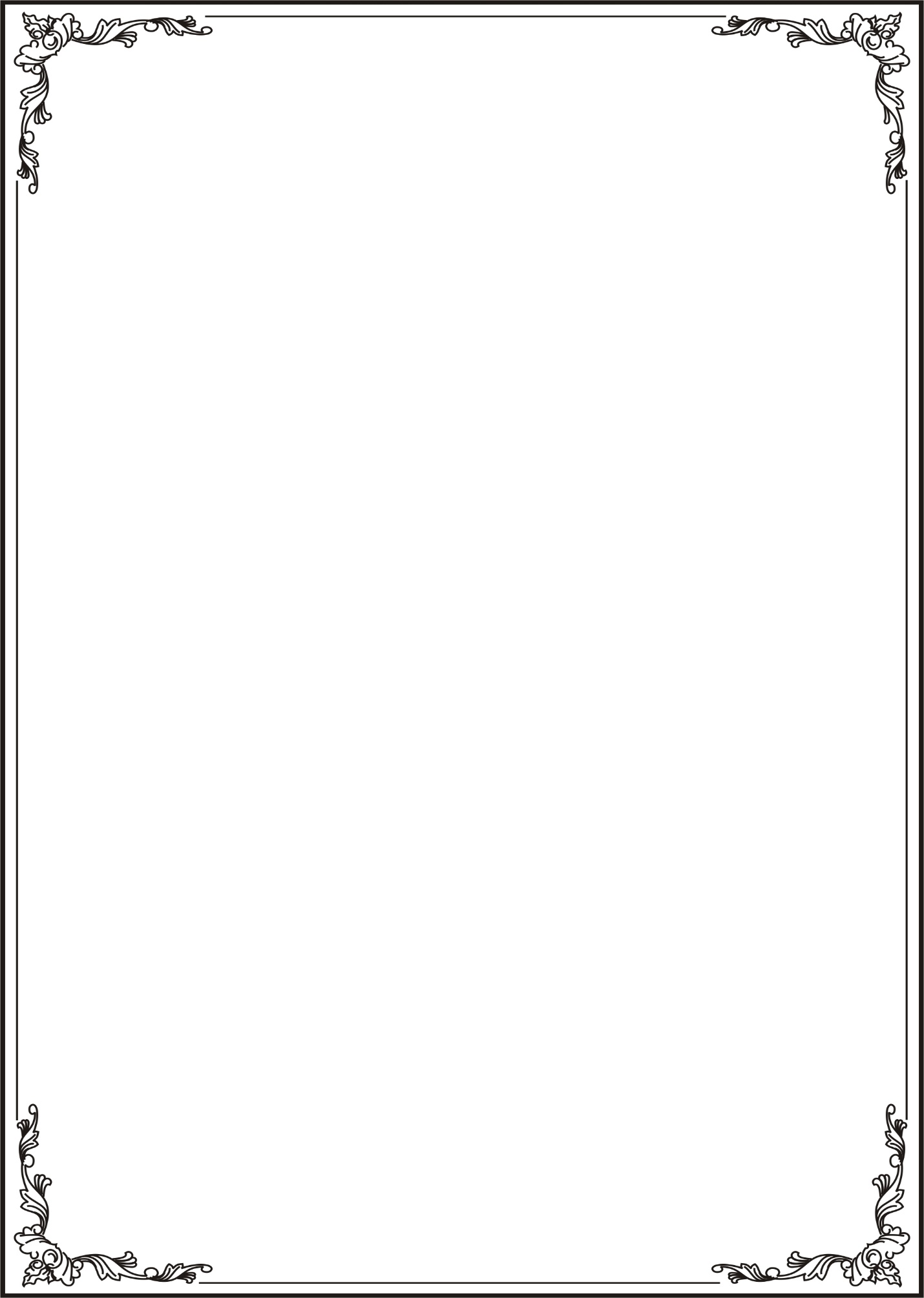
****

**ĐẠI HỌC ĐÀ NẴNG**

**TRƯỜNG ĐẠI HỌC KINH TẾ**

🙣 🕮 🙡

****

***Báo cáo***

***Hệ Thống Quản Lý Bán Hàng***

***Đơn vị báo cáo: Quán Cơm Gà Gia Vĩnh***

***Môn: Quản Trị Cơ Sở Dữ Liệu***

*Lớp sinh hoạt: 47K14*

|  |  |
| --- | --- |
| ***Giảng viên hướng dẫn***  **Cao Thị Nhâm** | ***Nhóm 10***  Nguyễn Thị Kim Anh  Nguyễn Hữu Trường Giang  Dương Tuyết Nhi  **Phạm Thị Minh Nguyệt**  Nguyễn Huy Tuấn Vũ |
|  |  |

*Đà Nẵng, tháng 11 năm 2023*

**MỤC LỤC**

[1.Hóa Đơn: 3](#_Toc152244424)

[2. Thiết kế cơ sở dữ liệu: 4](#_Toc152244425)

[2.1 Mức khái niệm: 4](#_Toc152244426)

[2.2 Thiết kế CSDL mức logic: 7](#_Toc152244427)

[2.3 Mức vật lý 8](#_Toc152244428)

[3. Xây dựng cơ sở dữ liệu: 10](#_Toc152244429)

[4. Xác định và tạo index: 10](#_Toc152244430)

[5. Xây dựng cơ chế bảo mật cho cơ sở dữ liệu: 10](#_Toc152244431)

[6. Xây dựng cơ chế backup dữ liệu cho cơ sở dữ liệu: 15](#_Toc152244432)

[7. Sử dụng một ngôn ngữ bất kì để phát triển một ứng dụng mang tên TESTDB thực hiện chức năng: thêm, sửa, xóa và tìm kiếm cho CSDL vừa tạo. 21](#_Toc152244433)

[8. Phương án giải quyết tốt nhất cho việc lượng dữ liệu rất lớn, không đủ lưu trữ trong một ổ đĩa. 21](#_Toc152244434)

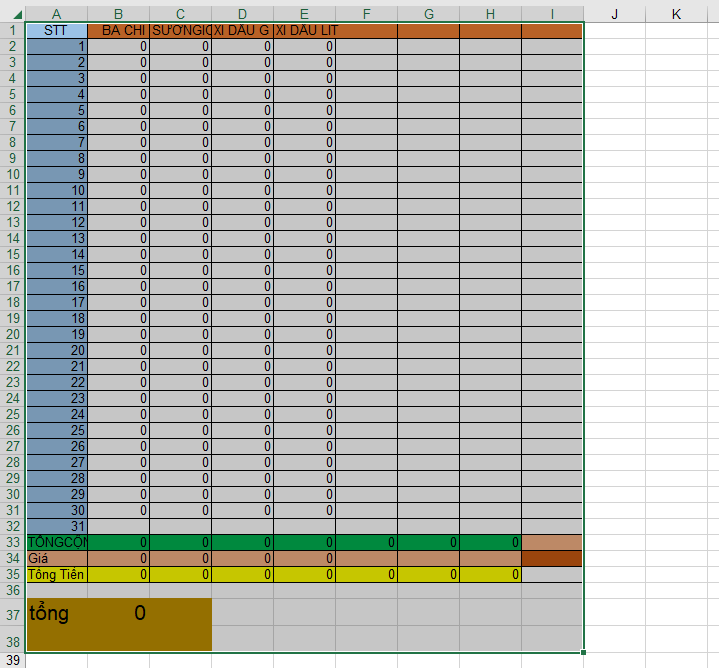
[9. Phương án tốt nhất để đề phòng phương án hệ thống TESTDB bị tấn công bằng SQL Injection. 22](#_Toc152244435)

[PHỤ LỤC 25](#_Toc152244436)

# 1.Hóa Đơn:







# 2. Thiết kế cơ sở dữ liệu:

## 2.1 Mức khái niệm:

#### 2.1.1 Hóa đơn Nhập Hàng:

* Bước 1: Chọn lọc thông tin

|  |  |  |
| --- | --- | --- |
| Từ trong hóa đơn | Từ rõ nghĩa | Từ viết tắt |
| ~~Số~~ | ~~mã đơn nhập hàng~~ | ~~MaNH~~ |
| ~~Tên đơn vị~~ | ~~tên đơn vị nhập hàng~~ | ~~TenDVNH~~ |
| ~~Địa chỉ~~ | ~~địa chỉ đơn vị nhập~~ | ~~DiaChiNH~~ |
| Thời Gian | thời gian nhập hàng | ThoiGian |
| ~~Tên sản phẩm~~ | ~~tên sản phẩm~~ | ~~TenNL~~ |
| ~~Đơn giá~~ | ~~đơn giá~~ | ~~DongiaNH~~ |
| ~~Đơn vị~~ | ~~đơn vị tính~~ | ~~DVT~~ |
| Số lượng | số lượng nhập hàng | SoLuong |
| Thành tiền | thành tiền nhập hàng | ThanhTien |

* Bước 2: Xác định thực thể thuộc tính

+ Đơn vị nhập (MaDV, TenDVNH, DiaChiNH)

+ Nguyên liệu (MaNL, TenNL, DVT, DonGiaNH)

* Bước 3: Xác định quan hệ
* Nhập (MaNH, ThoiGian, SoLuong, ThanhTien, MaNL, MaDV)
* Bước 4: Vẽ sơ đồ quan hệ

A diagram of a network

Description automatically generated

#### 2.1.2 Hóa đơn Bán hàng:

Bước 1: Chọn lọc thông tin:

|  |  |  |
| --- | --- | --- |
| Từ trong HSDL | Từ rõ nghĩa | Viết tắt |
| số hóa đơn | mã hóa đơn | MaHD |
| ~~Bàn~~ | ~~mã bàn~~ | ~~MaBan~~ |
| Thời gian | thời gian đặt hang | ThoiGian |
| ~~Tên món~~ | ~~tên món~~ | ~~TenMon~~ |
| ~~Đơn giá~~ | ~~đơn giá~~ | ~~Dongia~~ |
| Số lượng | số lượng | Soluong |
| thành tiền | thành tiền hóa đơn | ThanhTien |

Bước 2: Xác định thực thể thuộc tính:

* Bàn (MaBan,TrangThai)
* Món ăn (MaMA, TenMon, DonGia)

Bước 3: Xác định quan hệ:

* Bán (MaHD, SDT, DiaChi, ThoiGian, SoLuong, ThanhTien, MaBan, MaMA)

Bước 4: Vẽ sơ đồ quan hệ:

A diagram of a diagram

Description automatically generated

#### 2.1.3 Hóa đơn ER tích hợp:

Several blue and orange objects

Description automatically generated

## Thiết kế CSDL mức logic:

* Chuyển thực thể thu được:
* Đơn vị nhập (MaDV, TenDVNH, DiaChiNH)
* Nguyên liệu (MaNL, TenNL, DVT, DonGiaNH)
* Bàn (MaBan, TrangThai)
* Món ăn (MaMA, TenMon, DonGia)
* Chuyển quan hệ thu được:
* Nhập (NgayNH, SoLuongNH, ThanhTienNH, MaNL, MaDV)
* Bán (MaHD, SDT, DiaChi, ThoiGian, SoLuong, ThanhTien, MaBan, MaMA)
* Chuẩn hóa:
* Chuẩn hóa quan hệ Nhập, thu được:

NhapHang (ThoiGian, SoLuong, MaNL, MaDV)

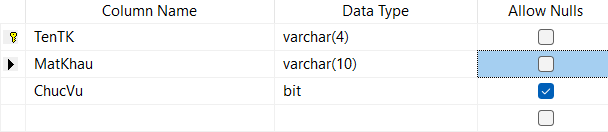
* Chuẩn hóa quan hệ Bán, thu được:
* HoaDon (MaHD, ThoiGian, MaMA, MaBan)
* HoaDonChiTiet (MaHDCT, TenMon, SoLuong, MaHD)
* Sơ đồ quan hệ:

A diagram of a company

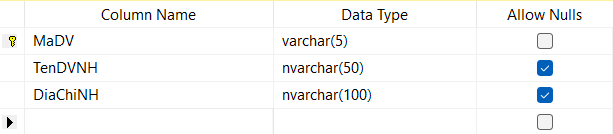
Description automatically generated

## 2.3 Mức vật lý

* Tài khoản:



* Đơn vị nhập:

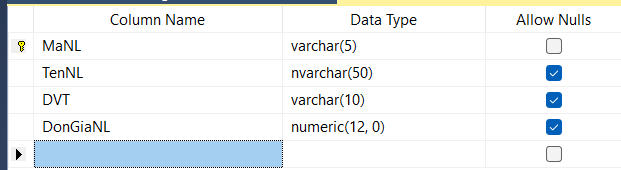


* Nhập hàng:

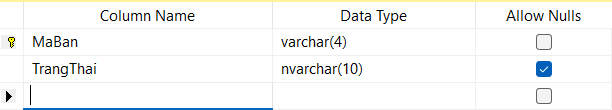
A screenshot of a data type

Description automatically generated

* Nguyên liệu:



* Bàn:

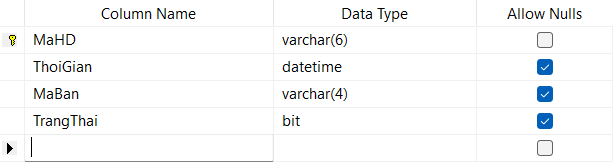


* Món ăn:

A screenshot of a data

Description automatically generated

* Hóa đơn



* Hóa đơn chi tiết

A screenshot of a computer

Description automatically generated

# 3. Xây dựng cơ sở dữ liệu:

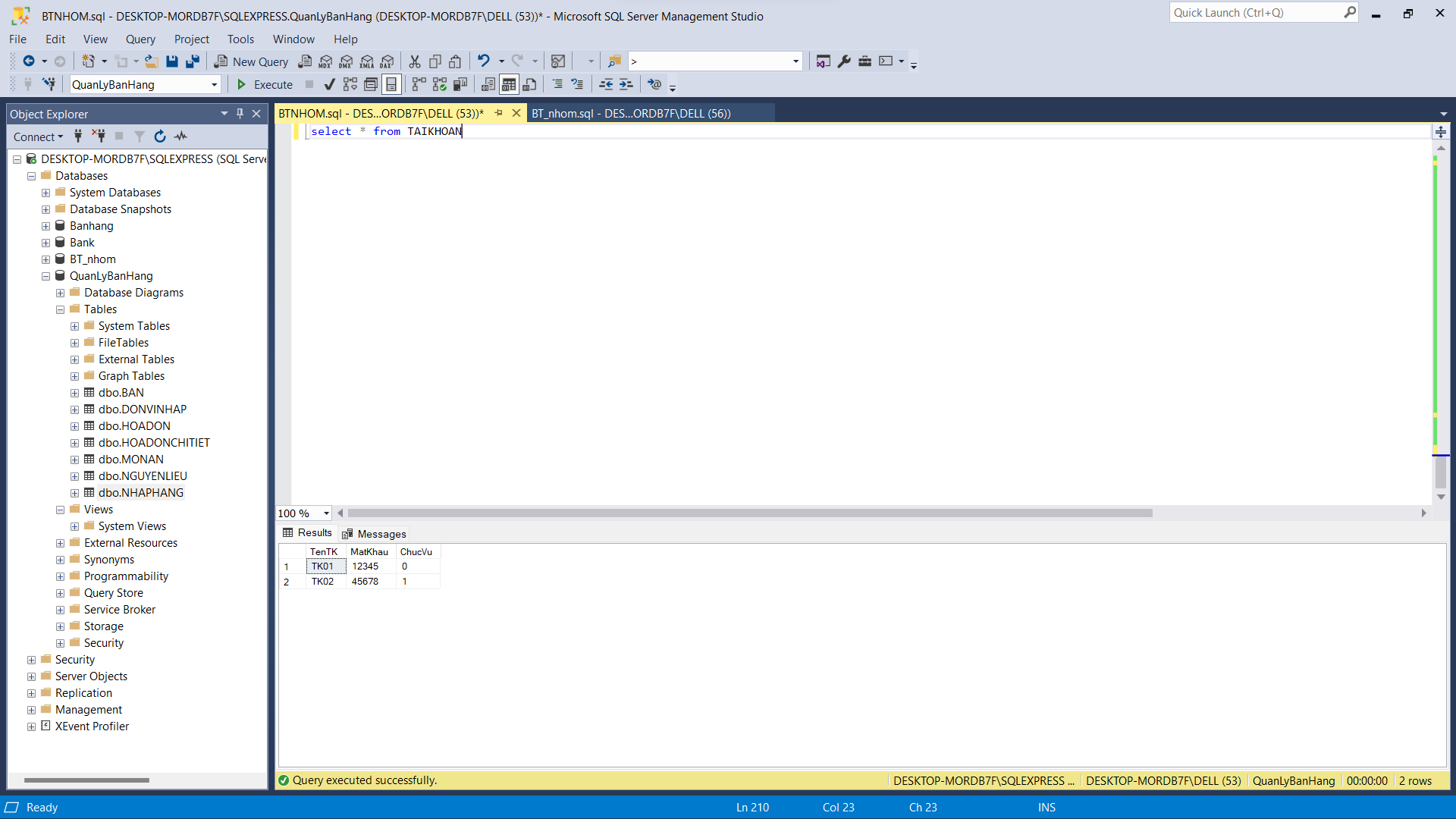
* Script tạo cơ sở dữ liệu nằm ở phần phụ lục.

# 4. Xác định và tạo index:

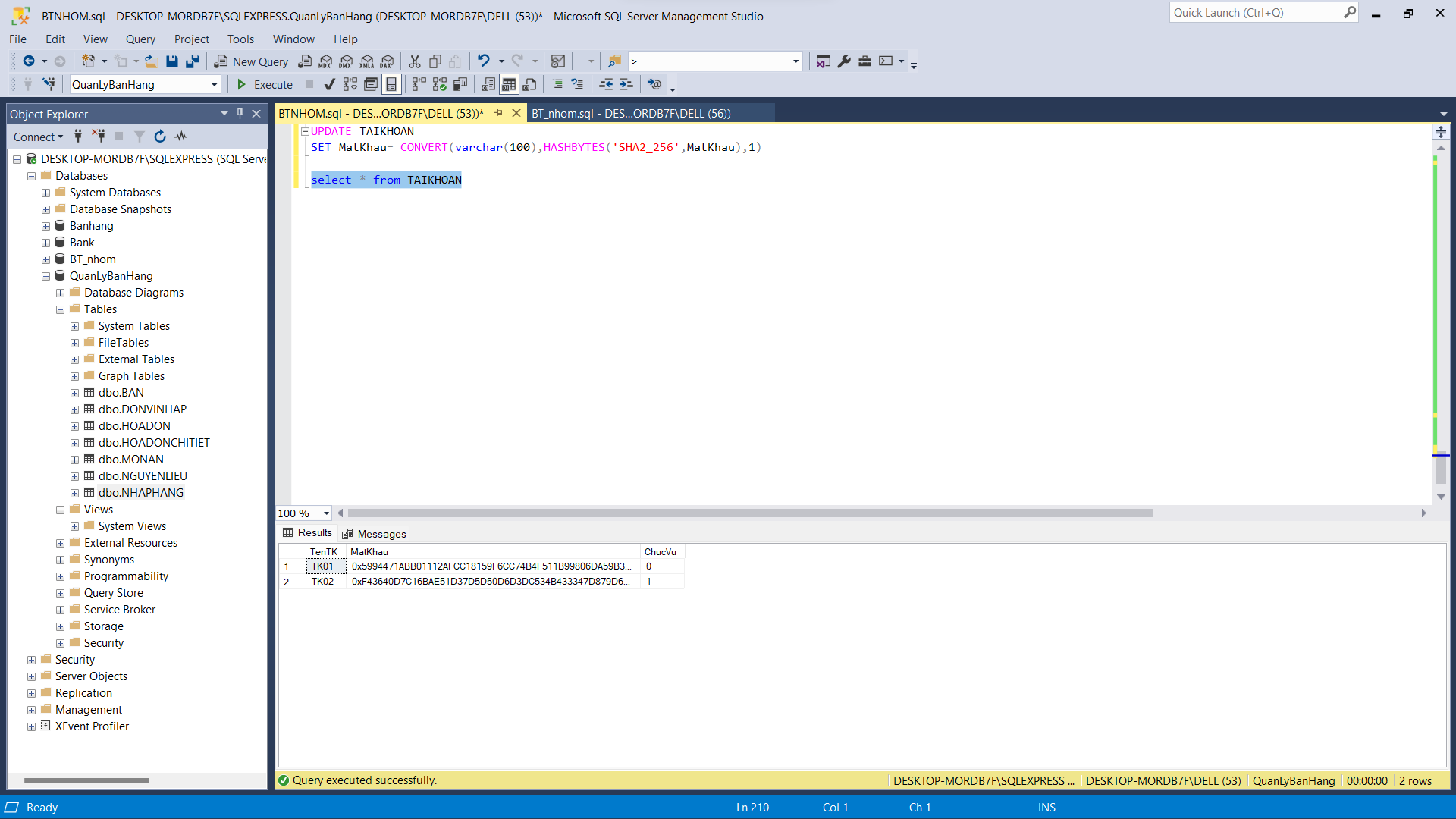
* Tạo index cho Bảng MONAN
* Cột được chọn để tạo non-clustered index: MONAN(TenMon)
* Lý do:
  + Bảng MONAN (TenMon) là một bảng có tần suất chỉnh sửa ít.
  + Món ăn được chia thành 4 loại: Cơm, mì, gà, cải=> Khi tạo index cho cột TenMon, các hàng hóa cùng 1 loại sẽ được sắp xếp gần nhau => Giúp hệ thống truy xuất dữ liệu nhanh hơn.
* Cú pháp: CREATE INDEX MonAn ON MONAN(TenMon)
* Tạo index cho Bảng NGUYENLIEU
* Cột được chọn để tạo non-clustered index: NGUYENLIEU(TenNL)
* Lý do:
  + Bảng NGUYENLIEU(TenNL)là một bảng có tần suất chỉnh sửa ít.
  + Món ăn được chia thành 4 loại: thịt, mì, rau, khác=> Khi tạo index cho cột TenNL, các hàng hóa cùng 1 loại sẽ được sắp xếp gần nhau => Giúp hệ thống truy xuất dữ liệu nhanh hơn.
* Cú pháp: CREATE INDEX NguyenLieu ON NGUYENLIEU(TenNL)

# 5. Xây dựng cơ chế bảo mật cho cơ sở dữ liệu:

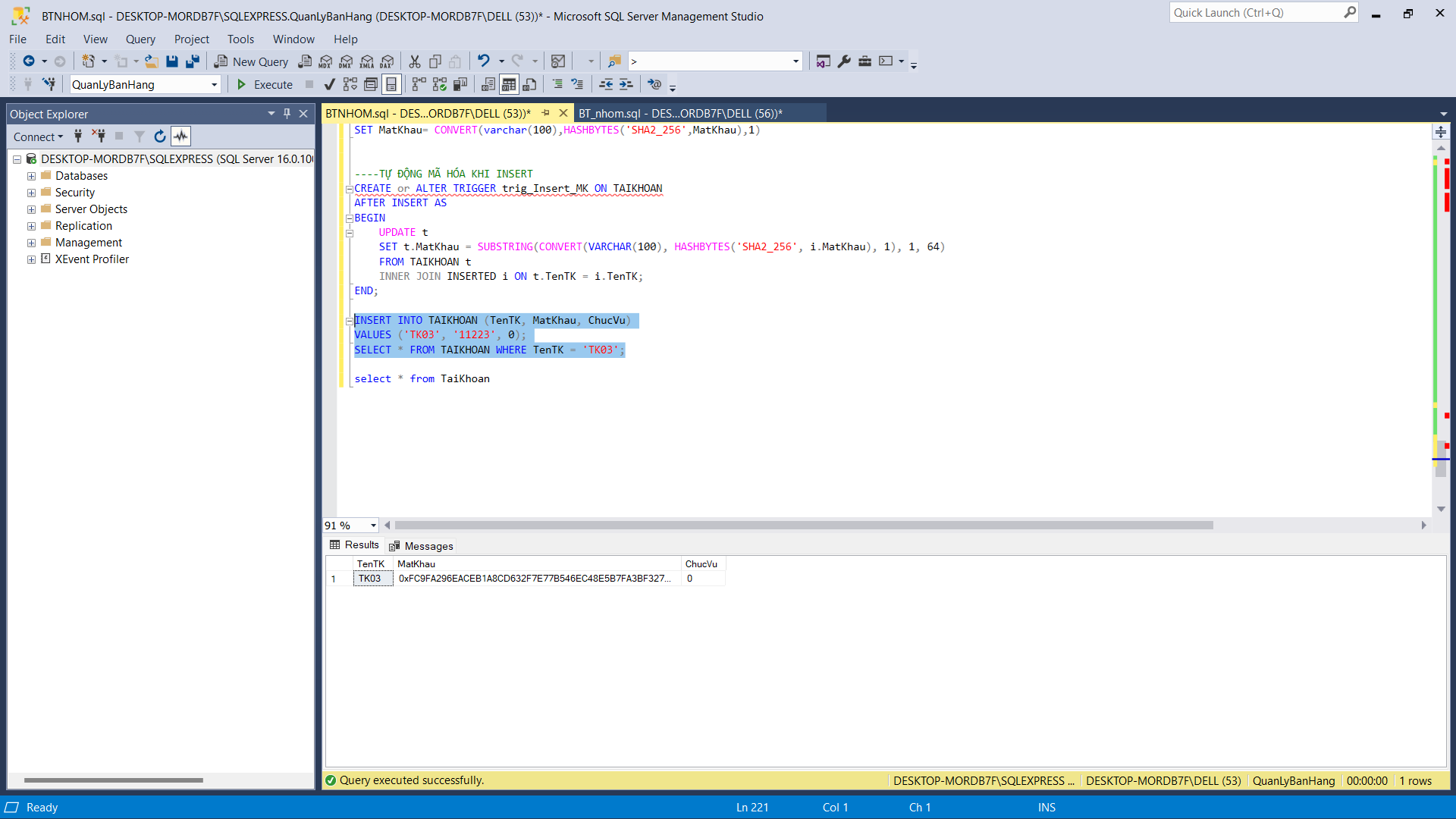
* Cơ chế bảo mật: Mã hóa bằng mã hóa một chiều hash (SHA2\_256)
* Lý do nhóm chọn giải pháp bảo mật nàyTrước khi mã hóa:
* Khi người dùng tạo hoặc thay đổi mật khẩu, hệ thống sẽ không lưu trữ mật khẩu đó mà thay vào đó sẽ lưu trữ giá trị hash của mật khẩu đó, được tạo ra bởi hàm SHA-256. Khi người dùng đăng nhập, hệ thống sẽ lấy mật khẩu nhập vào, chuyển nó thành giá trị hash bằng cách sử dụng SHA-256, và so sánh nó với giá trị hash đã lưu trữ. Nếu chúng khớp, mật khẩu được xác nhận và người dùng được đăng nhập.
* Cung cấp độ dài hash lớn, tăng cường độ an toàn. Hiện là một trong những tiêu chuẩn bảo mật cao, thích hợp cho nhiều ứng dụng khác nhau.
* Với cơ sở dữ liệu trong bài báo cáo này, nhóm sử dụng mã hóa một chiều SHA2\_246 để mã hóa mật khẩu của các tài khoản.
* Thực hiện xây dựng cơ chế mã hóa cột MatKhau trong bảng TaiKhoan.
* Trước khi mã hóa:



* Sau khi mã hóa:



* Cách 1: Tạo trigger tự động
* Tạo trigger tự động mã hóa khi insert dữ liệu vào bảng:



* Tài khoản được Insert vào và mật khẩu tự động chuyển hóa.
* Tạo trigger tự động mã hóa khi insert dữ liệu vào bảng:
  + Trước khi update:

A computer screen with text

Description automatically generated

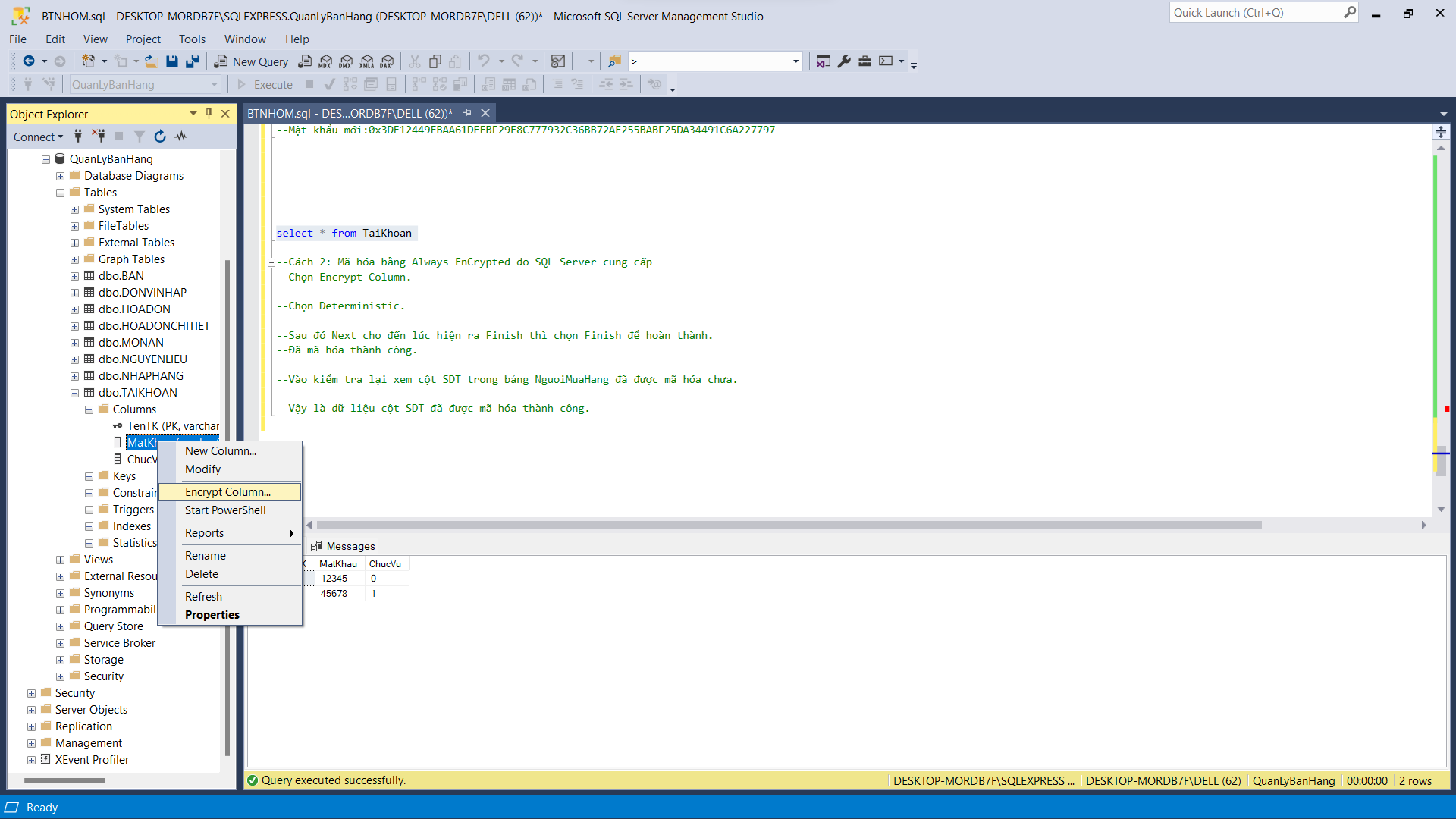
* + Sau khi update:

A computer screen with text

Description automatically generated

* Cách 2: Mã hóa bằng Always EnCrypted do SQL Server cung cấp

Bước 1: Chọn cột cần mã hóa rồi chọn Encrypt Column.



Bước 2: Chọn Deterministic.

A screenshot of a computer

Description automatically generated

Bước 3: Sau đó Next cho đến lúc hiện ra Finish thì chọn Finish để hoàn thành. 🡪 Đã mã hóa thành công.

A screenshot of a computer

Description automatically generated

Bước 4: Vào kiểm tra lại xem cột MatKhau trong bảng TaiKhoan đã được mã hóa chưa.

A computer screen with a white box

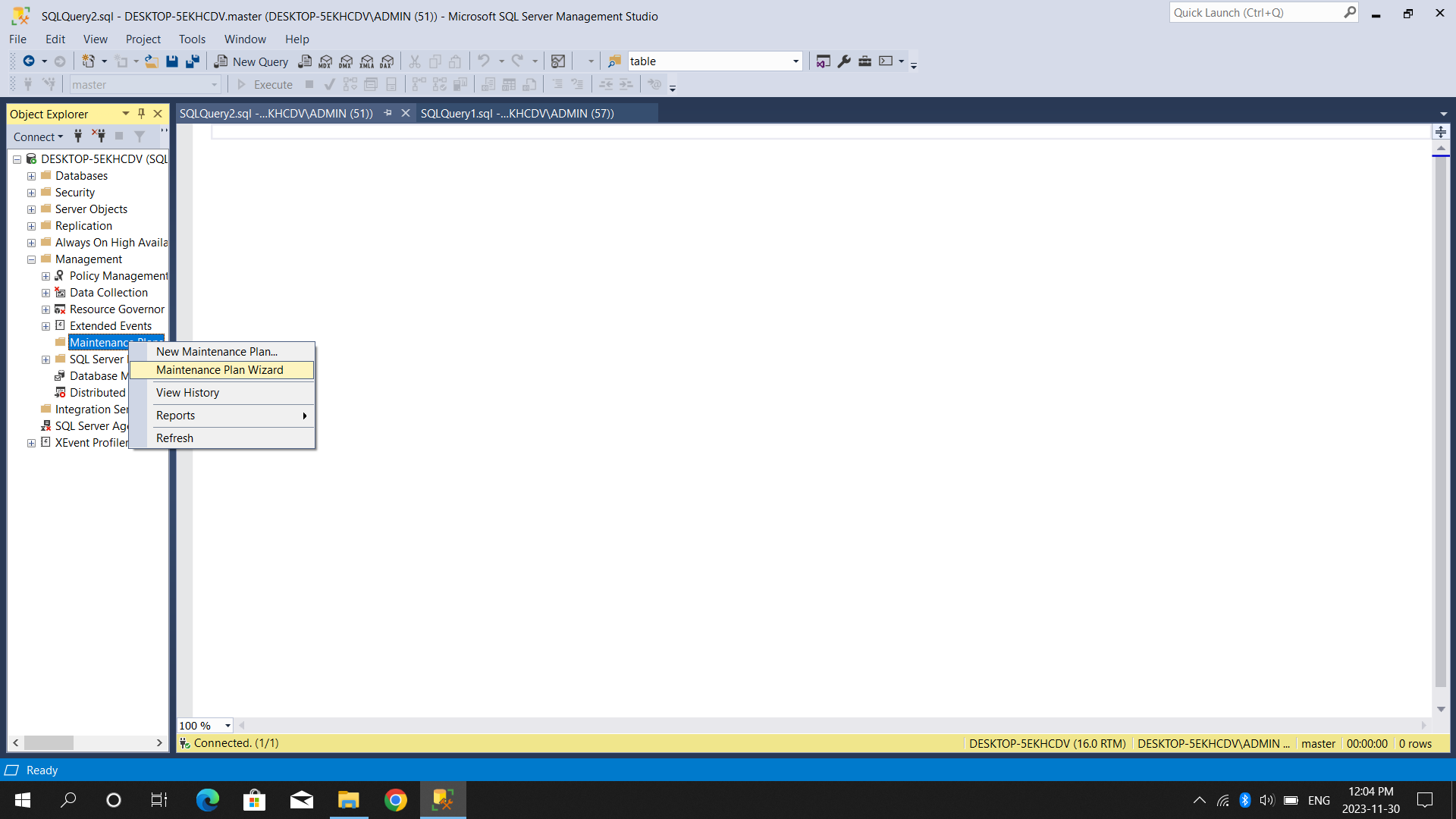
Description automatically generated

**🡪Vậy là dữ liệu cột MatKhau đã được mã hóa thành công.**

# 6. Xây dựng cơ chế backup dữ liệu cho cơ sở dữ liệu:

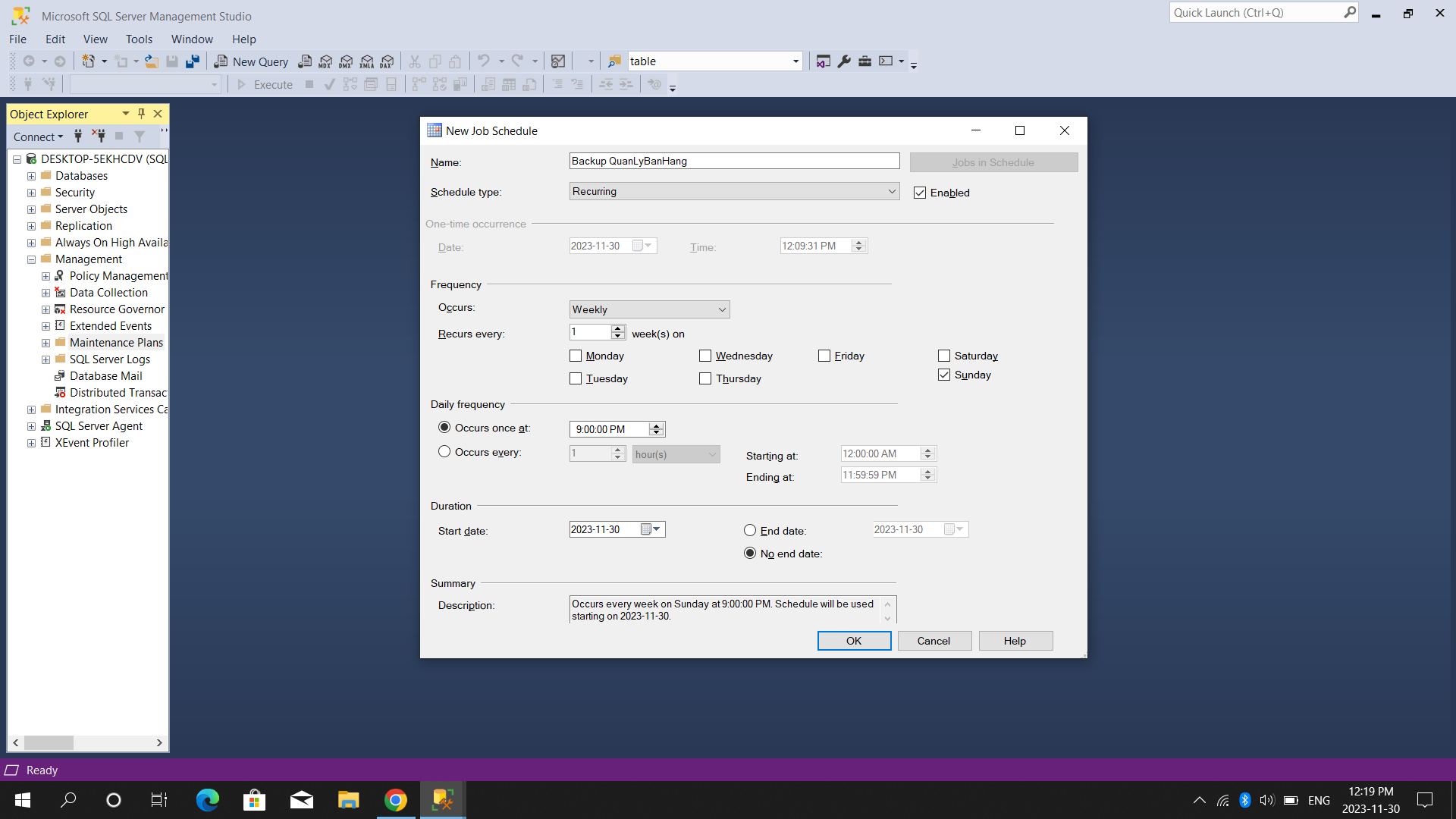
* **BACKUP**

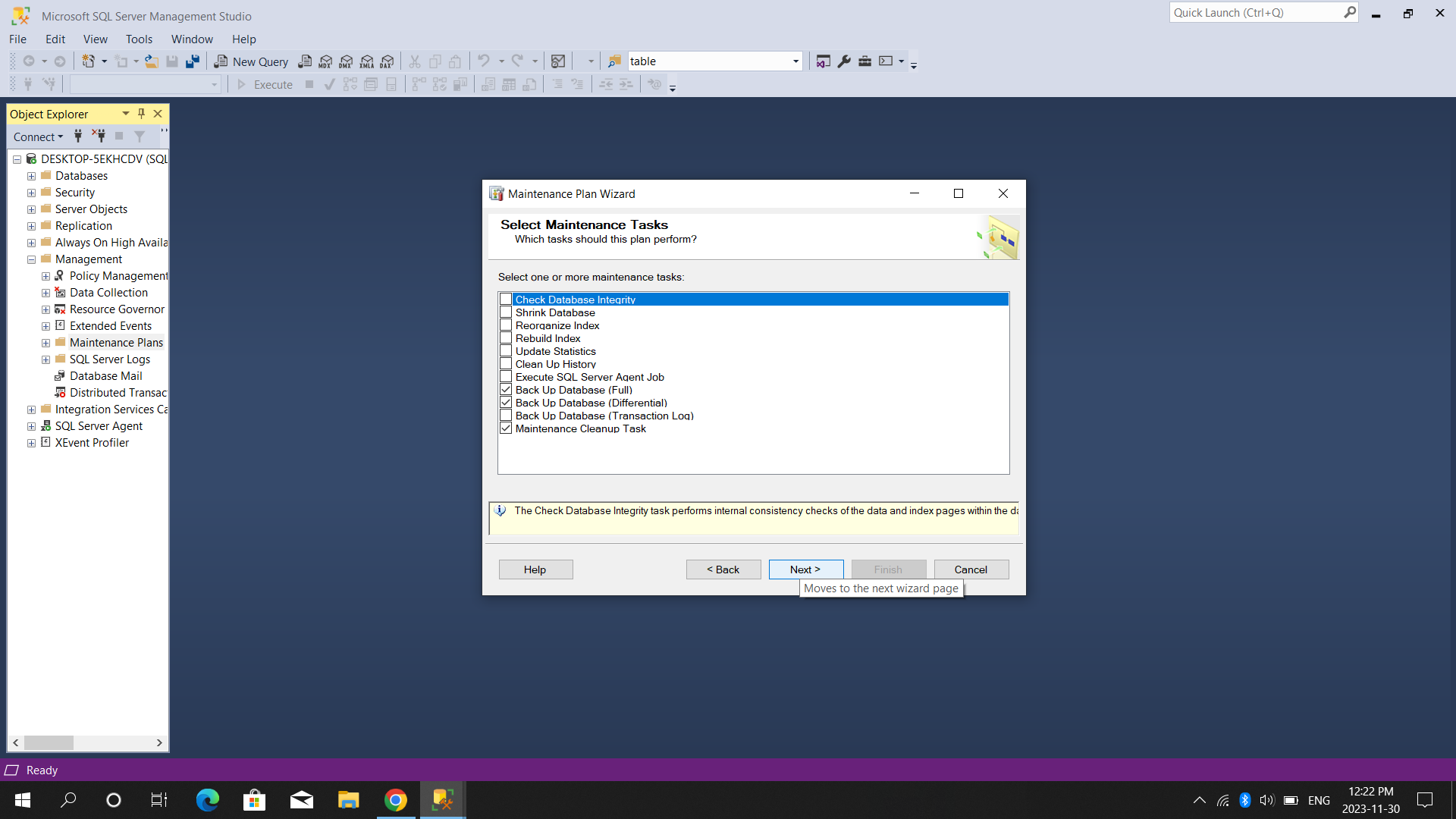
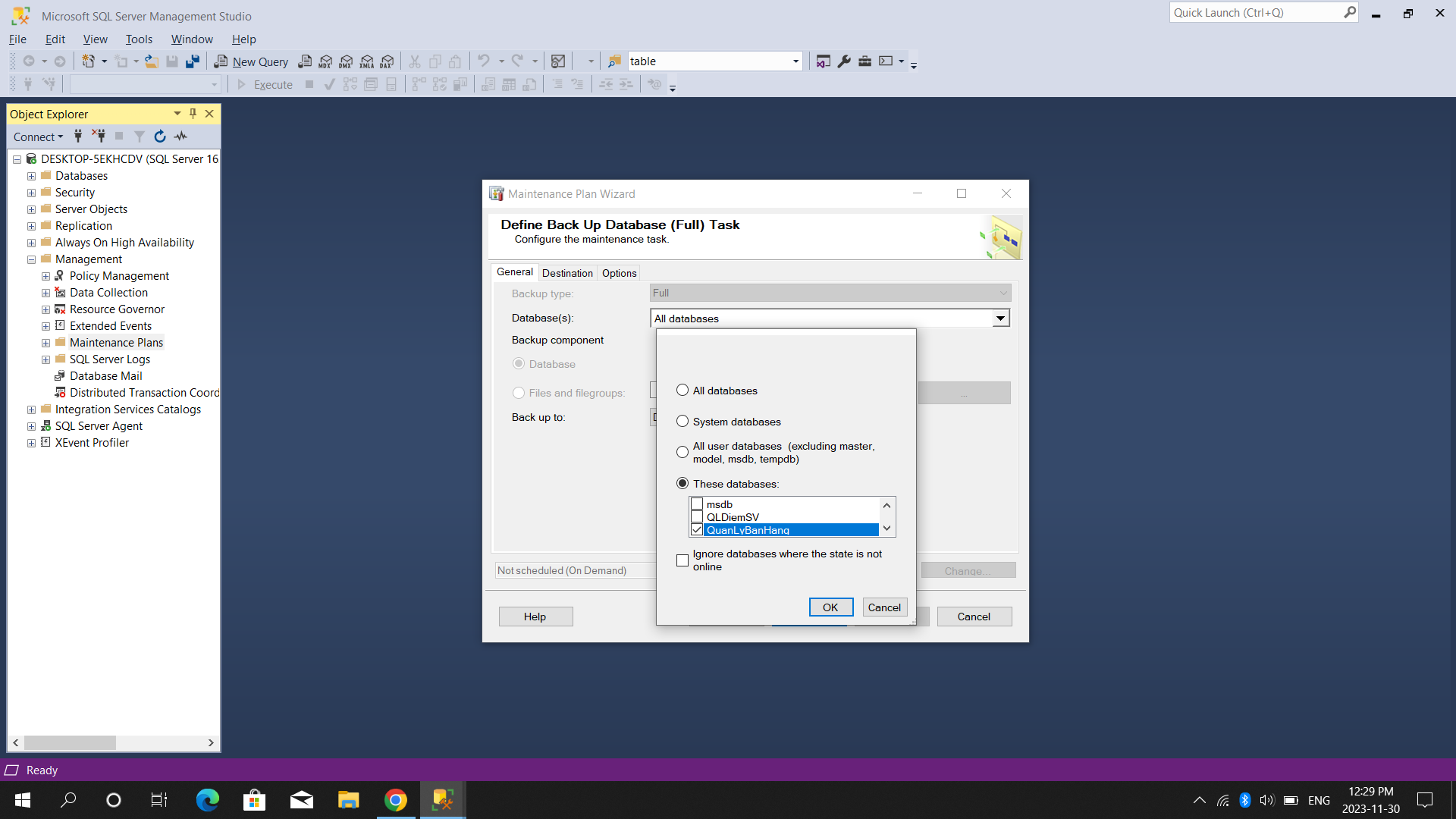
+ Bước 1: Lựa chọn thao tác backup:

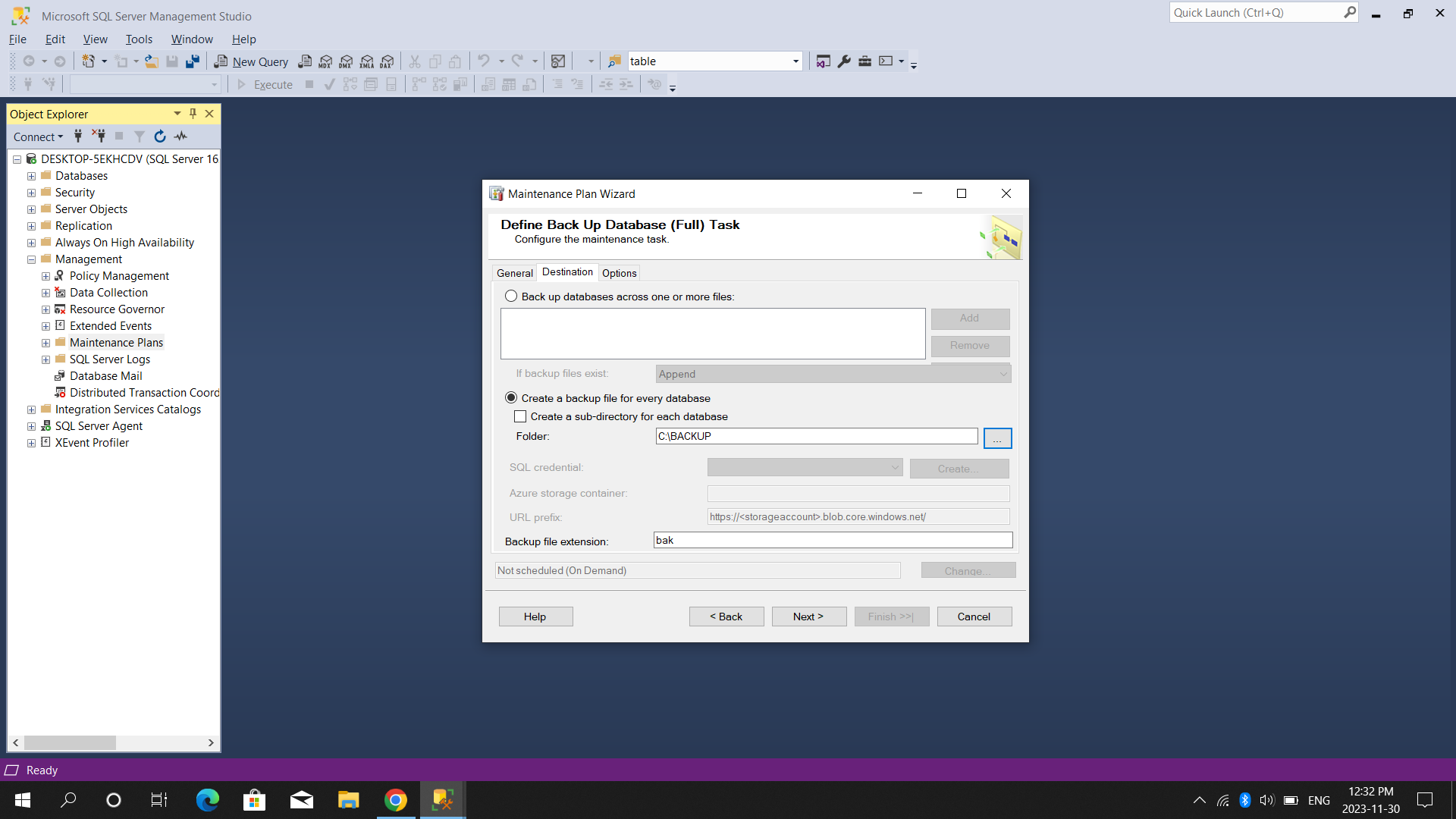


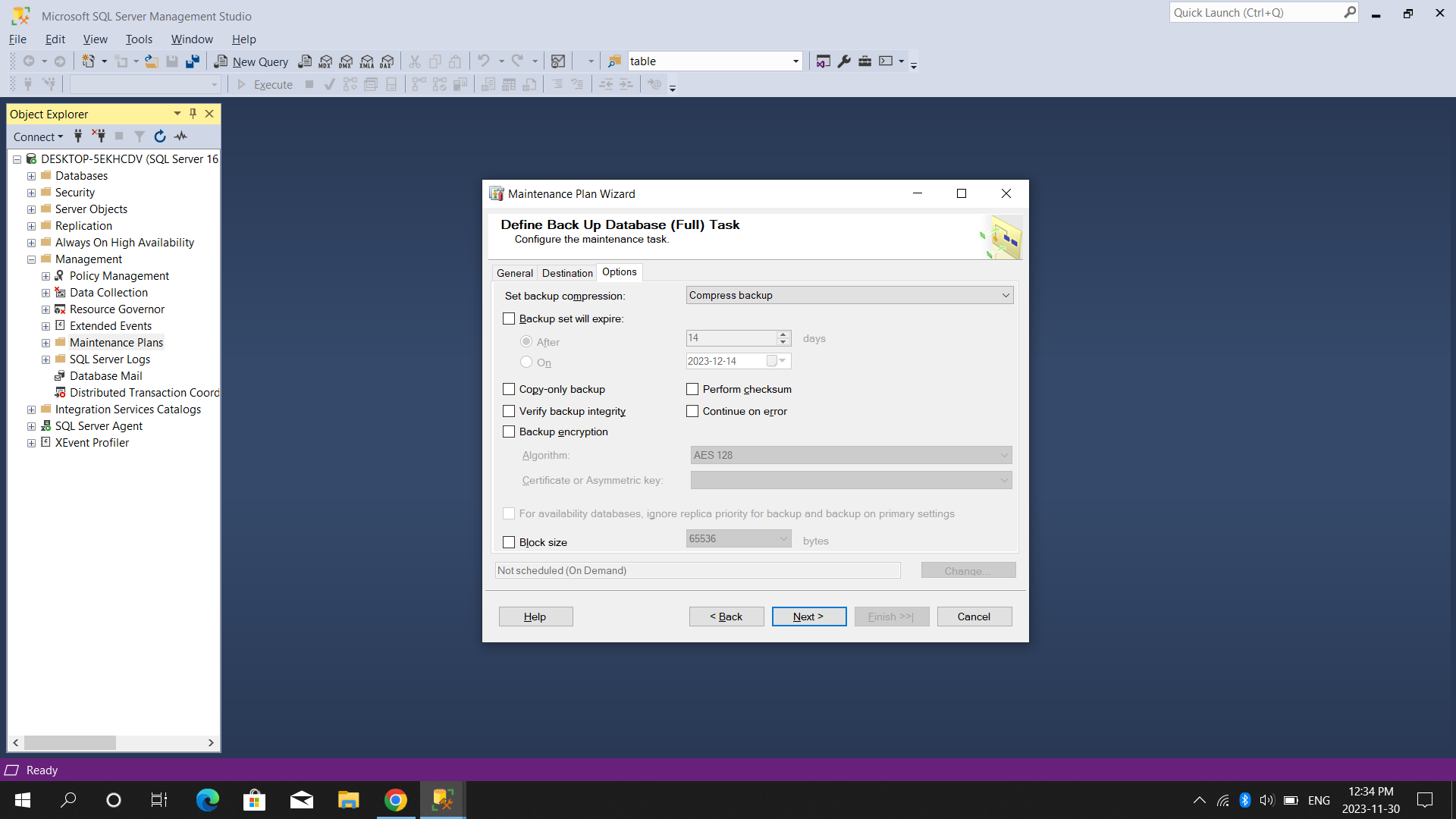
+ Bước 2: Thiết lập các tùy chọn backup:

* Database: lựa chọn tên cơ sở dữ liệu backup
* Backup type: có thể lựa chọn Full hoặc Differential (chỉ lựa chọn differential khi đã thực hiện ít nhất một lần full trước đó).
* Backup To: có thể lựa chọn lưu bản backup vào ổ đĩa cứng hoặc lưu lên một máy tính khác trên internet.

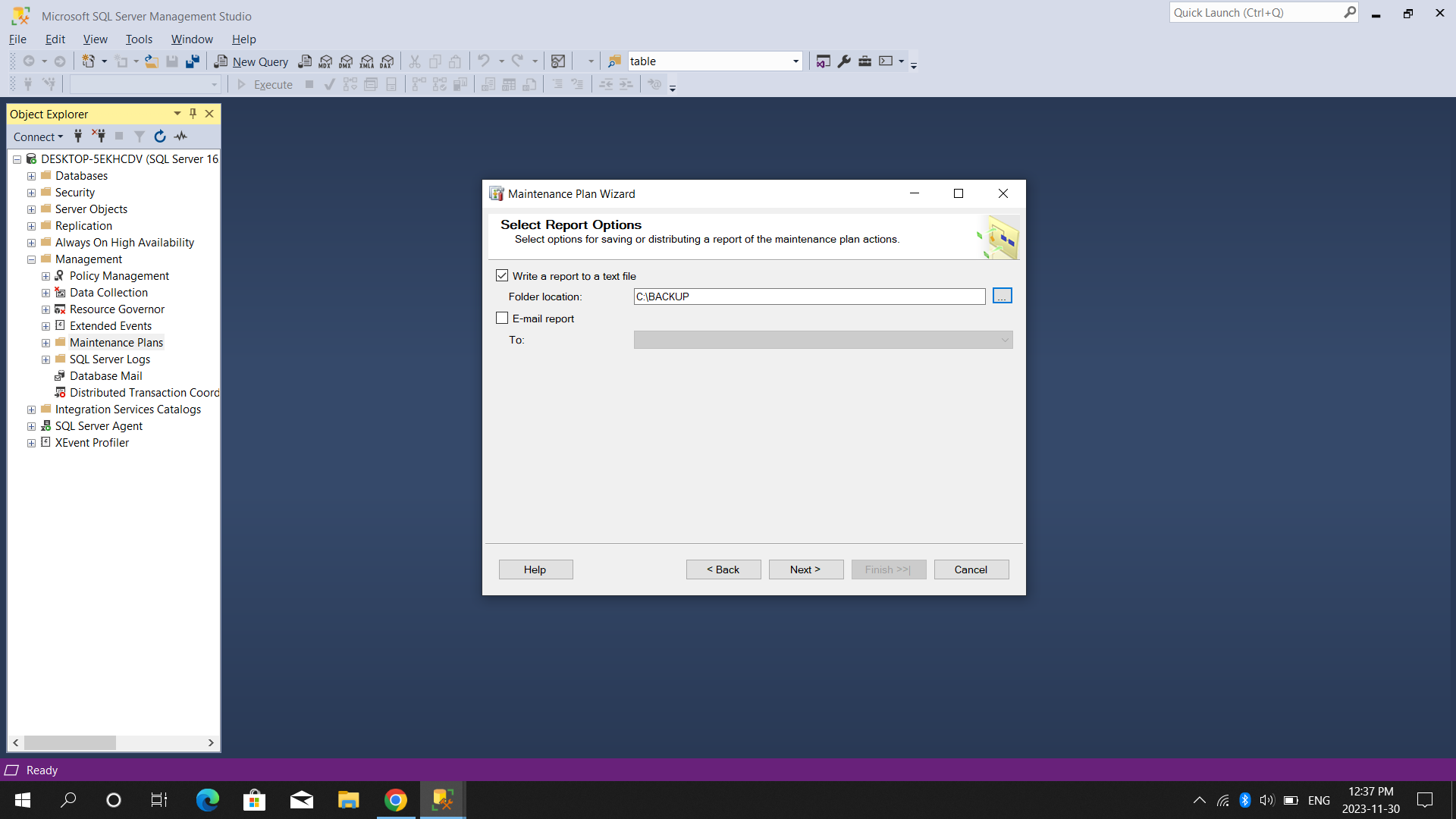


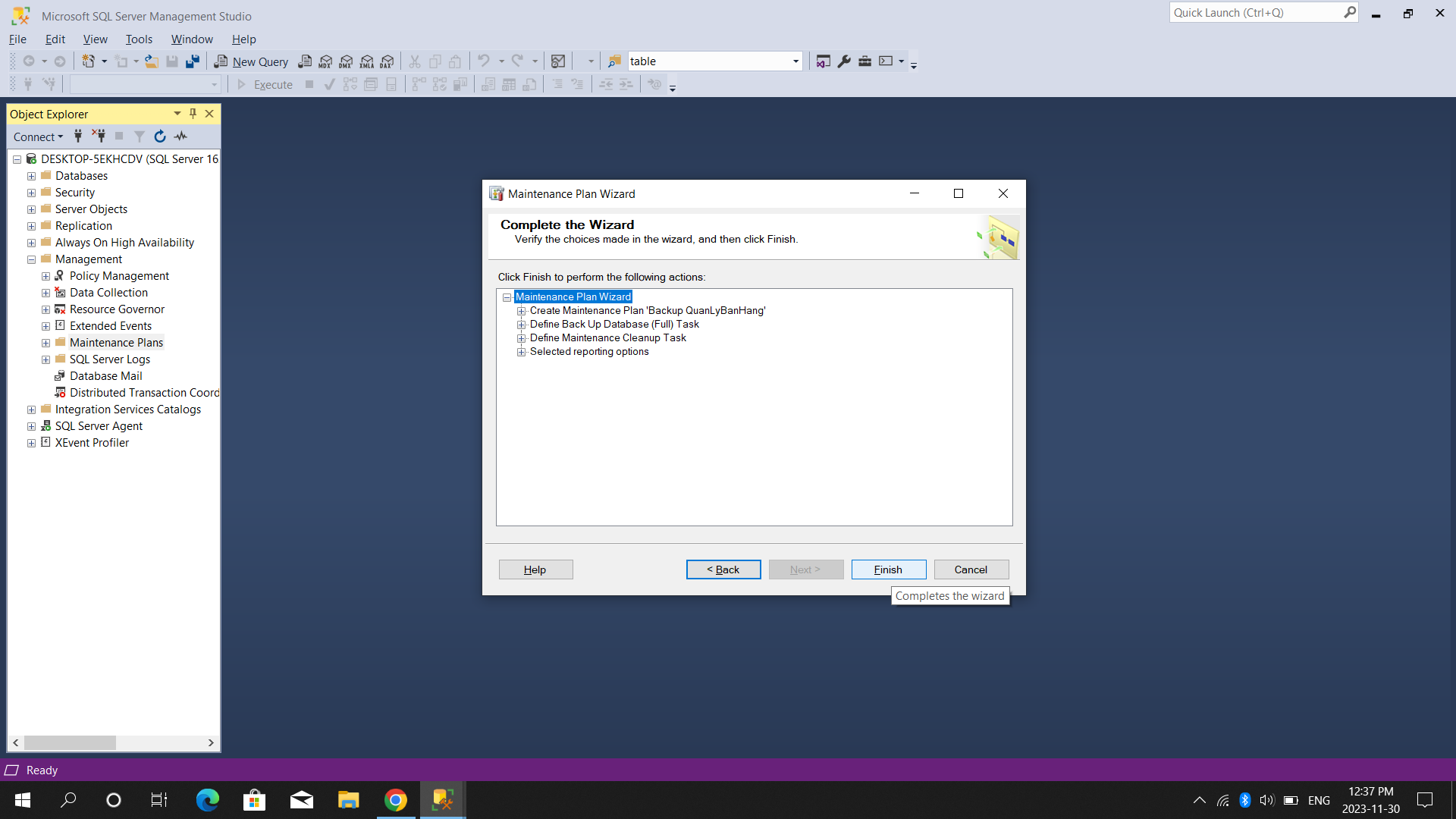


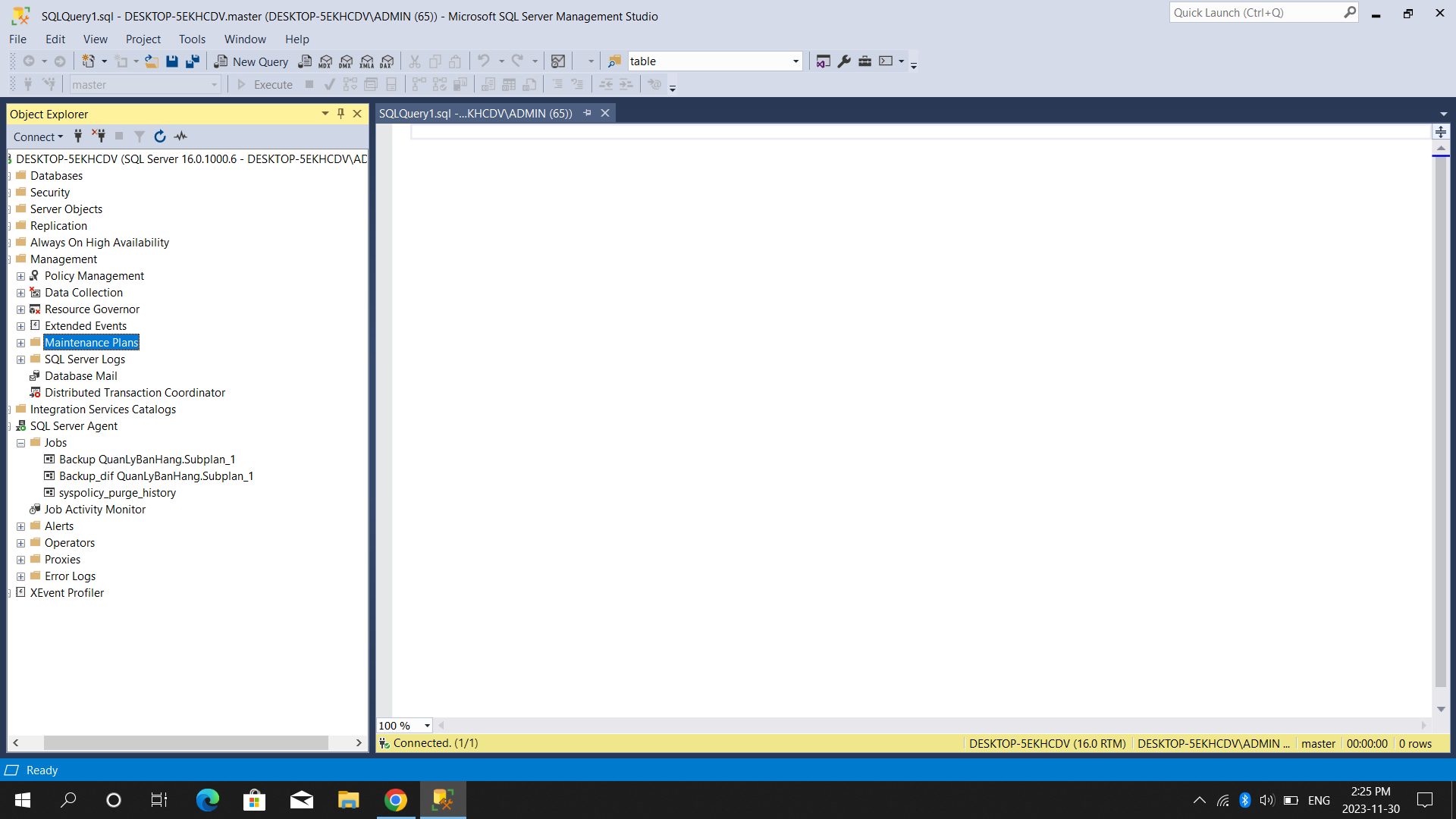




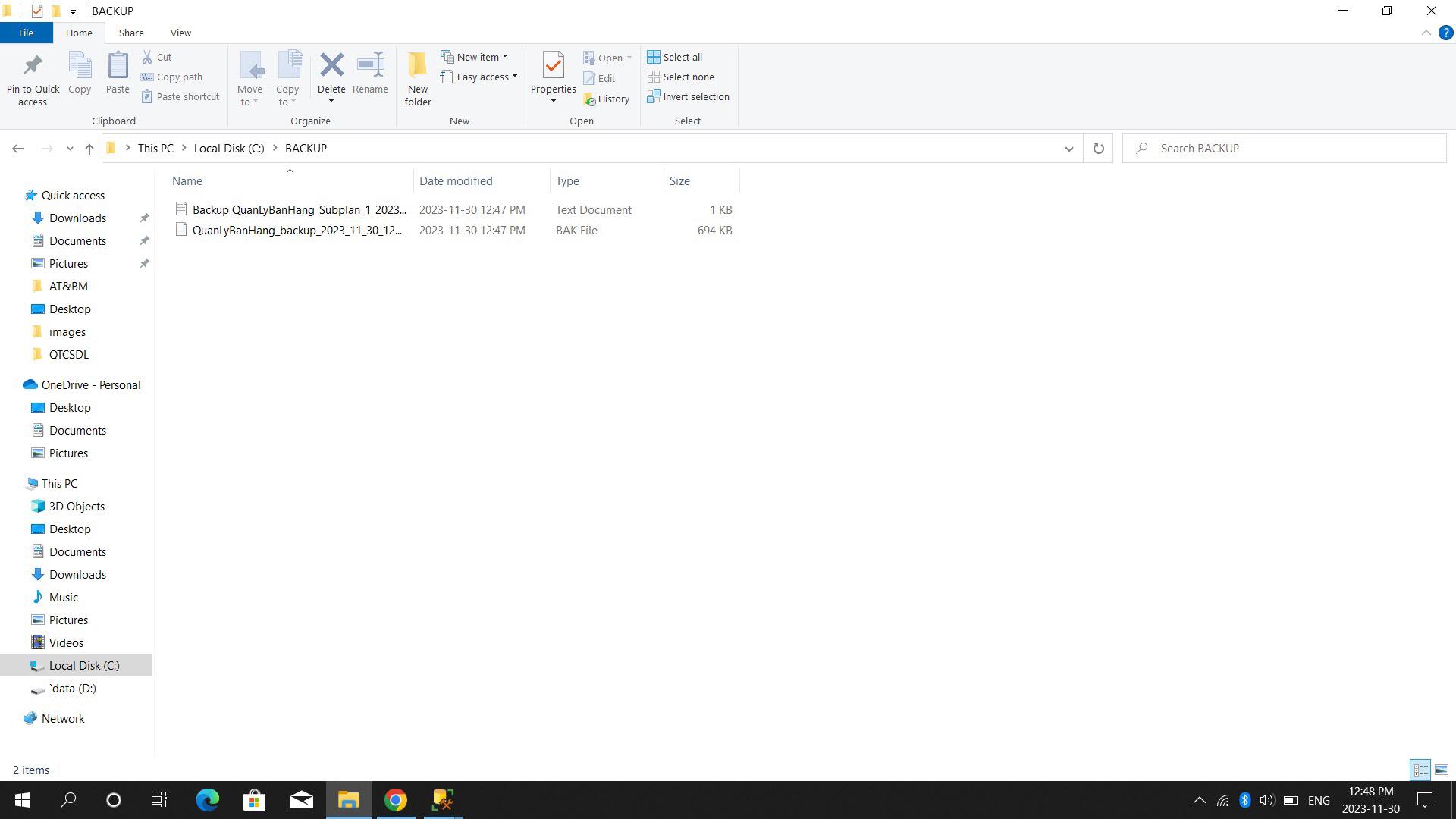


+ Bước 3: Nhấn finish hiển thị đã Backup thành công





+ Bước 4: Kiểm tra xem đã có folder BACK UP hay chưa



# 7. Sử dụng một ngôn ngữ bất kì để phát triển một ứng dụng mang tên TESTDB thực hiện chức năng: thêm, sửa, xóa và tìm kiếm cho CSDL vừa tạo.

-- Source Code

[MinhNguyt/HeThongQuanLyBanHang (github.com)](https://github.com/MinhNguyt/HeThongQuanLyBanHang)

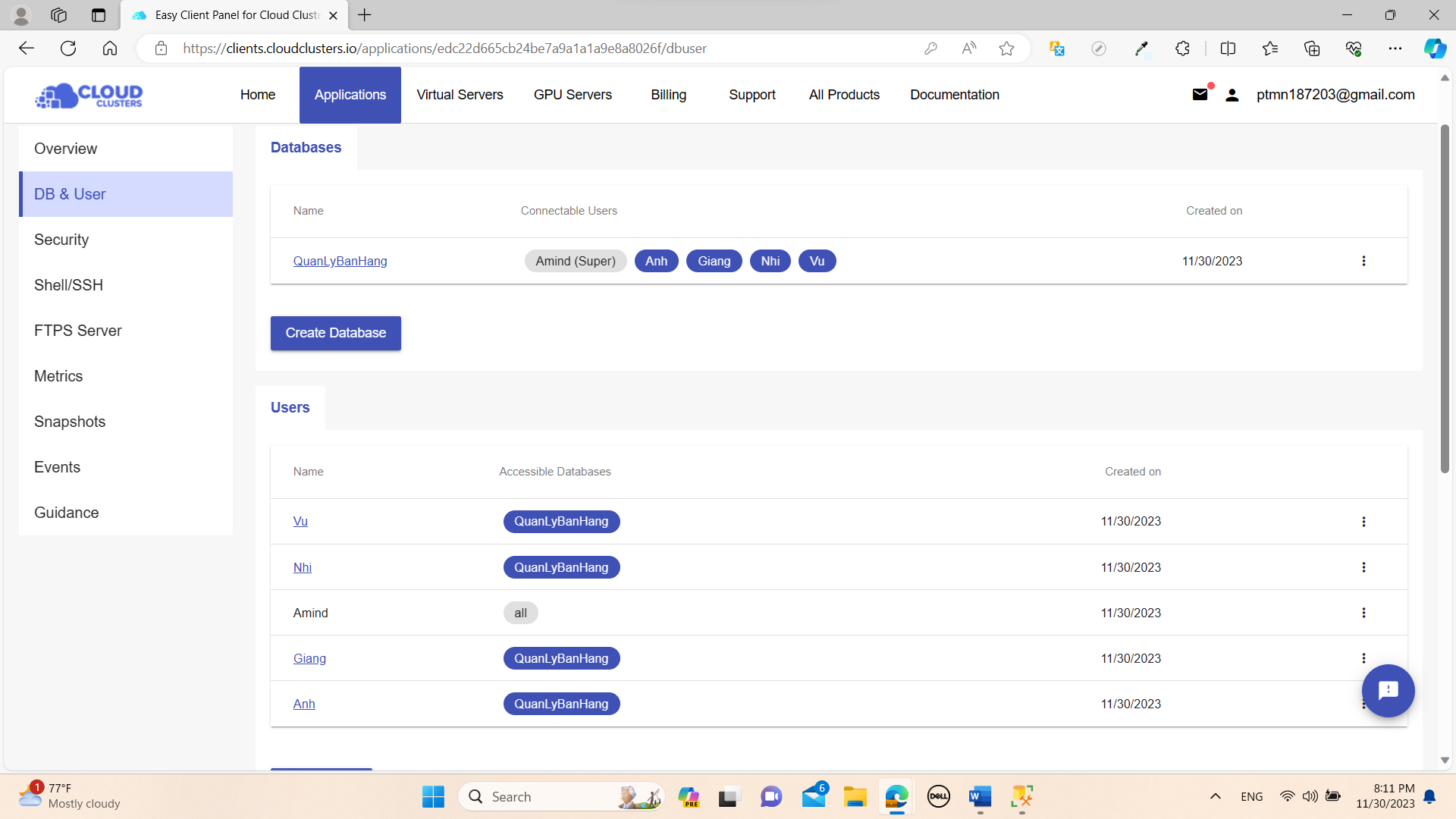
-- DataBase

[MinhNguyt/QTCSDL\_HeThongQuanLyBanHang\_DATA\_47K14\_10 (github.com)](https://github.com/MinhNguyt/QTCSDL_HeThongQuanLyBanHang_DATA_47K14_10)

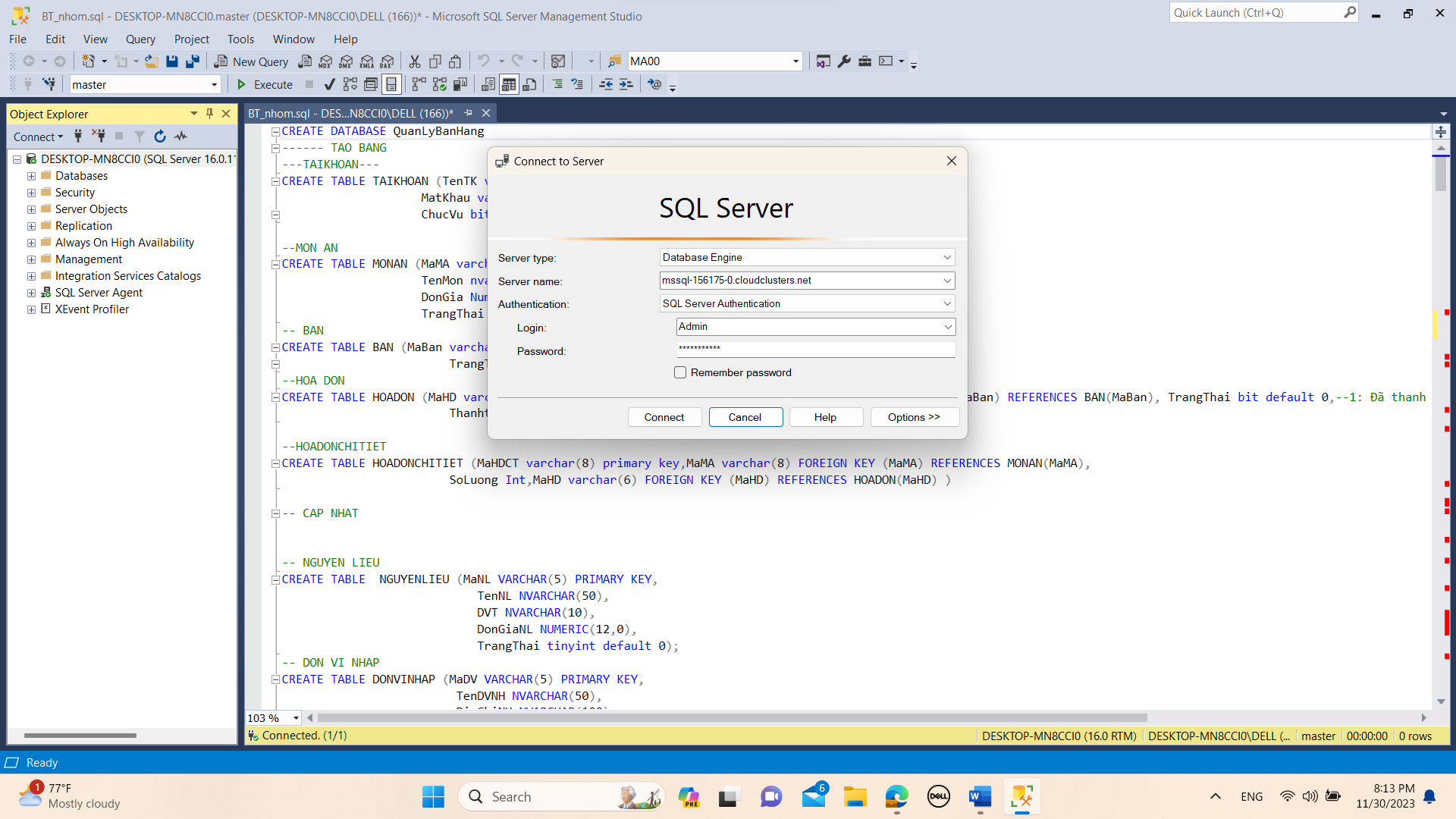
# 8. Phương án giải quyết tốt nhất cho việc lượng dữ liệu rất lớn, không đủ lưu trữ trong một ổ đĩa.

Phương pháp nhóm sử dụng ở đây là Lưu trữ đám mây:

* Lý do: Việc lưu trữ đám mây giúp tối ưu hóa quản lý tài nguyên, giảm chi phí, và cung cấp một môi trường linh hoạt và hiệu quả cho việc triển khai ứng dụng và dịch vụ trực tuyến. Ngoài ra, dịch vụ đám mây thường cung cấp cập nhật tự động và bảo dưỡng hạ tầng, giúp bạn duy trì ứng dụng và dịch vụ của mình với phiên bản mới nhất.
* Dịch vụ đám mây sử dụng: Cloud Clusters
* Cách thực hiện:
* Bước 1: Tạo Database và các User, cấp quyền cho tài khoản thành viên sẽ được Read & Write.



* Bước 2: lấy dữ liệu từ trên cloud clusters bằng cách connect trong SQL server với server name: mssql-156175-0.cloudclusters.net

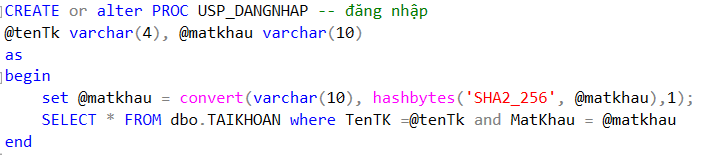


# **9. Phương án tốt nhất để đề phòng phương án hệ thống TESTDB bị tấn công bằng SQL Injection.**

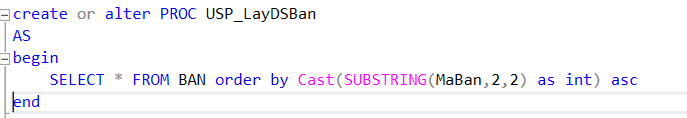
SQL Injection là một trong những mối quan tâm lớn về an ninh thông tin trong hệ thống quản lý cơ sở dữ liệu. Để đối phó với SQL Injection, nhóm chúng em đã chọn giải pháp

sử dụng Stored Procedures:

* Lý do:
* Thực Hiện Truy Vấn An Toàn:
  + Stored Procedures cho phép bạn đặt các thủ tục xử lý truy vấn trực tiếp trong cơ sở dữ liệu.
  + Các thủ tục này được định nghĩa trước và chỉ được thực thi thông qua gọi các thủ tục, giảm khả năng thực hiện truy vấn không an toàn trực tiếp từ ứng dụng.
* Ngăn Chặn SQL Injection:
  + Bằng cách sử dụng các thủ tục lưu trữ, bạn có thể tránh được việc người dùng có thể chèn các đoạn mã SQL độc hại vào câu truy vấn.
  + Các tham số trong stored procedures thường được xử lý một cách an toàn, không tạo cơ hội cho SQL Injection.
* Tăng Tính Tương Thích:
  + Sử dụng Stored Procedures tạo ra một lớp trung gian giữa ứng dụng và cơ sở dữ liệu. Điều này giúp giảm sự phụ thuộc trực tiếp vào câu truy vấn SQL từ phía ứng dụng.
  + Nếu có sự thay đổi trong cấu trúc cơ sở dữ liệu, chỉ cần điều chỉnh stored procedure mà không ảnh hưởng đến mã nguồn ứng dụng.
* Quản lý Quyền Truy Cập:
  + Stored Procedures có thể được xác định với quyền truy cập cụ thể, giúp kiểm soát quyền truy cập vào dữ liệu.
  + Người quản trị cơ sở dữ liệu có thể cấp quyền thực thi stored procedure mà không cần cấp quyền trực tiếp trên các bảng hoặc cơ sở dữ liệu.
* Hiệu Suất:
  + Stored Procedures thường được biên dịch và tối ưu hóa một lần khi được tạo ra, cung cấp hiệu suất tốt hơn so với các truy vấn SQL động.
  + Điều này giúp giảm tải cho cả ứng dụng và cơ sở dữ liệu
* Thực hiện:
* Thực hiện đăng nhập



* Thực hiện lấy danh sách bàn



Đây là một số ví dụ về bảo mật bằng sử dụng Stored Procedures.

# PHỤ LỤC

1. Thiết kế cơ sở dữ liệu

https://drive.google.com/file/d/18\_g7RtqvLmErrvXgl6wN0aMAA7w4vlAM/view?usp=sharing