Mã sinh viên: 21010471

Họ và tên: Nguyễn Thành Phát

Lớp: DHHTTT17A

Hạn nộp: 23h59, ngày 23/03/2024.

**PHẦN: THỰC HÀNH TUẦN 5**

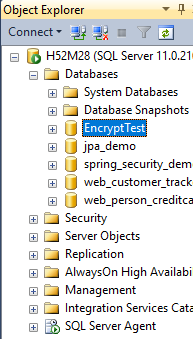
**Bài 1: Thực hiện mã hóa và giải mã theo các lệnh sau, bạn hãy giải thích ý nghĩa, chức năng của từng lệnh đã thực hiện Encryption**

**1.1. Tạo database**

**Mã SQL:**

CREATE DATABASE EncryptTest

**Kết quả:**

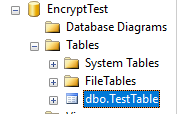
****

**1.2. Tạo table**

**Mã SQL:**

CREATE TABLE TestTable (FirstCol INT, SecondCol VARBINARY(256))

**Kết quả:**

****

**1.3. Tạo database Master Key**

**Mã SQL:**

CREATE MASTER KEY ENCRYPTION

BY PASSWORD = 'SQLAuthority'

GO

**Kết quả:**

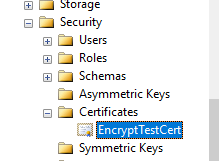
**1.4 Tạo Encryption Certificate**

**Mã SQL:**

CREATE CERTIFICATE EncryptTestCert

WITH SUBJECT = 'SQLAuthority'

**Kết quả:**

****

**1.5 Tạo Symmetric Key**

**Mã SQL:**

CREATE SYMMETRIC KEY TestTableKey

WITH ALGORITHM = TRIPLE\_DES ENCRYPTION

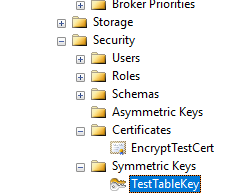
BY CERTIFICATE EncryptTestCert

**Kích hoạt**

OPEN SYMMETRIC KEY TestTableKey DECRYPTION BY CERTIFICATE

EncryptTestCert

**Kết quả:**

****

**1.6 Insert dữ liệu**

**Mã SQL:**

INSERT INTO TestTable

values(1,ENCRYPTBYKEY(KEY\_GUID('TestTableKey'),'Hello'))

INSERT INTO TestTable

values(2,ENCRYPTBYKEY(KEY\_GUID('TestTableKey'),'123456'))

INSERT INTO TestTable

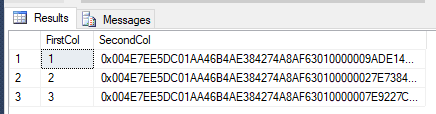
values(3,ENCRYPTBYKEY(KEY\_GUID('TestTableKey'),'gogogo'))

go

-- check

SELECT \* FROM TestTable

**Kết quả:**

****

**1.7 Decrypt the data of the SecondCol**

**Mã SQL:**

OPEN SYMMETRIC KEY TestTableKey DECRYPTION BY CERTIFICATE EncryptTestCert

**Kết quả:**

**1.8 Select data**

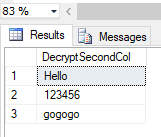
**Mã SQL:**

SELECT CONVERT(VARCHAR(50),DECRYPTBYKEY(SecondCol)) AS

DecryptSecondCol

FROM TestTable

**Kết quả:**

****

**1.9 Đóng Symmetric**

**Mã SQL:**

CLOSE SYMMETRIC KEY TestTableKey

**Kết quả:**

**BÀI 2 Thực hiện mã hóa và giải mã theo các lệnh sau, bạn hãy giải thích ý nghĩa, chức năng của từng lệnh đã thực hiện**

**2.1 Tạo database**

**Mã SQL:**

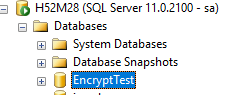
CREATE DATABASE EncryptTest

ON PRIMARY ( NAME = N'EncryptTest', FILENAME = N'C:\EncryptTest.mdf')

LOG ON ( NAME = N'EncryptTest\_log', FILENAME =N'C:\EncryptTest\_log.ldf')

GO

**Kết quả:**

****

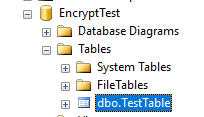
**2.2 Create table and insert data in the table**

**Mã SQL:**

CREATE TABLE TestTable (FirstCol INT, SecondCol VARCHAR(50))

GO

**Kết quả:**

****

**2.3 Insert dữ liệu**

**Mã SQL:**

INSERT INTO TestTable (FirstCol, SecondCol) SELECT 1,'First'

UNION ALL SELECT 2,'Second'

UNION ALL SELECT 3,'Third'

UNION ALL SELECT 4,'Fourth'

UNION ALL SELECT 5,'Fifth'

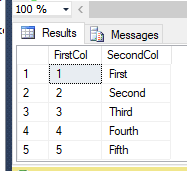
**Check**

SELECT \*

FROM TestTable

GO

**Kết quả:**

****

**2.4 Decrypt the data of the SecondCol**

**Mã SQL:**

CREATE MASTER KEY ENCRYPTION

BY PASSWORD = 'SQLAuthority'

GO

**Kết quả:**

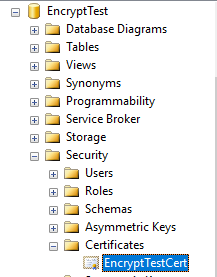
**2.5 Create Encryption Certificate**

**Mã SQL:**

CREATE CERTIFICATE EncryptTestCert

WITH SUBJECT = 'SQLAuthority'

**Kết quả:**

****

**2.6 Create Symmetric Key**

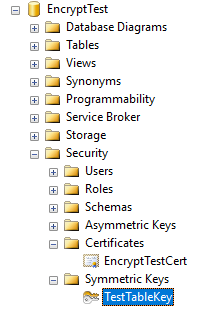
**Mã SQL:**

CREATE SYMMETRIC KEY TestTableKey

WITH ALGORITHM = TRIPLE\_DES ENCRYPTION

BY CERTIFICATE EncryptTestCert

**Kết quả:**

****

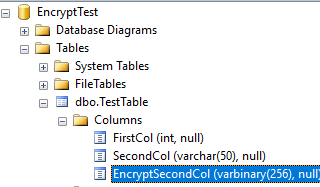
**2.7 Encrypt Data using Key and Certificate Add Columns which will hold the encrypted data in binary**

**Mã SQL:**

ALTER TABLE TestTable

ADD EncryptSecondCol VARBINARY(256)

**Kết quả:**

****

**2.8 Update binary column with encrypted data created by certificate and key**

**Mã SQL:**

GO

OPEN SYMMETRIC KEY TestTableKey DECRYPTION

BY CERTIFICATE EncryptTestCert

UPDATE TestTable

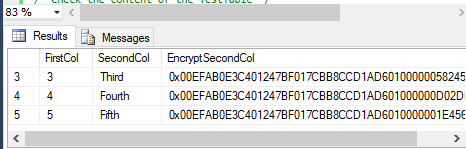
SET EncryptSecondCol =ENCRYPTBYKEY(KEY\_GUID('TestTableKey'),SecondCol)

GO

**Check**

select \* from TestTable

**Kết quả:**

****

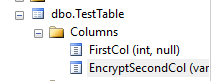
**2.9 DROP original column which was encrypted for protect the data**

**Mã SQL:**

ALTER TABLE TestTable

DROP COLUMN SecondCol

**Kết quả:**

****

**2.10 Check the content of the TestTable**

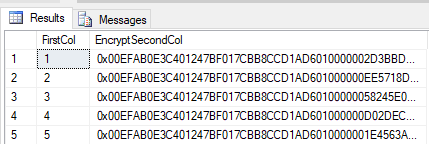
**Mã SQL:**

SELECT \*

FROM TestTable

GO

**Kết quả:**

****

**2.11 Decrypt the data of the SecondCol**

**Mã SQL:**

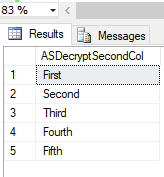
OPEN SYMMETRIC KEY TestTableKey DECRYPTION

BY CERTIFICATE EncryptTestCert

SELECT CONVERT(VARCHAR(50),DECRYPTBYKEY(EncryptSecondCol)) ASDecryptSecondCol

FROM TestTable

**Kết quả:**

****

**2.12 Clean up database**

**Mã SQL:**

CLOSE SYMMETRIC KEY TestTableKey

GO

DROP SYMMETRIC KEY TestTableKey

GO

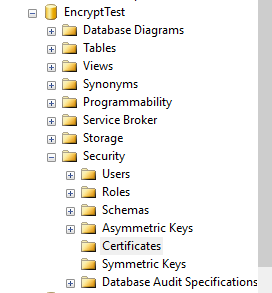
DROP CERTIFICATE EncryptTestCert

GO

DROP MASTER KEY

GO

**Kết quả:**

****

**2.13 Drop database**

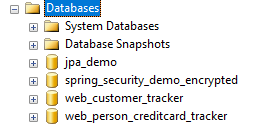
**Mã SQL:**

USE [master]

GO

DROP DATABASE [EncryptTest]

**Kết quả:**

****

**BÀI 3: Thực hiện mã hóa và giải mã theo các lệnh sau, bạn hãy giải thích ý nghĩa, chức năng của từng lệnh đã thực hiện.**

**3.1) Mã hóa mức cột:**

**Mã SQL:**

USE AdventureWorks2008R2;

**3.2) If there is no master key, create one now.**

**Mã SQL:**

IF NOT EXISTS (SELECT \* FROM sys.symmetric\_keys WHERE symmetric\_key\_id =

101) CREATE MASTER KEY ENCRYPTION BY PASSWORD =

'Th15i$aS7riN&ofR@nD0m!T3%t'

GO

**Kết quả:**

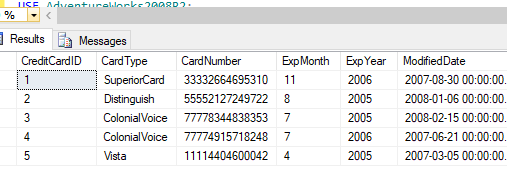
**3.3) Mã hóa cột sử dụng mật khẩu - Encrypting Columns Using a Passphrase**

**Mã SQL:**

select top 5 \* from Sales.CreditCard

go

**Kết quả:**

****

**3.4) Select:**

**Mã SQL:**

select CreditCardID, CardType, CardNumber\_encrypt = CONVERT(varbinary(256),

CardNumber), ExpMonth, ExpYear, ModifiedDate into Sales.CreditCard\_encrypt

from Sales.CreditCard where 1 = 2

**Kết quả:**

**3.5) Insert dữ liệu qua bảng Sales.CreditCard\_encrypt:**

**Mã SQL:**

declare @passphrase varchar(128)

set @passphrase = 'unencrypted credit card numbers are bad, um-kay'

insert Sales.CreditCard\_encrypt ( CardType, CardNumber\_encrypt, ExpMonth, ExpYear, ModifiedDate)

select top 5 CardType, CardNumber\_encrypt = EncryptByPassPhrase(@passphrase,

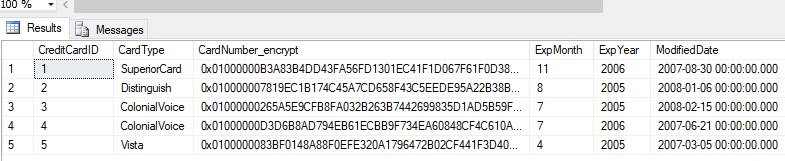
CardNumber), ExpMonth, ExpYear, ModifiedDate from Sales.CreditCard

**Check**

select \* from Sales.CreditCard\_encrypt

go

**Kết quả:**

****

**3.6) Select data:**

**Mã SQL:**

declare @passphrase varchar(128)

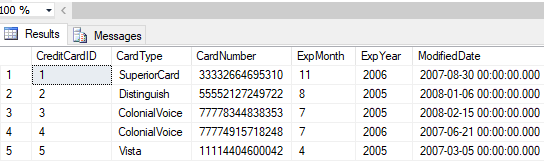
set @passphrase = 'unencrypted credit card numbers are bad, um-kay'

select CreditCardID, CardType, CardNumber = convert(nvarchar(25),

DecryptByPassPhrase(@passphrase, CardNumber\_encrypt)), ExpMonth, ExpYear,

ModifiedDate from Sales.CreditCard\_encrypt

**Kết quả:**

****

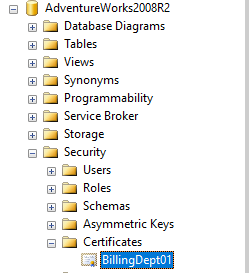
**3.7) The first step is to create the certificate with the CREATE CERTIFICATE command:**

**Mã SQL:**

CREATE CERTIFICATE BillingDept01 WITH SUBJECT = 'Credit Card Billing'

GO

**Kết quả:**

****

**3.8) Create Symmetric Key:**

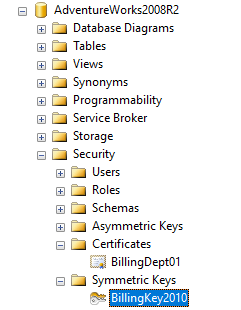
**Mã SQL:**

CREATE SYMMETRIC KEY BillingKey2010 WITH ALGORITHM = AES\_256

ENCRYPTION BY CERTIFICATE BillingDept01;

GO

**Kết quả:**

****

**3.9) Truncate table:**

**Mã SQL:**

Truncate table Sales.CreditCard\_encrypt

USE AdventureWorks2008R2;

**Kết quả:**

**3.10) First, decrypt the key using the BillingDept01 certificate**

**Mã SQL:**

OPEN SYMMETRIC KEY BillingKey2010 DECRYPTION BY CERTIFICATE BillingDept01

**Kết quả:**

**3.11) Now, insert the rows using the symmetric key encrypted by the certificate**

**Mã SQL:**

insert Sales.CreditCard\_encrypt ( CardType, CardNumber\_encrypt, ExpMonth, ExpYear,

ModifiedDate )

select top 5 CardType, CardNumber\_encrypt =

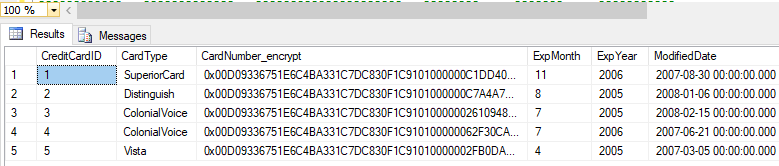
EncryptByKey(KEY\_GUID('BillingKey2010'), CardNumber), ExpMonth, ExpYear,

ModifiedDate from Sales.CreditCard

**check**

select \* from Sales.CreditCard\_encrypt

**Kết quả:**

****

**3.12) Giải mãt:**

**Mã SQL:**

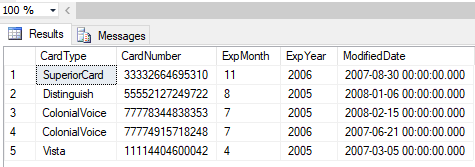
OPEN SYMMETRIC KEY BillingKey2010 DECRYPTION BY CERTIFICATE

BillingDept01 Select CardType, CardNumber = convert(nvarchar(25),

DecryptByKey(CardNumber\_encrypt)), ExpMonth, ExpYear, ModifiedDate from

Sales.CreditCard\_encrypt

**Kết quả:**

****

**3.13) Close:**

**Mã SQL:**

CLOSE SYMMETRIC KEY BillingKey2010

**Kết quả:**

**3.14) Select:**

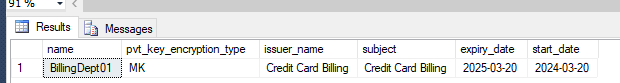
**Mã SQL:**

select name, pvt\_key\_encryption\_type, issuer\_name, subject, expiry\_date =

CAST(expiry\_date as DATE), start\_date = CAST(start\_date as DATE) from sys.certificates

go

**Kết quả:**

****

**3.15) Select:**

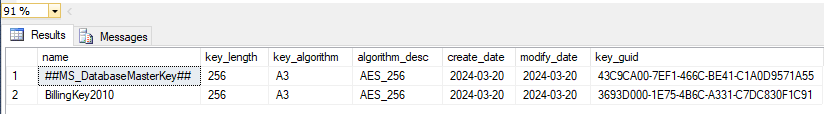
**Mã SQL:**

select name, key\_length, key\_algorithm, algorithm\_desc, create\_date = CAST(create\_date as

DATE), modify\_date = CAST(create\_date as DATE), key\_guid from sys.symmetric\_keys

go

**Kết quả:**

****

**3.16) If the usage of the key and certificate are no longer needed, they should be dropped from**

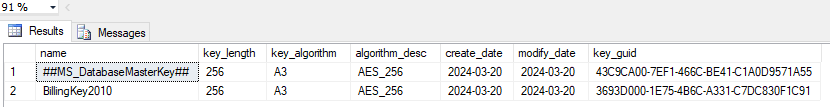
**Mã SQL:**

select name, key\_length, key\_algorithm, algorithm\_desc, create\_date = CAST(create\_date as

DATE), modify\_date = CAST(create\_date as DATE), key\_guid from sys.symmetric\_keys

go

**Kết quả:**

****

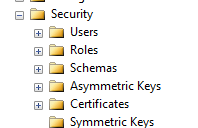
**3.16) If the usage of the key and certificate are no longer needed, they should be dropped from the database**

**Mã SQL:**

DROP SYMMETRIC KEY BillingKey2010

DROP CERTIFICATE BillingDept01

**Kết quả:**

****

**2) Mã hóa dữ liệu trong suốt**

**3.18) Create a certificate that is also stored in the master -- database. This certificate will be used to encrypt a user database**

**Mã SQL:**

CREATE CERTIFICATE MyCertificate with SUBJECT = 'Certificate stored in Master Db'

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

**3.19) Create a Database Encryption Key (DEK) that is based on the previously created certificate The DEK is stored in the user database**

**Mã SQL:**

USE AdventureWorks2008R2

GO

CREATE DATABASE ENCRYPTION KEY WITH ALGORITHM = AES\_256

ENCRYPTION BY SERVER CERTIFICATE MyCertificate

GO

**Kết quả:**

**Bài 4: Thực hiện mã hóa và giải mã theo các lệnh sau, bạn hãy giải thích ý nghĩa, chức năng của từng lệnh đã thực hiện**

**4.1. Backing up the certificate, private key, and master key for the server is relatively  
straightforward**

**Mã SQL:**

BACKUP MASTER KEY TO FILE =

'E:\HK6\BaoMat\_CSDL\TH\buoi7\test\_b4\backup\masterkey'

ENCRYPTION BY PASSWORD = 'somekeybackuppassword$$'

BACKUP CERTIFICATE MyCertificate TO FILE

= 'E:\HK6\BaoMat\_CSDL\TH\buoi7\test\_b4\backup\MyCertificate\TDE\_CERT.cer'

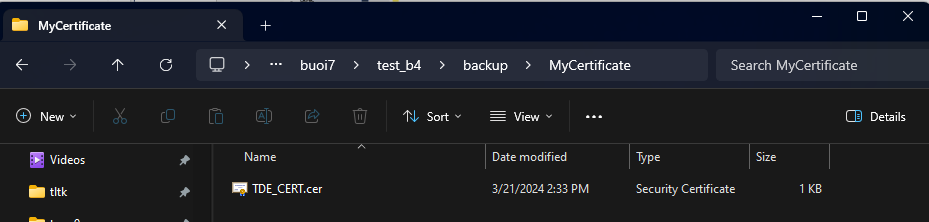
WITH PRIVATE KEY

( FILE

= 'E:\HK6\BaoMat\_CSDL\TH\buoi7\test\_b4\backup\MyCertificatePrivateKey\pkey.key' ,

ENCRYPTION BY PASSWORD = 'somecertbackuppassword$$' )

**Kết quả:**

****

**A screenshot of a computer

Description automatically generated**

**BÀI 5**: **Ứng dụng bài toán thực tế  
1) Tạo CSDL QLDA các thông số tùy ý**

**Mã SQL:**

use master

go

CREATE DATABASE QLDA

ON PRIMARY ( NAME = N'QLDA',

FILENAME = N'E:\HK6\BaoMat\_CSDL\TH\buoi7\test\_b5\EncryptTestt.mdf')

LOG ON ( NAME = N'QLDA\_log',

FILENAME =N'E:\HK6\BaoMat\_CSDL\TH\buoi7\test\_b5EncryptTest\_logg.ldf')

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

**2) Tạo các Table sau:  
Phongban(mapb, tenpb,mota)  
NhanVien(Manv, tennv, pass, phai, DienThoai, email, mapb)  
Duan(Mada, TenDa, NgayBD, KinhPhi, MaPB)  
Thamgia(Manv, Mada, NgayTG, MucLuong, CongViec)**  
**Kiểu dữ liệu sinh viên tự qui định. Yêu cầu tạo đầy đủ các ràng buộc khóa chính và  
khóa ngoại**

**Mã SQL:**

-- Tạo bảng PhongBan

CREATE TABLE PhongBan (

mapb INT identity(1,1) PRIMARY KEY,

tenpb NVARCHAR(100),

mota NVARCHAR(MAX)

);

-- Tạo bảng NhanVien

CREATE TABLE NhanVien (

Manv INT identity(1,1) PRIMARY KEY,

tennv NVARCHAR(100),

pass NVARCHAR(100),

phai bit,

DienThoai NVARCHAR(20),

email NVARCHAR(100),

mapb INT,

FOREIGN KEY (mapb) REFERENCES PhongBan(mapb)

);

-- Tạo bảng DuAn

CREATE TABLE DuAn (

Mada INT identity(1,1) PRIMARY KEY,

TenDa NVARCHAR(100),

NgayBD DATETIME,

KinhPhi int,

MaPB INT,

FOREIGN KEY (MaPB) REFERENCES PhongBan(mapb)

);

-- Tạo bảng ThamGia

CREATE TABLE ThamGia (

Manv INT,

Mada INT,

NgayTG DATETIME,

MucLuong int,

CongViec NVARCHAR(MAX),

PRIMARY KEY (Manv, Mada), -- Khóa chính là sự kết hợp của hai khóa ngoại

FOREIGN KEY (Manv) REFERENCES NhanVien(Manv),

FOREIGN KEY (Mada) REFERENCES DuAn(Mada)

);

**Kết quả:**

A screenshot of a computer

Description automatically generated

**3) Theo bạn, bạn sẽ mã hóa theo cột nào trong bảng NhanVien? Vì sao? Bạn hãy thực hiện mã hóa và giải mã cho cột mà bạn đã chọn theo hai cách dùng password và dùng chứng chỉ. Cho biết kết quả sau khi mã hóa và giải mã.**

- Theo em sẽ mã hóa cột (pass, DienThoai, email) vì đây điều là thông tin cá nhân quan trọng. Hacker nếu lấy được có thể truy cập các ứng dụng hoặc tạo email giả mạo.

**3.1 Mã hóa dữ liệu**

**Mã SQL:** Insert data

-- Thêm dữ PhongBan

insert into PhongBan (tenpb, mota)

values (N'Nhân sự', N'phòng nhân sự')

insert into PhongBan (tenpb, mota)

values (N'Kế toán', N'phòng Kế toán')

insert into PhongBan (tenpb, mota)

values (N'Marketing', N'phòng Marketing')

insert into PhongBan (tenpb, mota)

values (N'Quản lý', N'phòng Quản lý')

insert into PhongBan (tenpb, mota)

values (N'Tài chính', N'phòng Tài chính')

--check

select \* from PhongBan

-- Thêm dữ NhanVien

insert into NhanVien (tennv, pass, phai, DienThoai, email, mapb)

values(N'Thành Phát 1', N'thanhphat1', '1', '123', 'abc', 11)

insert into NhanVien (tennv, pass, phai, DienThoai, email, mapb)

values(N'Thành Phát 2', N'thanhphat2', '0', '456', 'xyz', 12)

insert into NhanVien (tennv, pass, phai, DienThoai, email, mapb)

values(N'Thành Phát 3', N'thanhphat3', '1', '876', 'fgs', 13)

insert into NhanVien (tennv, pass, phai, DienThoai, email, mapb)

values(N'Thành Phát 4', N'thanhphat4', '0', '678', 'dfg', 14)

insert into NhanVien (tennv, pass, phai, DienThoai, email, mapb)

values(N'Thành Phát 5', N'thanhphat5', '1', '789', 'rty', 15)

--check

select \* from NhanVien

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**3.2 Mã hóa bằng mật khẩu**

**3.2.1 Thêm column các cột cần mã hóa**

**Mã SQL:**

alter table NhanVien

add pass\_mahoa varbinary(max)

alter table NhanVien

add DienThoai\_mahoa varbinary(max)

alter table NhanVien

add email\_mahoa varbinary(max)

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

**3.2.2 Update cột mã hóa bằng ENCRYPTBYPASSPHRASE**

**Mã SQL:**

update NhanVien

set pass\_mahoa = ENCRYPTBYPASSPHRASE('123', pass)

update NhanVien

set DienThoai\_mahoa = ENCRYPTBYPASSPHRASE('123', DienThoai)

update NhanVien

set email\_mahoa = ENCRYPTBYPASSPHRASE('123', email)

**- Check**

select CONVERT(nvarchar(max), DECRYPTBYPASSPHRASE('123', pass\_mahoa)) as pass,

CONVERT(nvarchar(max), DECRYPTBYPASSPHRASE('123', DienThoai\_mahoa)) as DienThoai,

CONVERT(nvarchar(max), DECRYPTBYPASSPHRASE('123', email\_mahoa)) as email

from NhanVien

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

**3.3 Mã hóa bằng chứng chỉ**

**3.3.1 Tạo master key**

**Mã SQL:**

USE QLDA;

IF NOT EXISTS (SELECT \* FROM sys.symmetric\_keys WHERE symmetric\_key\_id =

101) CREATE MASTER KEY ENCRYPTION BY PASSWORD =

'Th15i$aS7riN&ofR@nD0m!T3%t'

GO

-- check

SELECT \* FROM sys.symmetric\_keys

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

**3.3.2 Tạo certificate**

**Mã SQL:**

CREATE CERTIFICATE NhanVien01 WITH SUBJECT = 'NhanVienCer'

GO

-- check

select \* from sys.certificates

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

**3.3.3 Tạo Symmetric**

**Mã SQL:**

CREATE SYMMETRIC KEY NhanVienKeySym WITH ALGORITHM = AES\_256

ENCRYPTION BY CERTIFICATE NhanVien01;

GO

-- check

SELECT \* FROM sys.symmetric\_keys

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

**3.3.4 Tạo bảng để mã hóa**

**Mã SQL:**

select Manv, tennv,

pass\_encrypt = CONVERT(varbinary(256),

pass), phai, DienThoai\_encrypt = CONVERT(varbinary(256), DienThoai),

email\_encrypt = CONVERT(varbinary(256), email), mapb into NhanVien\_encrypt

from NhanVien where 1 = 2

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

**3.3.5 Truncate table**

**Mã SQL:**

Truncate table NhanVien\_encrypt

**3.3.6 Open Symmetric key**

**Mã SQL:**

OPEN SYMMETRIC KEY NhanVienKeySym DECRYPTION BY CERTIFICATE NhanVien01

**3.3.7 Insert dữ liệu và mã hóa**

**Mã SQL:**

insert NhanVien\_encrypt (tennv,

pass\_encrypt, phai, DienThoai\_encrypt, email\_encrypt, mapb)

select top 5 tennv,

pass\_encrypt = EncryptByKey(KEY\_GUID('NhanVienKeySym'), pass),

phai, DienThoai\_encrypt = EncryptByKey(KEY\_GUID('NhanVienKeySym'), DienThoai),

email\_encrypt = EncryptByKey(KEY\_GUID('NhanVienKeySym'), email), mapb

from NhanVien

-- check

select \* from NhanVien\_encrypt

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

**3.4 Giải mã bằng chứng chỉ**

**Mã SQL:** Open Symetric

USE QLDA;

OPEN SYMMETRIC KEY NhanVienKeySym DECRYPTION BY CERTIFICATE

NhanVien01 Select tennv,

pass = convert(nvarchar(25), DecryptByKey(pass\_encrypt)), phai,

DienThoai = convert(nvarchar(25), DecryptByKey(DienThoai\_encrypt)),

email = convert(nvarchar(25), DecryptByKey(email\_encrypt)), mapb

from NhanVien\_encrypt

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

**3.4.2 Close Symetric**

**Mã SQL:**

CLOSE SYMMETRIC KEY NhanVienKeySym

select name, pvt\_key\_encryption\_type, issuer\_name, subject, expiry\_date =

CAST(expiry\_date as DATE), start\_date = CAST(start\_date as DATE) from sys.certificates

go

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

**Mã SQL:**

select name, key\_length, key\_algorithm, algorithm\_desc, create\_date = CAST(create\_date as

DATE), modify\_date = CAST(create\_date as DATE), key\_guid from sys.symmetric\_keys

go

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

**3.5 Xóa Symetric Key và Certificate nếu không sử dụng**

**Mã SQL:**

DROP SYMMETRIC KEY NhanVienKeySym

DROP CERTIFICATE NhanVien01

**Kết quả:**

**A screenshot of a computer

Description automatically generated**

**4) Thực hiện mã hóa và giải mã dữ liệu trong suốt cho cơ sở dữ liệu QLDA**

**4.1). Create the master key which is stored in the master database**

**Mã SQL:**

USE master;

GO

IF NOT EXISTS (SELECT \* FROM sys.symmetric\_keys WHERE symmetric\_key\_id =

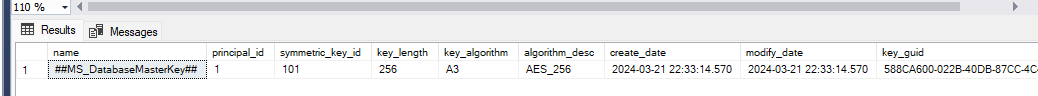
101) CREATE MASTER KEY ENCRYPTION BY PASSWORD =

'Th15i$aS7riN&ofR@nD0m!T3%t'

-- check

SELECT \* FROM sys.symmetric\_keys

**Kết quả:**



**4.2). Create a certificate**

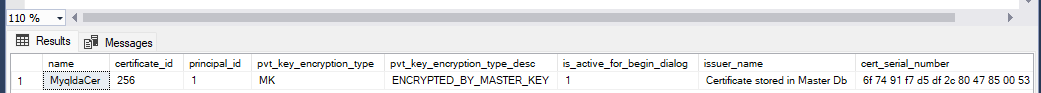
**Mã SQL:**

CREATE CERTIFICATE MyqldaCer with SUBJECT = 'Certificate stored in Master Db'

-- check

SELECT \* FROM sys.symmetric\_keys

**Kết quả:**



**4.3). Tạo Khóa mã hóa cơ sở dữ liệu (DEK) dựa trên trên chứng chỉ đã tạo trước đó**

**Mã SQL:**

USE QLDA

GO

CREATE DATABASE ENCRYPTION KEY WITH ALGORITHM = AES\_256

ENCRYPTION BY SERVER CERTIFICATE MyqldaCer

GO

**Kết quả:**

**4.4). Bật mã hóa cho QLDA**

**Mã SQL:**

ALTER DATABASE QLDA SET ENCRYPTION ON

GO

**Kết quả:**

**4.5). Xem quá trình mã hóa**

**Mã SQL:**

SELECT DBName = DB\_NAME(database\_id), encryption\_state FROM sys.dm\_database\_encryption\_keys

**Kết quả:**