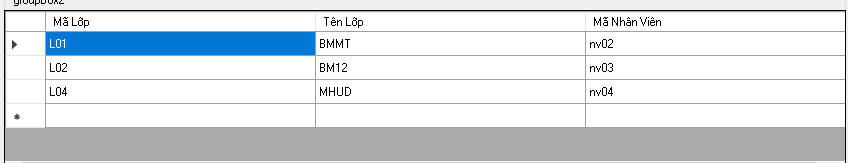


* Xây dựng chuỗi kết nối đến cơ sở dữ liệu SQL Server.
* Mở kết nối đến cơ sở dữ liệu và thực thi một thủ tục lưu trữ để cập nhật thông tin của một lớp trong cơ sở dữ liệu.
* Hiển thị thông báo thành công sau khi cập nhật dữ liệu.
* Tải lại dữ liệu trong giao diện người dùng và xóa nội dung đã nhập trong các ô văn bản.

Xóa lớp học



* Chọn lớp cần xóa, nhấn nút 



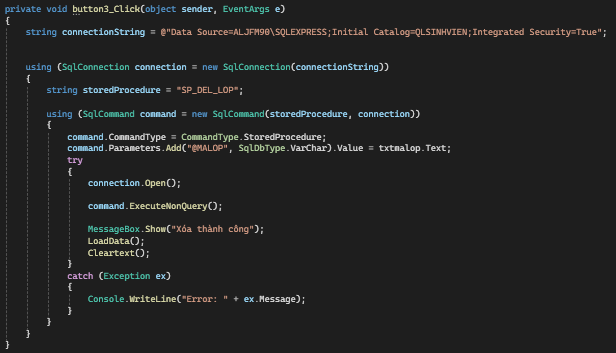
* Xóa thành công

SQL Server Profiler



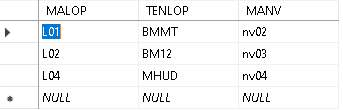


Code

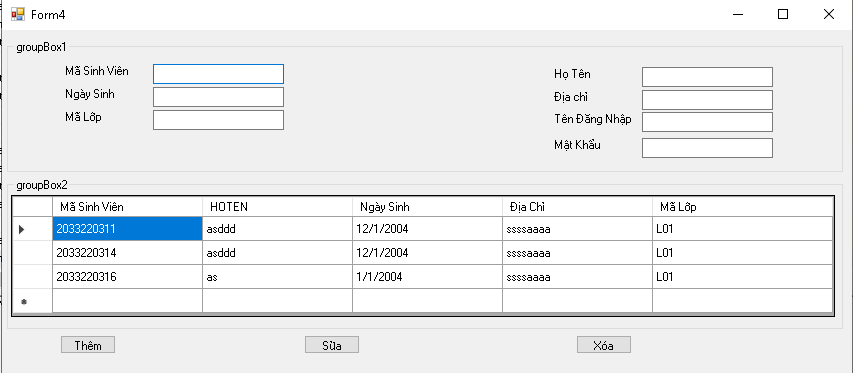


* Xây dựng chuỗi kết nối đến cơ sở dữ liệu SQL Server.
* Mở kết nối đến cơ sở dữ liệu và thực thi một thủ tục lưu trữ để xóa thông tin của một lớp trong cơ sở dữ liệu, dựa trên mã lớp được nhập vào từ ô văn bản `txtmalop`.
* Hiển thị thông báo thành công sau khi xóa dữ liệu.
* Tải lại dữ liệu trong giao diện người dùng và xóa nội dung đã nhập trong các ô văn bản.

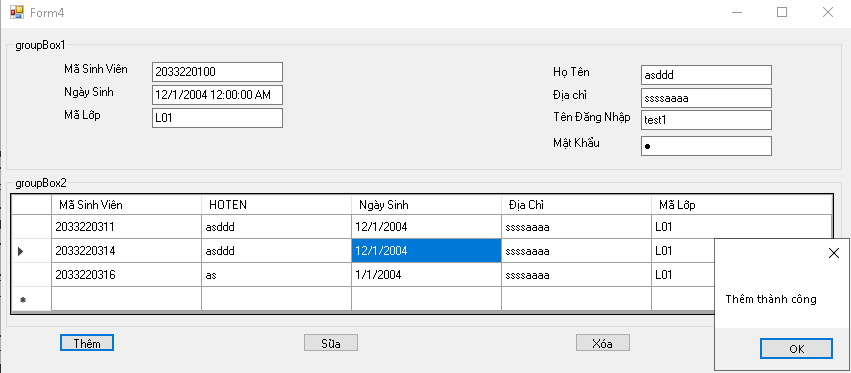
Dữ liệu hiển thị phía CSDL



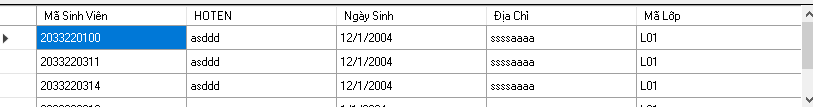
Giao diện quản lý sinh viên



Thêm sinh viên



* Nhập thông tin sinh viên, nhấn nút  ( không được trùng MSSV)



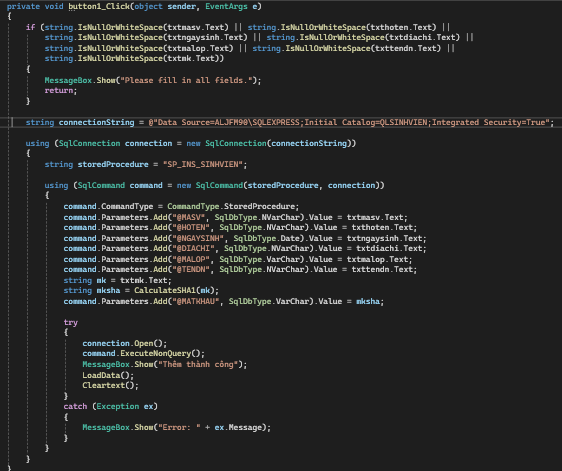
* Thêm thành công

SQL Server Profiler



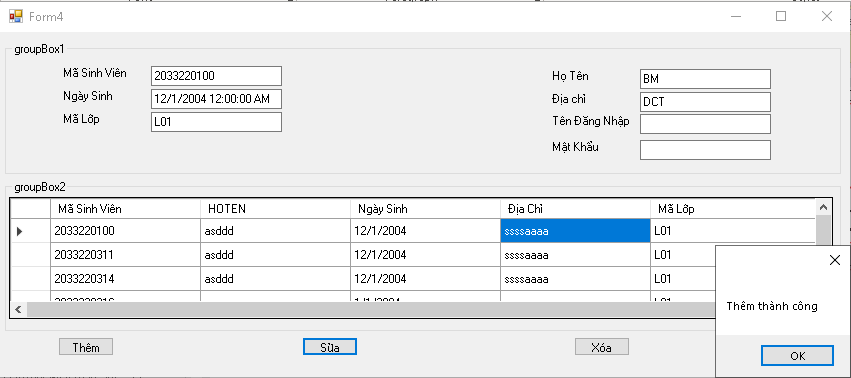
@MASV=N'2033220100',@HOTEN=N'asddd',@NGAYSINH='2004-12-01',@DIACHI=N'ssssaaaa',@MALOP='L01',@TENDN=N'test1',@MATKHAU='356a192b7913b04c54574d18c28d46e6395428ab' ( MATKHAU đã được mã hóa bằng RSA)

Code

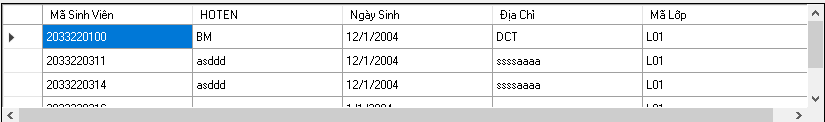


* Kiểm tra xem các ô văn bản nhập liệu có trống không. Nếu có bất kỳ ô nào trống, hiển thị thông báo lỗi và dừng lại.
* Xây dựng chuỗi kết nối đến cơ sở dữ liệu SQL Server.
* Mở kết nối đến cơ sở dữ liệu và thực thi một thủ tục lưu trữ để chèn thông tin về một sinh viên vào cơ sở dữ liệu. Thông tin được nhập từ các ô văn bản.
* Hiển thị thông báo thành công sau khi thêm dữ liệu vào cơ sở dữ liệu.
* Tải lại dữ liệu trong giao diện người dùng và xóa nội dung đã nhập trong các ô văn bản.

Sửa thông tin Sinh Viên



* Chọn sinh viên muốn sửa, nhấn 

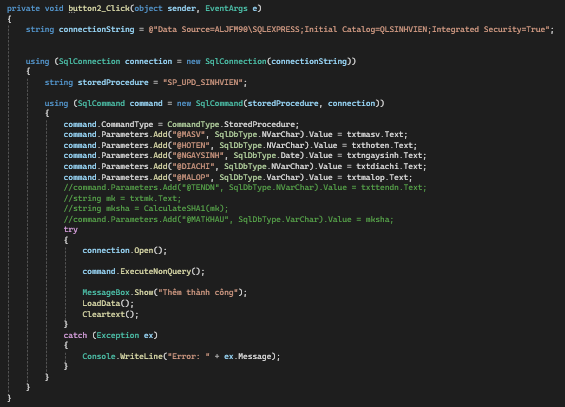


* Sửa thành công

SQL Server Profiler

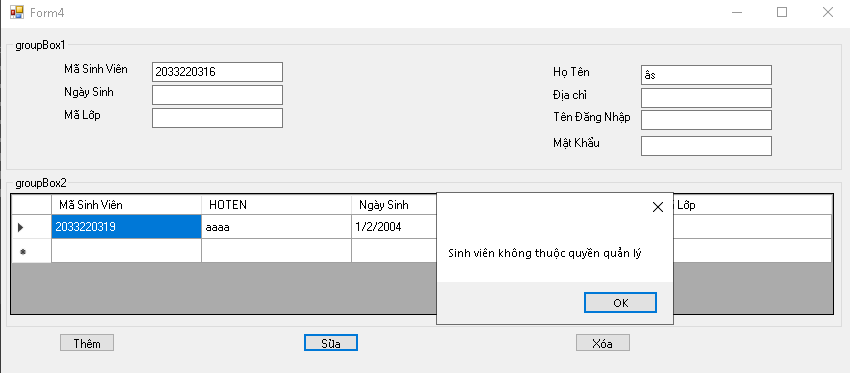


Code

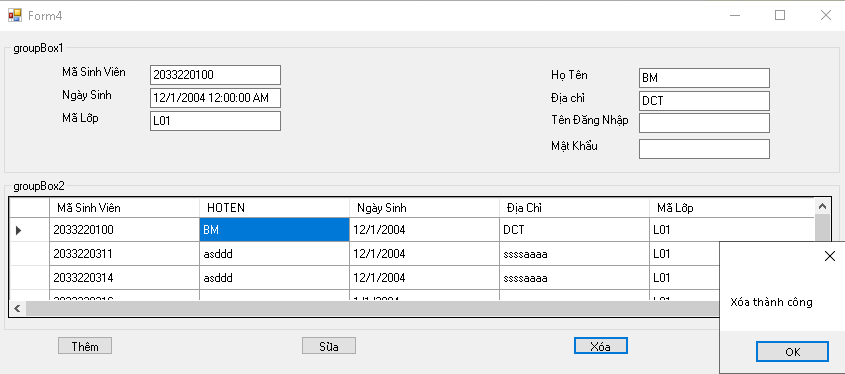


* Xây dựng chuỗi kết nối đến cơ sở dữ liệu SQL Server.
* Mở kết nối đến cơ sở dữ liệu và thực thi một thủ tục lưu trữ để cập nhật thông tin của một sinh viên trong cơ sở dữ liệu. Thông tin được nhập từ các ô văn bản.
* Hiển thị thông báo thành công sau khi cập nhật dữ liệu.
* Tải lại dữ liệu trong giao diện người dùng và xóa nội dung đã nhập trong các ô văn bản.

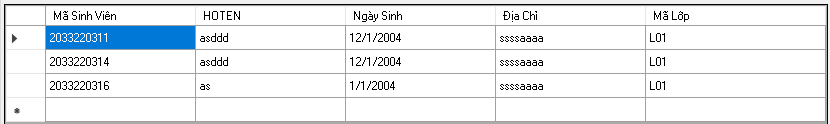
TH: Nếu sinh viên không thuộc nhân viên quản lí thông báo cho người dùng không sửa được



Xóa Sinh Viên



* Chọn sinh viên cần xóa, nhấn nút 



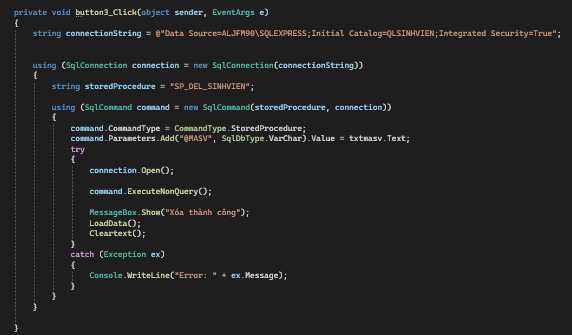
* Xóa thành công

SQL Server Profiler



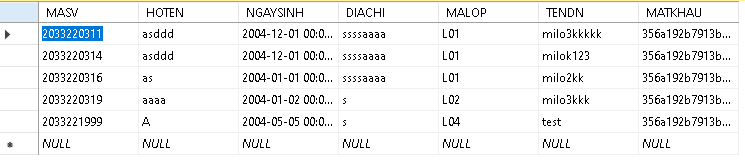


Code

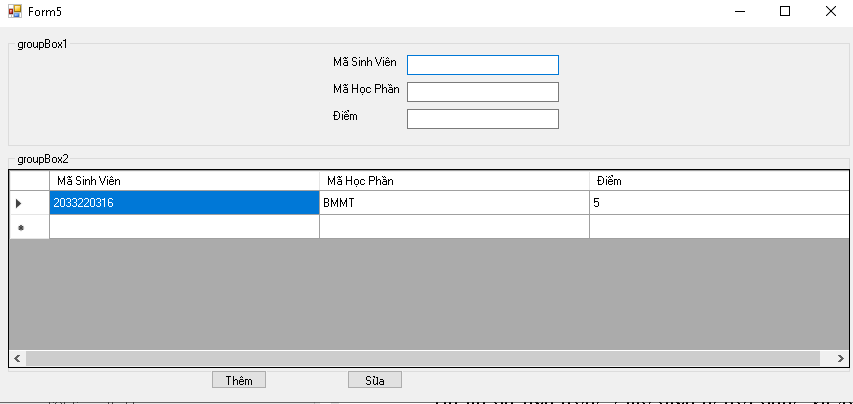


* Xây dựng chuỗi kết nối đến cơ sở dữ liệu SQL Server.
* Mở kết nối đến cơ sở dữ liệu và thực thi một thủ tục lưu trữ để xóa thông tin của một sinh viên từ cơ sở dữ liệu, dựa trên mã sinh viên được nhập vào từ ô văn bản `txtmasv`.
* Hiển thị thông báo thành công sau khi xóa dữ liệu.
* Tải lại dữ liệu trong giao diện người dùng và xóa nội dung đã nhập trong các ô văn bản.

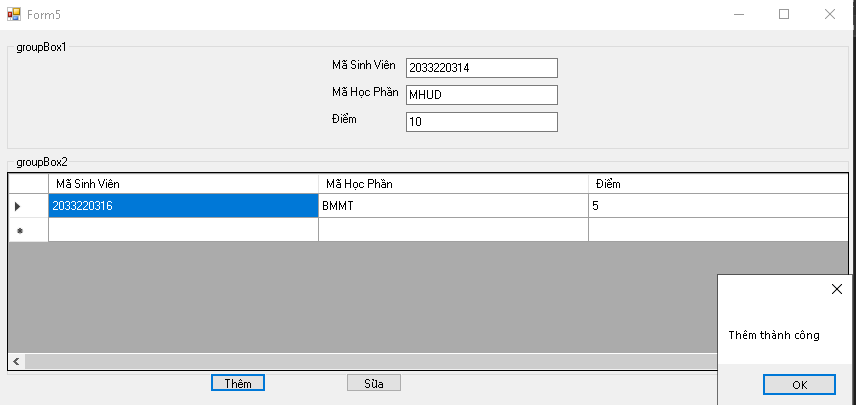
Dữ liệu hiển thị phía CSDL

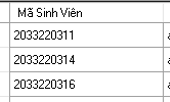


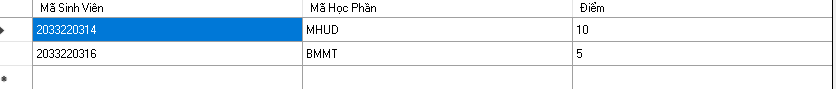
Giao diện quản lý điểm sinh viên



Thêm điểm dựa trên MSSV



MSSV dựa trên sinh viên ở phần quản lý sinh viên , nhấn nút 



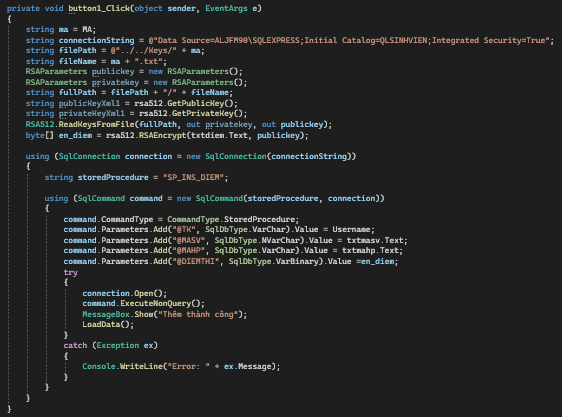
* Thêm thành công

SQL Server Profiler



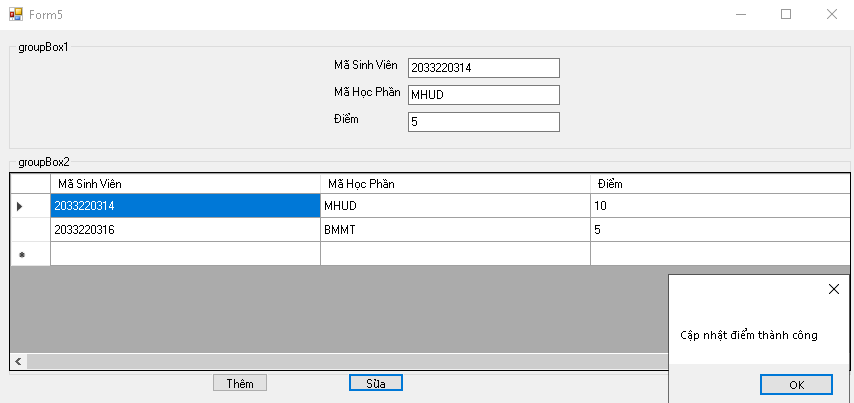
exec SP\_INS\_DIEM @TK='milo2',@MASV=N'2033220314',@MAHP='MHUD',@DIEMTHI=0x5F720CBA75856F12A9DAE41448D48110165EFB5743C9BCBC4B80EE6CC1A9BD6E58812FC5D408362D306CD601C554D29B5CD46661A535F15B3426E16AFA06E4DF ( điểm thi đã được mã hóa bằng RSA ).

Code

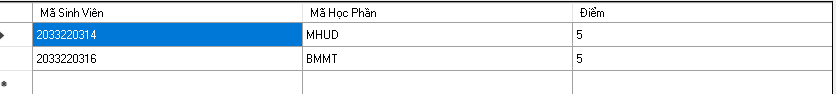


* + Xây dựng chuỗi kết nối đến cơ sở dữ liệu SQL Server.
  + Xác định đường dẫn đến thư mục chứa khóa RSA dựa trên một mã được chỉ định trước đó.
  + Tạo tên tệp tin từ mã và mở rộng .txt.
  + Đọc khóa công khai và khóa riêng tư từ tệp tin đã được chỉ định.
  + Mã hóa điểm thi nhập từ ô văn bản txtdiem bằng khóa công khai RSA.
  + Mở kết nối đến cơ sở dữ liệu và thực thi một thủ tục lưu trữ để chèn thông tin điểm thi của một sinh viên vào cơ sở dữ liệu. Thông tin được nhập từ các ô văn bản và mã hóa điểm thi.
  + Hiển thị thông báo thành công sau khi thêm dữ liệu vào cơ sở dữ liệu.
  + Tải lại dữ liệu trong giao diện người dùng.

Sửa điểm sinh viên

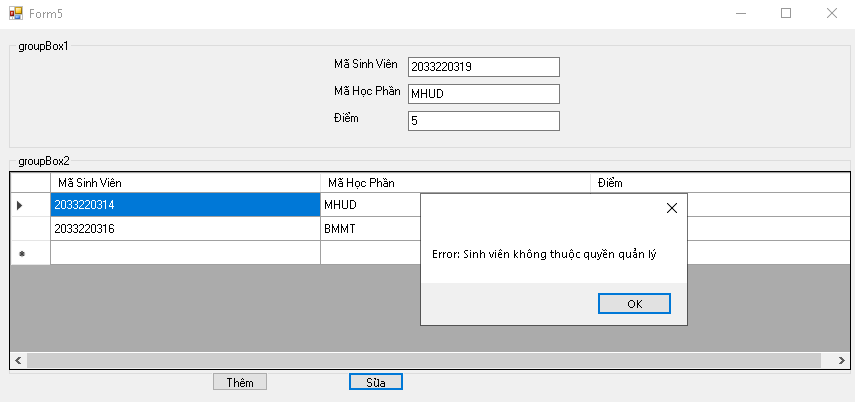


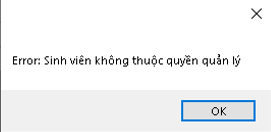
* Chọn sinh viên với mssv mà nhân viên dùng để đăng nhập quản lý, nhấn nút 



* Sửa thành công

TH: Nếu như sinh viên đó không do nhân viên đó quản lý, thông báo cho người dùng



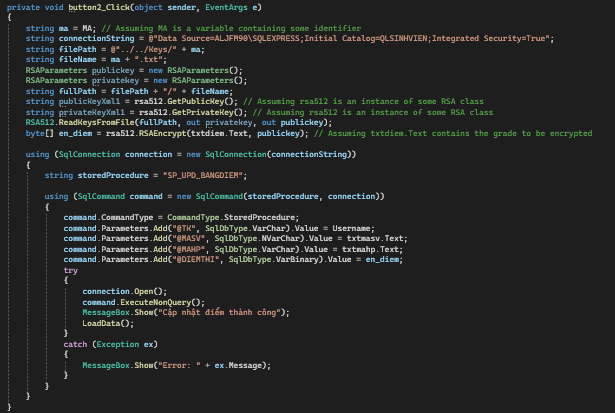
Thông báo 

SQL Server Profiler



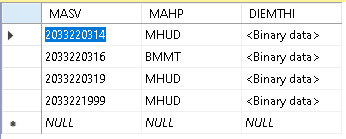
exec SP\_UPD\_BANGDIEM @TK='milo2',@MASV=N'2033220314',@MAHP='MHUD',@DIEMTHI=0x02D05C55569841E14B2FAAA9B857FA6D5C10F86A37F7548039CC4A929D76C4411F3B3957300D6C19639A0476F186A8563289FDDFD862161FCD225E42456358AA

Code

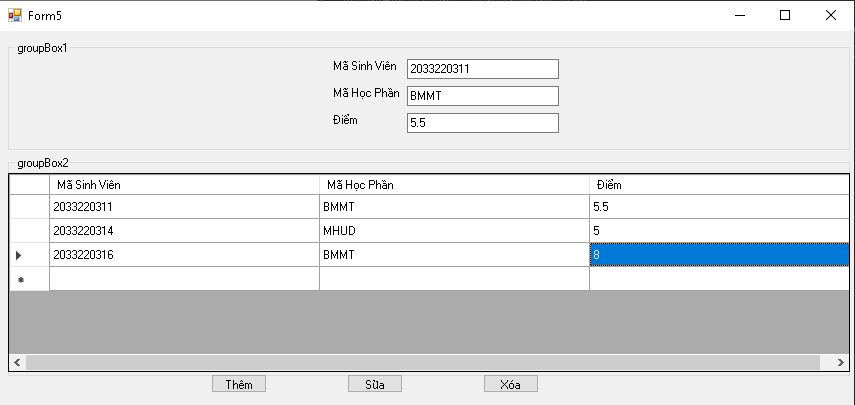


* Xây dựng chuỗi kết nối đến cơ sở dữ liệu SQL Server.
* Xác định đường dẫn đến thư mục chứa khóa RSA dựa trên một mã được xác định trước đó.
* Tạo tên tệp tin từ mã và mở rộng `.txt`.
* Đọc khóa công khai và khóa riêng tư từ tệp tin đã xác định.
* Mã hóa điểm thi nhập từ ô văn bản `txtdiem` bằng khóa công khai RSA.
* Mở kết nối đến cơ sở dữ liệu và thực thi một thủ tục lưu trữ để cập nhật thông tin điểm thi của một sinh viên trong cơ sở dữ liệu. Thông tin được nhập từ các ô văn bản và điểm thi được mã hóa.
* Hiển thị thông báo thành công sau khi cập nhật dữ liệu.
* Tải lại dữ liệu trong giao diện người dùng.

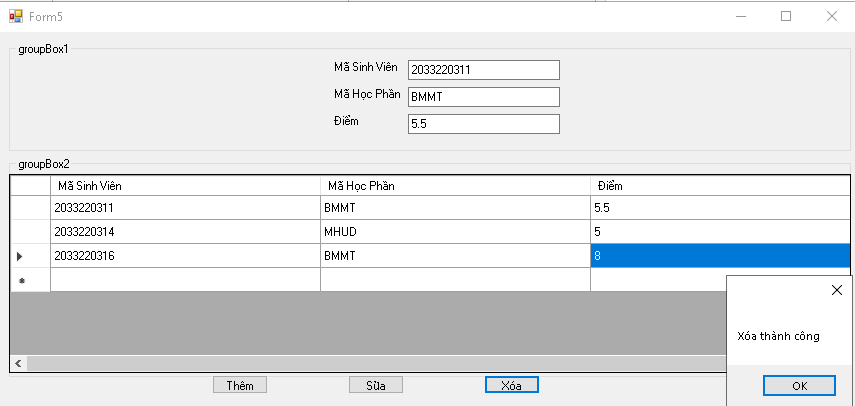
Dữ liệu hiển thị phía CSDL



Xóa toàn bộ thông tin điểm dựa trên MSSV



Chọn sinh viên cần xóa -> Nhấn nút 



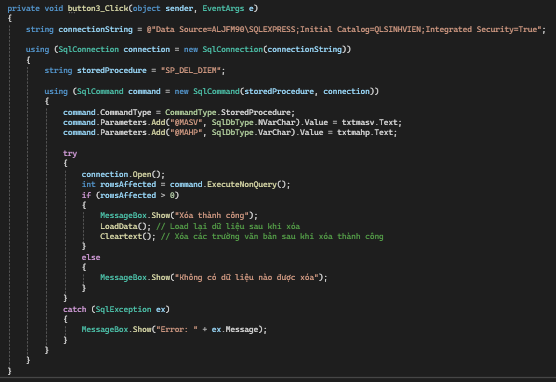
Xóa thành công

SQL Sever Profiler



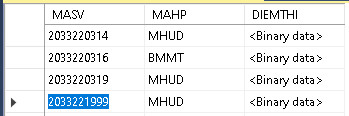


Code



* Xây dựng chuỗi kết nối đến cơ sở dữ liệu SQL Server.
* Gọi đến stored DEL\_DIEM.
* Code hiển thị và tải lại thông tin cho người dùng.

Dữ liệu hiển thị phía CSDL



PHẦN KẾT

Kết Luận

Qua 2 bài Lab cuối kì đã cho nhóm tụi em tìm hiểu được cách Mã Hóa và Bảo Mật cơ bản cơ sở dữ liệu ở cả 2 phía Cilent và người dùng. Cảm ơn thầy đã giải thích và hướng dẫn cho chúng em trong quá trình học tập.

Tài liệu tham khảo

[[DES] Thuật toán mã hóa và giải mã DES ~ VLSI TECHNOLOGY (nguyenquanicd.blogspot.com)](https://nguyenquanicd.blogspot.com/2017/08/background-thuat-toan-ma-hoa-va-giai-ma.html)

[[C#] Mã hóa RSA trong lập trình Csharp (laptrinhvb.net)](https://laptrinhvb.net/bai-viet/devexpress/---Csharp----Ma-hoa-RSA-trong-lap-trinh-Csharp/73fed50cf0bc0881.html)

[Mã hóa chuỗi bằng thuật toán MD5 Hash với C# (nonnguyen.com)](https://blog.nonnguyen.com/2019/10/ma-hoa-chuoi-bang-thuat-toan-md5-hash-voi-csharp.html)

https://youtu.be/opIt2CRkcic?si=-IKWjyLUaqehWjWp