**SQL SERVER:-**

* SQL server is an RDMS product
* **To** develop backend application to store the data.
* Found by MICROSOFT
* IN 1989
* BACKEND tool
* Backend app(oracle,mysql,etc..)

**FRONT END:-**

If user interacts directly with the app, then it is known as front end apps

TOOLS :- .net or java

**BACK END:-**

User can’t interact directly with an application, then it is known as backend apps

TOOLS:- SQL server , Oracle

**INSTALLATION OF SQL SERVER:-**

* We have to maintain windows OS only.
* SQL SERVER is pure platform dependent product
* PLATFORM :-

OS + PROCESSOR

Plat form independent :-

Combination of ANY OS + ANY PROCESSOR

Oracle(in back end)

Plat form dependent :-

Combination of ONE OS + ANY PROCESSOR

SQL Server

🡪 SQL provide more GUI (Graphical User Interface) services

CUI -- > With query (Time Consuming Process)

GUI -- > Without query (Efficiency Process)

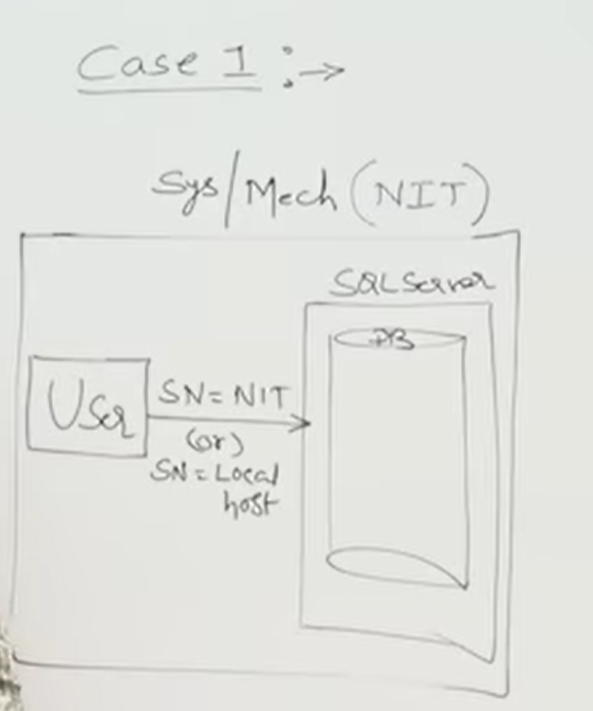
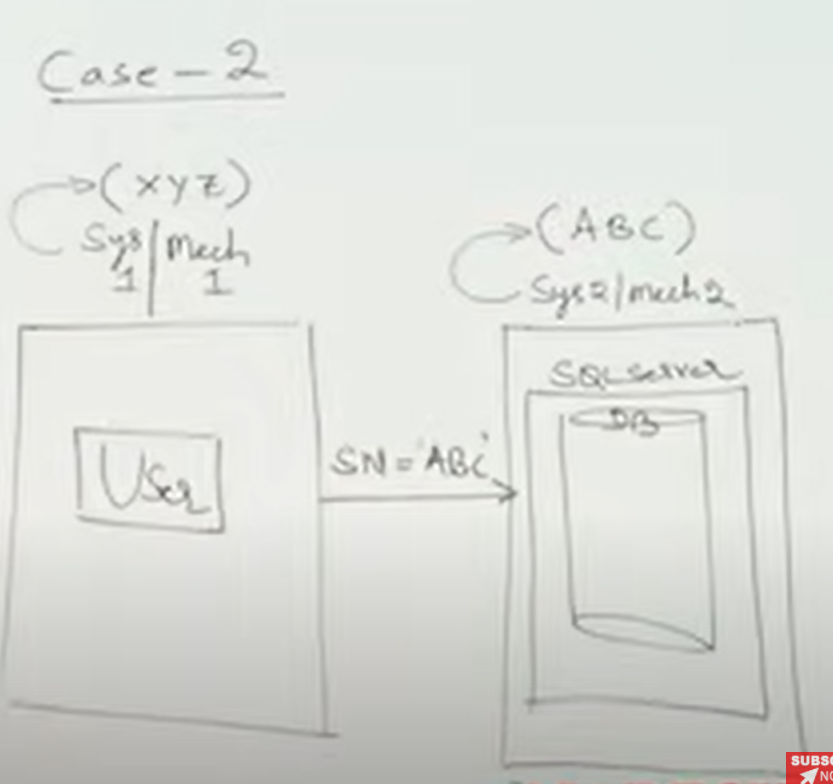
🡪 GUI provides DB Operations like Create, Insert , Delete, Update, Select

**WORKING WITH SQL Server :-**

SQL SERVER S/W ---- > Install ---- > Instance name(object)(Reference memory is allocated)

* Under a single instance memory we can create 32,767 Data Bases of SQL Server
* SQL Server is the collection of Data Bases.
* Data Bases is collection of different objects(eg- Tables,synonyms,functions,Triggers,….)

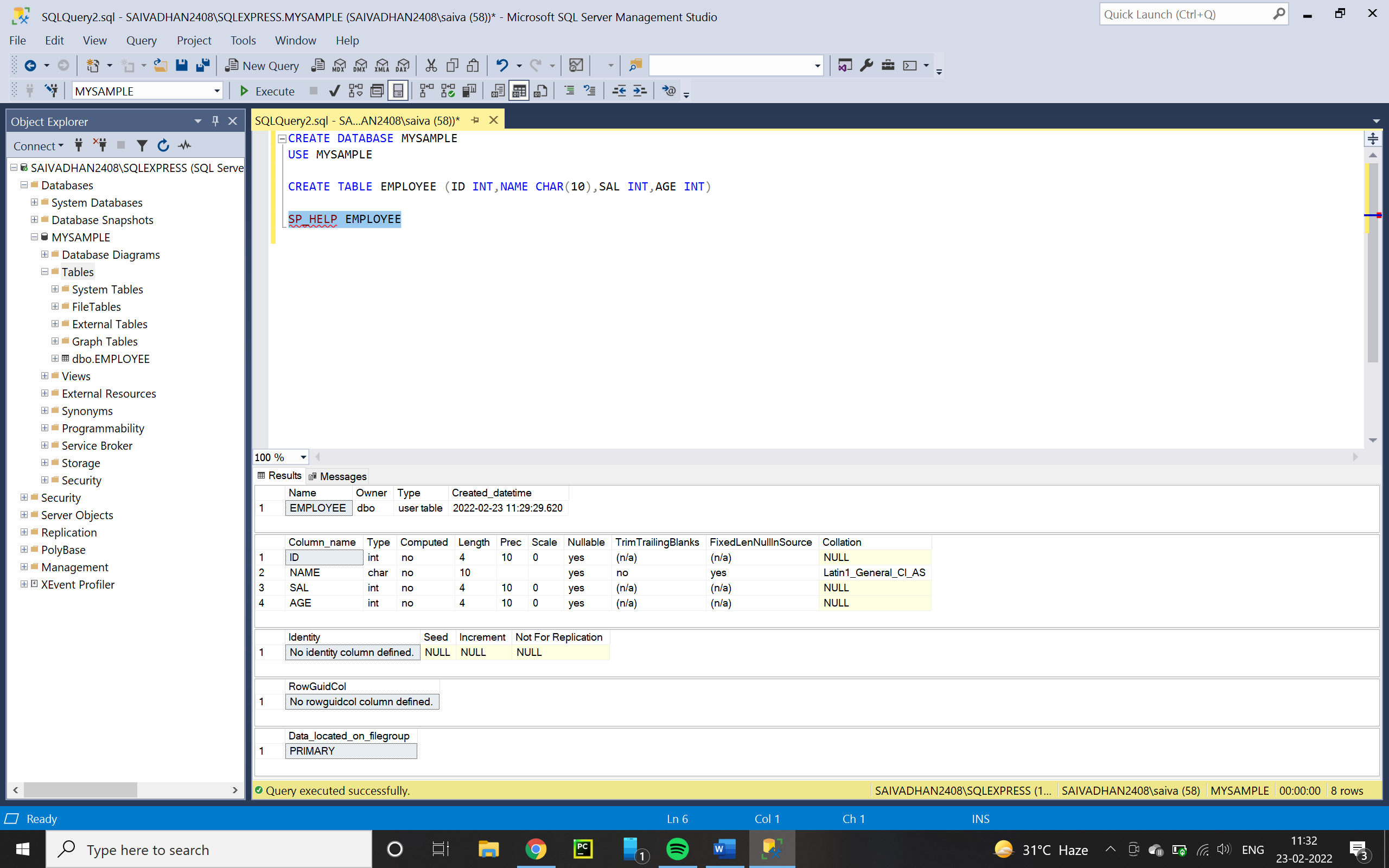
**SSMS TOOLS :- (**This will get after installation of SQL Server**)**

* SQL Server Management Studio
* One of the developing tool
* It can perform to types of operations
  + CUI
  + GUI
* These two are used to perform data base operations
  + Like creating a table, inserting, select, update etc, …
* **SERVER TYPES :-** 
  + Data Base Engine
  + Analysis Service (SSAS)
  + Reporting Services (SSRS)
  + Integration services (SSIS)
  + SSAS,SSRS,SSIS these are called as MSBI TOOLS
  + We need saperate tools BITS(Business Intellegance Developing Studio) to interact with MSBI TOOLS
  + For Data Base Engine there is a separate developing tool SSMS
  + **DATA BASE ENGINE : -**
    - CORE component of SQL Server
    - With out data base engine we cant expect DBMS
    - Three major reasons
      * Storing large amount of data [Tables (Rows and columns)]
      * Processing of data [ Insert, Update, Delete, Select ]
      * Security for the data [ to data ]
  + **ANALYSIS SERVICE : -**
    - MSBI TOOL
    - Used in Dataware-housing Environment(storing )
    - Data to be converted into graphical like (2D to 3D)
  + **REPORTING SERVICES : -**
    - SSRS TOOL
    - Document creation (Store the business related information)
    - Exporting the table information to any document (like ms word, Pdf etc)
  + **INTEGRATION SERVICES : -**
    - SSIS TOOL
    - Convert from one data base tables into another database understandable format
* **SERVER NAME : -**
  + In which system we had installed our software , that system name will be taken as the server name
  + Server name will be find in which system we installed SQL Server software that system name will be taken as a server name before connect to SQL Server Data Base.
  + CASE 1 :-
    - Local host will be only done when both the user and server should be in the same zone
    - 
  + CASE – 2 :-
    - User and Server are in different systems
    - 
* **AUTHENTICATION : -**
  + To verify user credentials
  + The data base admin will create all these credentials like user id and password
  + Authentication modes are two types : -
    - Windows authentication : -
      * Default authentication mode of SQL Server
      * We can access our sever with windows authentication only
      * We will play admin role (we have all rights)
    - SQL Server authentication : -
      * Mixed authentication mode of SQL Server
      * We can access our sever with windows authentication and SQL Server authentication .
      * We will play just user (We don’t have any rights)
      * Need permission from admin to do any operation
* **USER NAME : -**
  + Default “sa”
* **PASS WORD : -**
  + At installation time
* **Difference between Admin login and User login :-** 
  + **Admin login :-**
    - We can do any manipulations
    - We can permit to open every or any folder
    - No need to take any permission to open any folder
  + **User login : -**
    - We have to create our own login id and password
    - First of all we have to connect to SQL Server
    - **Synatax to create new login details : -**
      * CREATE LOGIN <LOGIN NAME> WITH PASSWORD=’<PASSWORD>’;
      * Login name and password will be created under administrator environment
    - Need to take permissions
    - The folder can not able to open unless we took permission from admin.
* **CREATION OF NEW DATA BASE (CUI): -**
  + **SYNTAX : -**
    - CREATE DATABASE <DB NAME>;
* **WORKING WITH SUB-LANGUAGES :-**
  + There are 5 types : -
    - **DDL(Data Definition Language)**
      * CREATE
      * ALTER
      * SP\_RENAME
      * TRUNCATE
      * DROP
    - **DML(Data Manipulation Language)**
      * INSERT
      * UPDATE
      * DELETE
    - **DQL(Data Query Language)**
      * SELECT
    - **TCL(Transactional Control Langugae)**
      * COMMIT
      * ROLLBACK
      * SAVEPOINT
    - **DCL(Data Control Language)**
      * GRANT
      * REVOKE
* **WORKING WITH DDL COMMANDS : -**
  + CREATE NEW OBJECTS(TABLE,VIEW,SYNONYM,PROCEDURE,FUNCTION,TRIGGER….)
  + MODIFY THE STRUCTURE OF THE OBJECTS.
  + DROP THE OBJECTS FROM DATA BASE.
    - **CREATE : -**
      * USED TO CREATE A NEW DATA BASE AND TABLE
      * **CREATION OF TABLE(SYNTAX) : -**
        + CREATE TABLE <TABLE NAME> (<COLUMN NAME1> <DATA TYPE>[SIZE](if character), <COLUMN NAME1> <DATA TYPE>);
      * **SELECTION TO STORE IN OUR DATA BASE**
        + TWO WAYS

USE <DATABASE NAME>

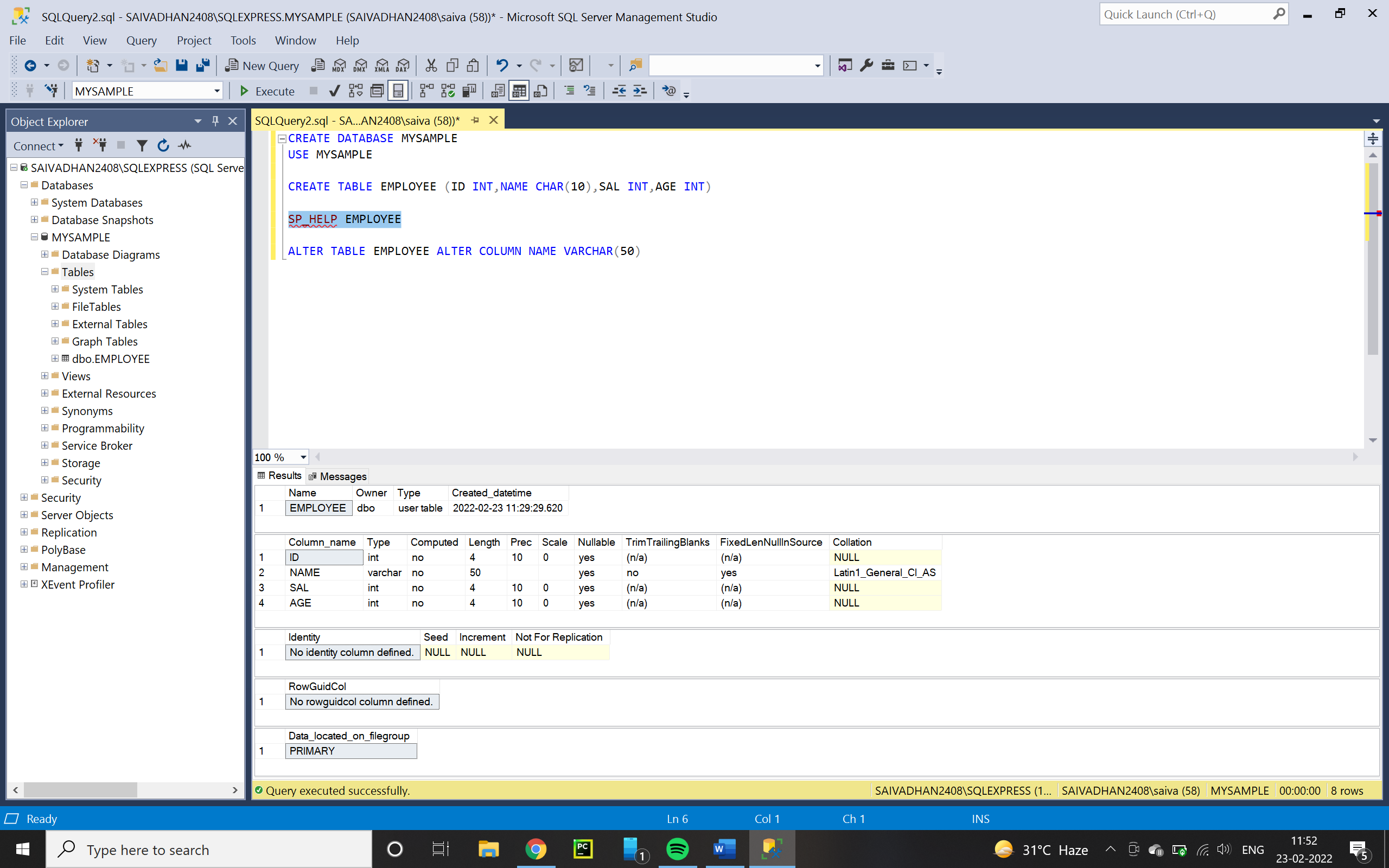
SELECTION AT DROP DOWN SEARCH

* + - * **VIEW THE STRUCTURE OF A TABLE (SYNTAX) : -**
        + SP\_HELP<TABLE NAME>



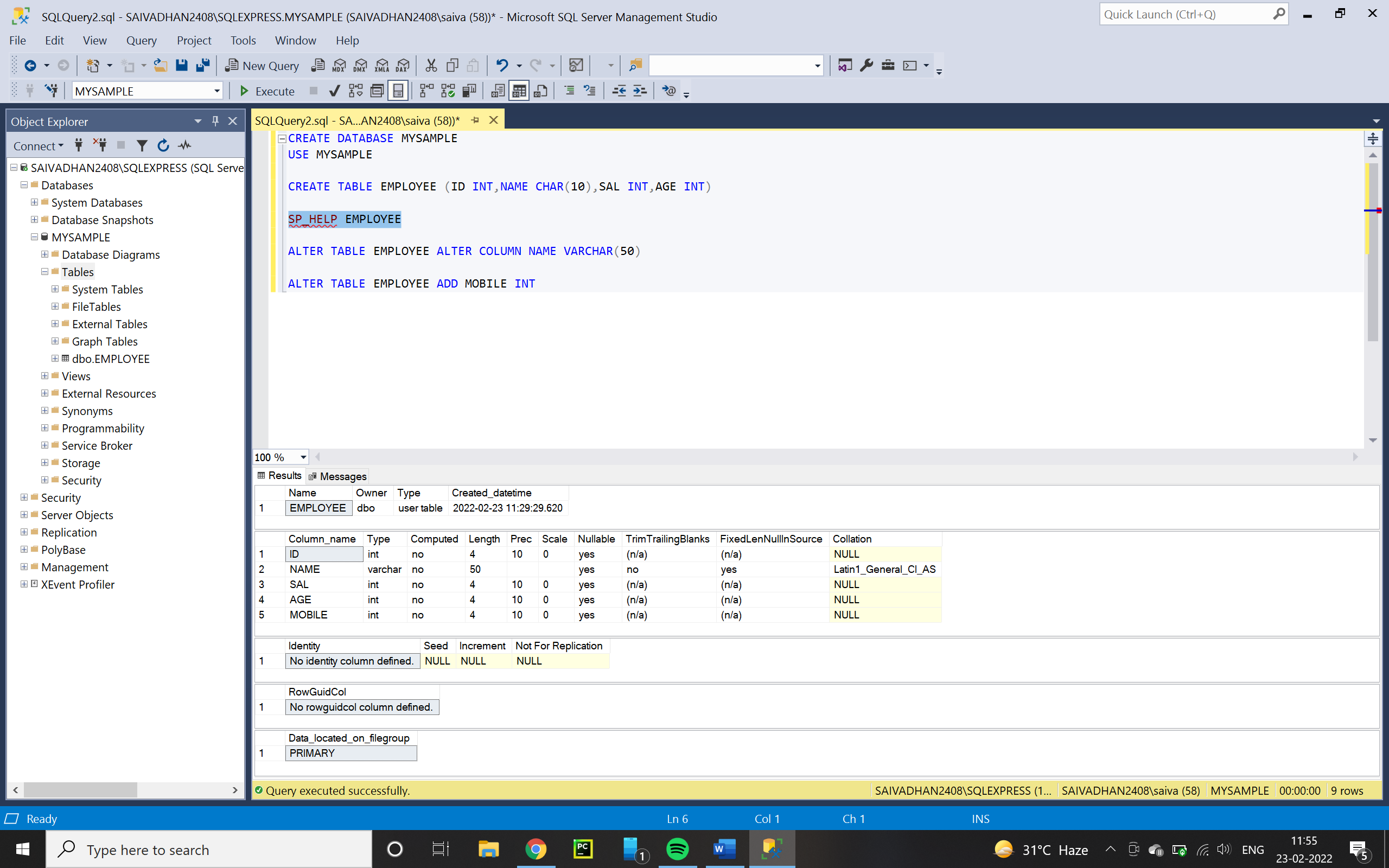
* + - **ALTER : -**
      * **SUB COMMANDS OF ALTER : -**
        + ALTER-ALTER COLUMN
        + ALTER-ADD
        + SP\_RENAME
        + ALTER-DROP
      * HOW TO CHANGE DATA TYPE AND ALSO SIZE OF THE DATA TYPE
        + SYNTAX : -

ALTER TABLE <TABLE NAME> ALTER COLUMN <COLUMN NAME> <NEW DATA TYPE>[NEW SIZE];



* + - * HOW TO ADD A NEW COLUMN
        + SYNTAX :-

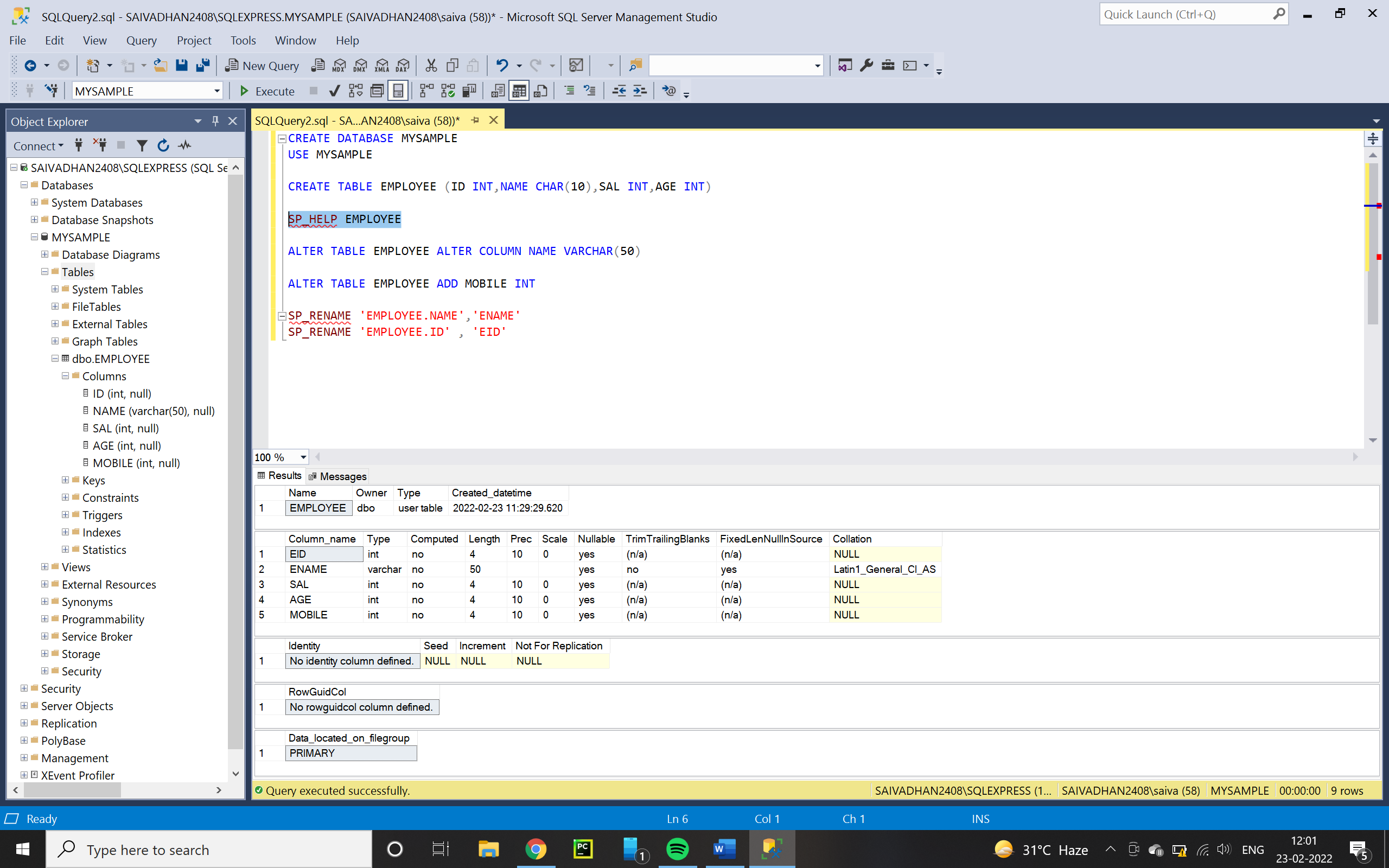
ALTER TABLE <TABLE NAME> ADD <NEW COLUMN NAME><DATA TYPE>[SIZE]



* + - * HOW TO CHANGE A COLUMN NAME
        + SP\_RENAME(stored procedure)
        + SYNTAX : -

FOR CHANGING COLUMN NAME

SP\_RENAME ‘<TABLE NAME>.<OLD COLUMN NAME>’ , ‘<NEW COLUMN NAME>’

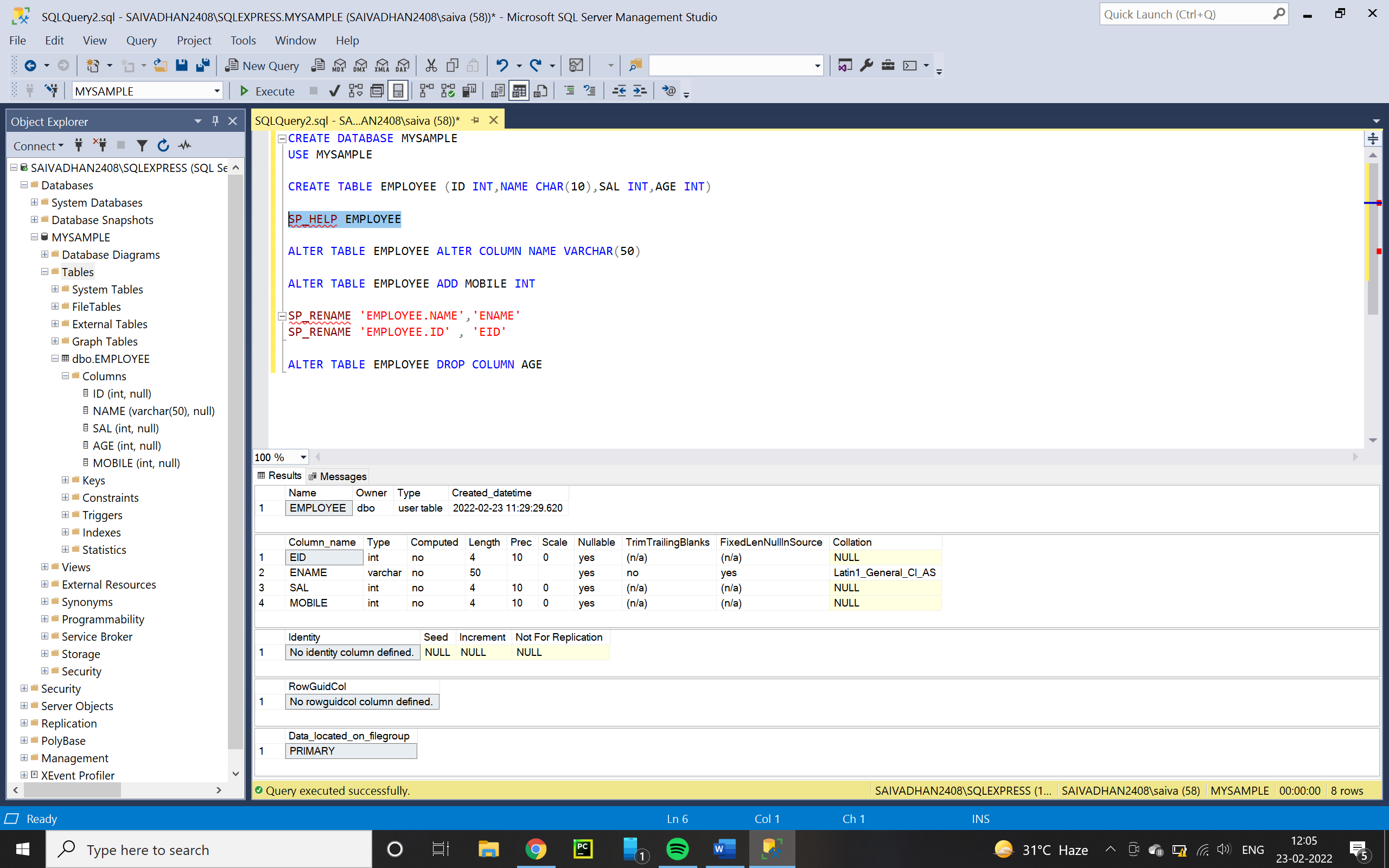


FOR CHANGING TABLE NAME

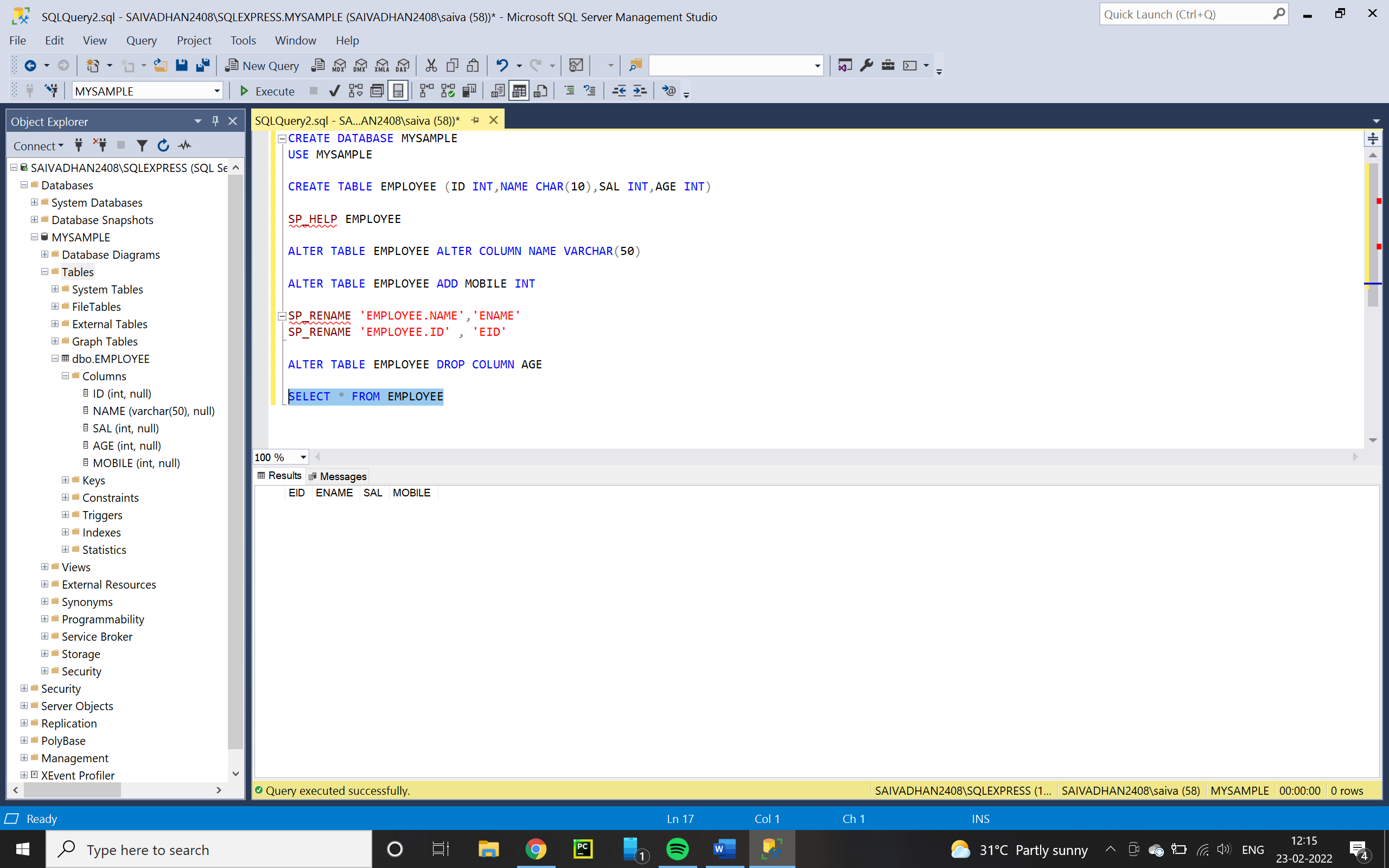
SP\_RENAME ‘<OLD TABLE NAME>’ , ‘<NEW TABLE NAME>’

* + - * HOW TO DROP THE UN WANTED COLUMNS
        + SYNTAX : -

ALTER TABLE <TN> DROP COLUMN <COLUMN NAME>



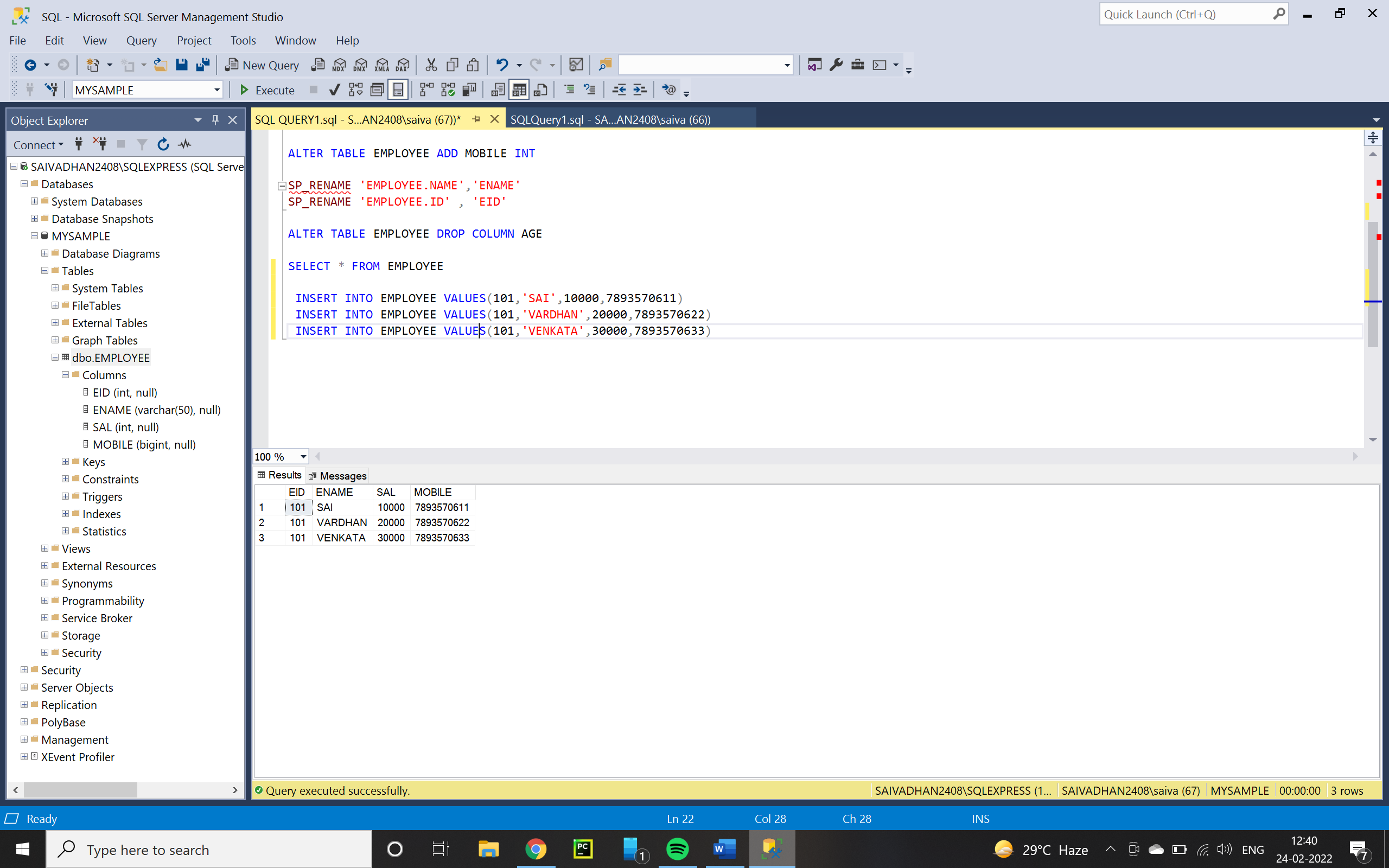
* + - **TRUNCATE :-**
      * TO DELETE ALL ROWS FROM THE TABLE AT A TIME BUT NOT THE STRUCTURE.
      * WE CAN NOT DELETE A SPECIFIC ROW FROM THE TABLE
      * IT NOT SUPPORTS “**WHERE**” KEY WORD CONDITION
      * SYNTAX : -
        + TRUNCATE TABLE <TN>
    - **DROP : -**
      * TO DELETE WHOLE TABLE
      * SYNTAX : -
        + DROP TABLE <TABLE NAME>
* **WORKING WITH DQL COMMANDS : -**
  + SELECTION AND VIW OF ALL THE DETAILS OF THE TABLE
  + WE WILL PERFORM 3 OPERATIONS
    - READ DATA
    - RETRIVE DATA
    - DISPLAY DATA
    - SYNTAX : -
      * SELECT \* /<list of column names> from <TN> [WHERE(CONDITION)]
    - RETRIVE METHODS : -
      * **PROJECTION** :-
        + WITH OUT CONDITION
        + EX : -
        + SELECT \* FROM STUDENT
      * **SELECTION** :-
        + WITH CONDITION
        + EX : -
        + SELECT \* FROM STUDENT WHERE SNAME = ‘AABBCC’
      * **JOINS** : -
        + WE CAN RETRIVE DATA FROM MORE THAN ONE TABLE OR MULTIPLE TABLE
    - **SELECT : -**
      * SYNTAX : -
        + SELECT \* FROM <TABLE NAME>



* **WORKING WITH DML COMMANDS : -**
  + USED FOR MANIPULATION
  + **INSERT : -**
    - INSERT NEW VALUES INTO TABLE
    - TWO METHODS :-
      * METHOD 1 :-
      * INSERT VALUES FOR ALL COLUMNS INTO TABLE
        + SYNTAX :-

INSERT [INTO] < TN > VALUES (VALUE1,VALU2,….)

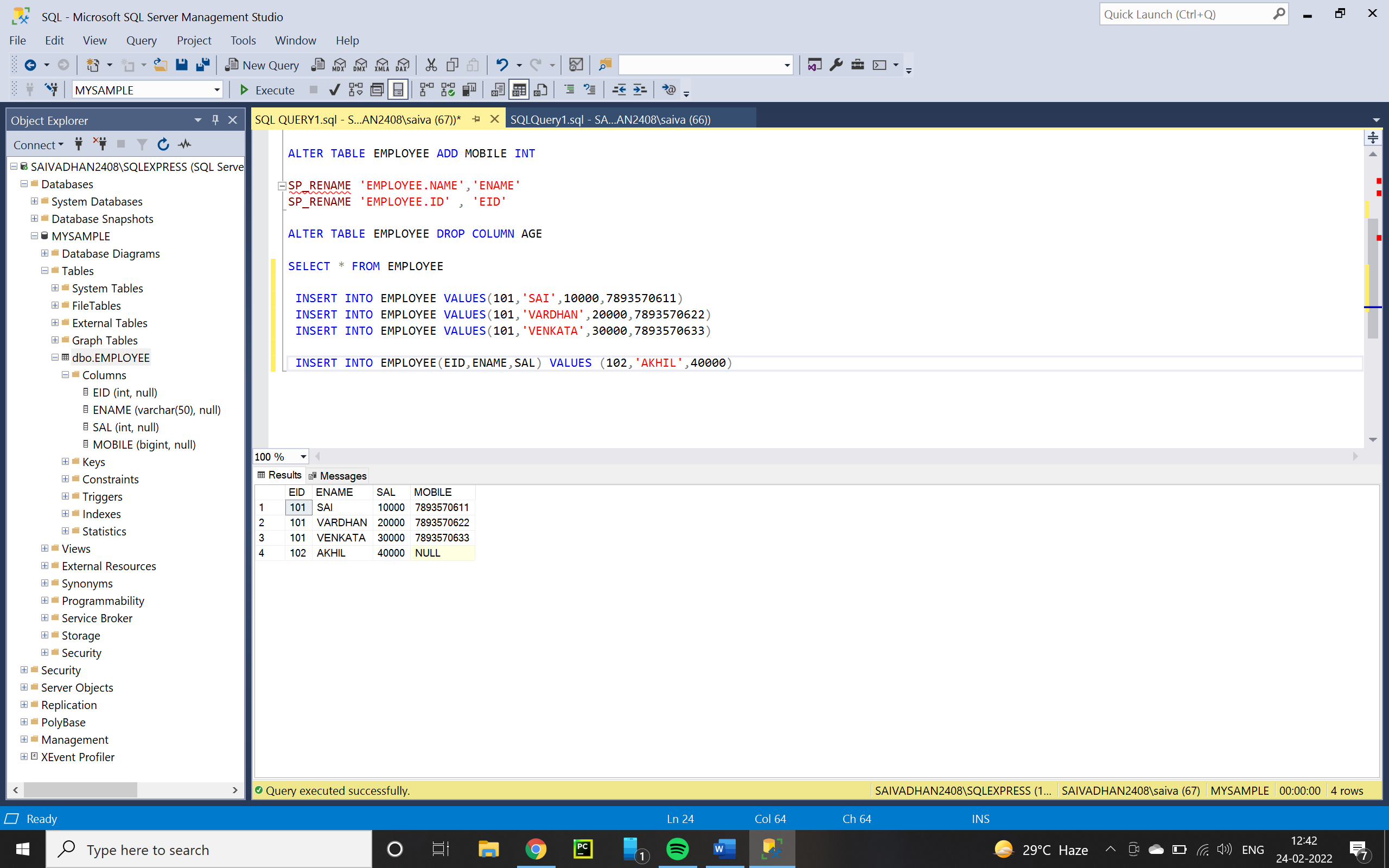
EACH ND EVERY COLUMN SHOULD BE INSERTED IN THIS METHOD



* + - * METHOD 2 :-
      * INSER VALUES FOR THE SELECTED COLUMNS
        + SYNTAX :-

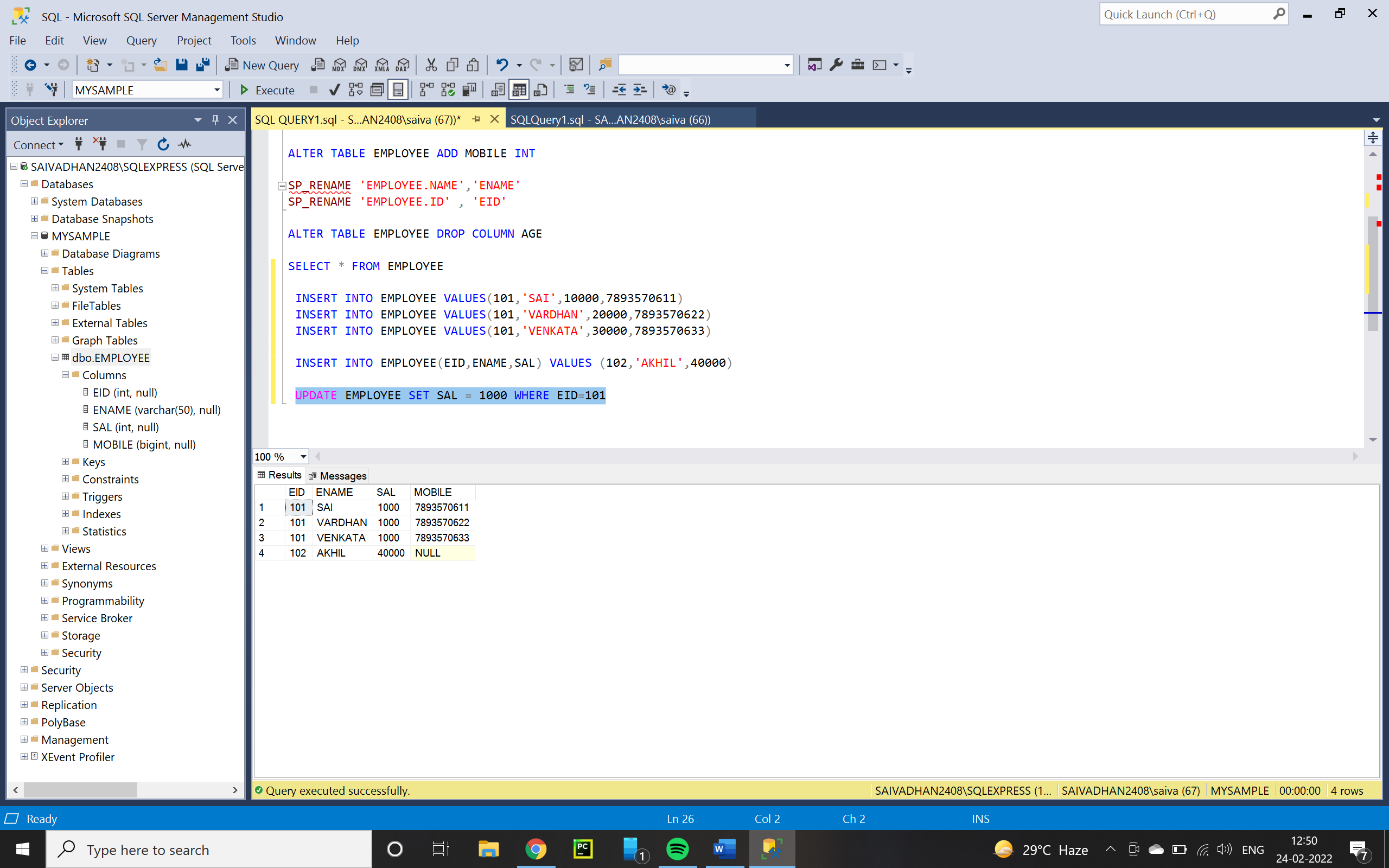
INSERT INTO <TN> (req.column names) VALUES(VALUE1,VALUE2,…..)

WE CAN INSERT VALUE AT OUR SPECIFIED FIELD



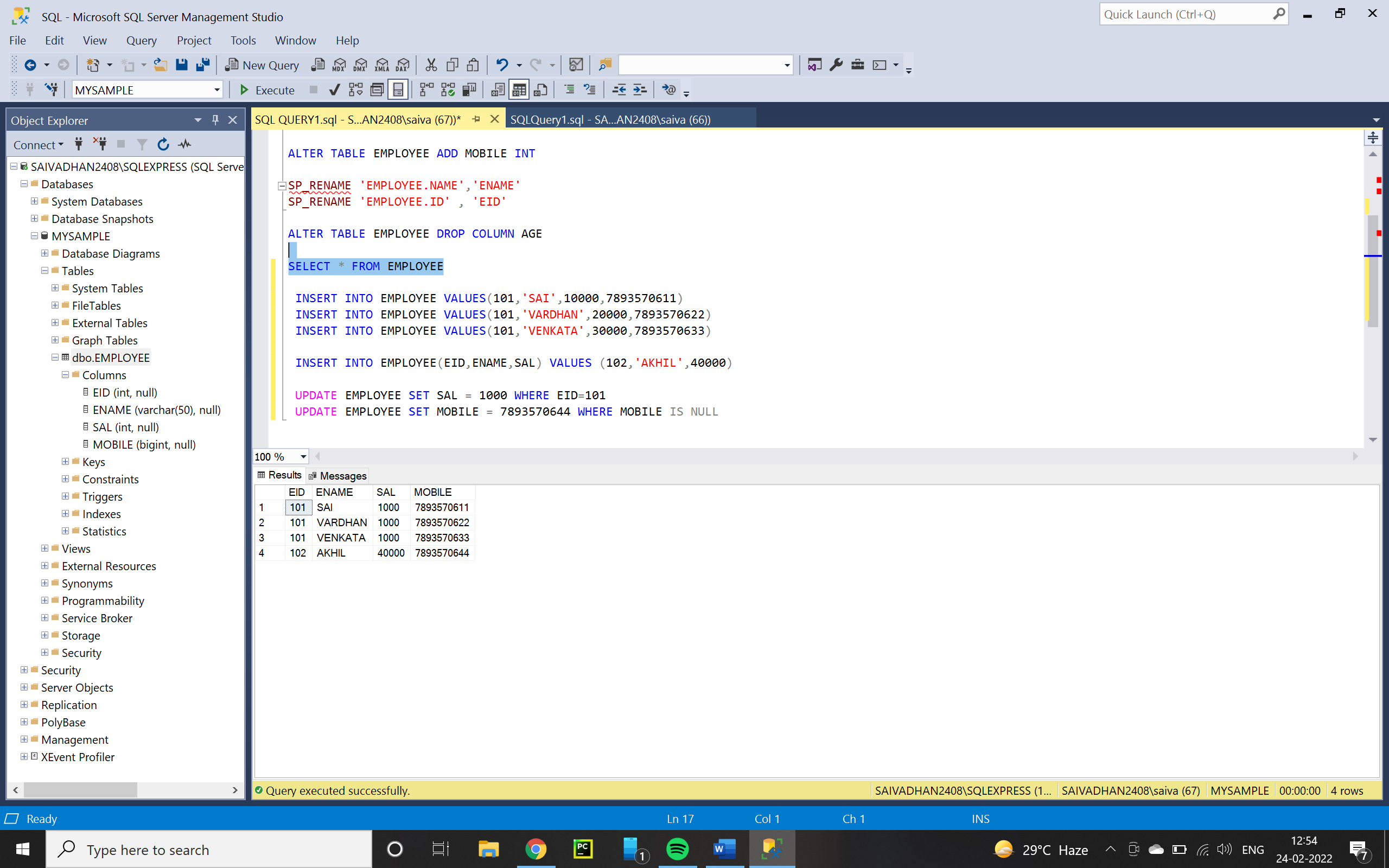
* + **UPDATE : -**
    - WE CAN UPDATE WHOLE ROWS OR WE CAN UPDATE A SINGLW ROW
    - WHENEVER WE WANT TO UPDATE PARTICULAR COLUMN WE HAVE TO USE “**WHERE”**  COMMAND
      * TO UPDATE ALL ROWS AND SPECIFIC ROW DATA IN A TABLE
        + SYNTAX : -

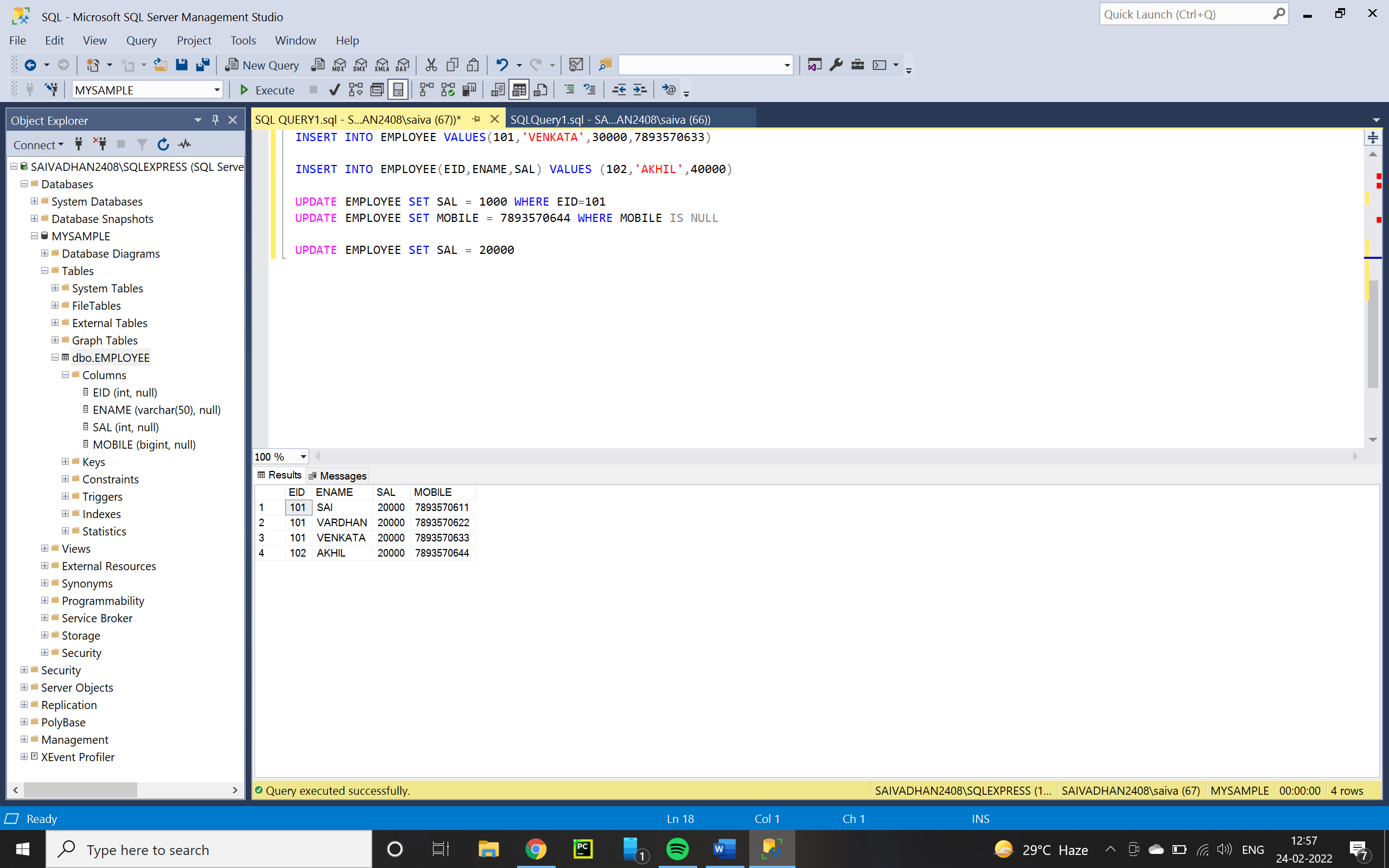
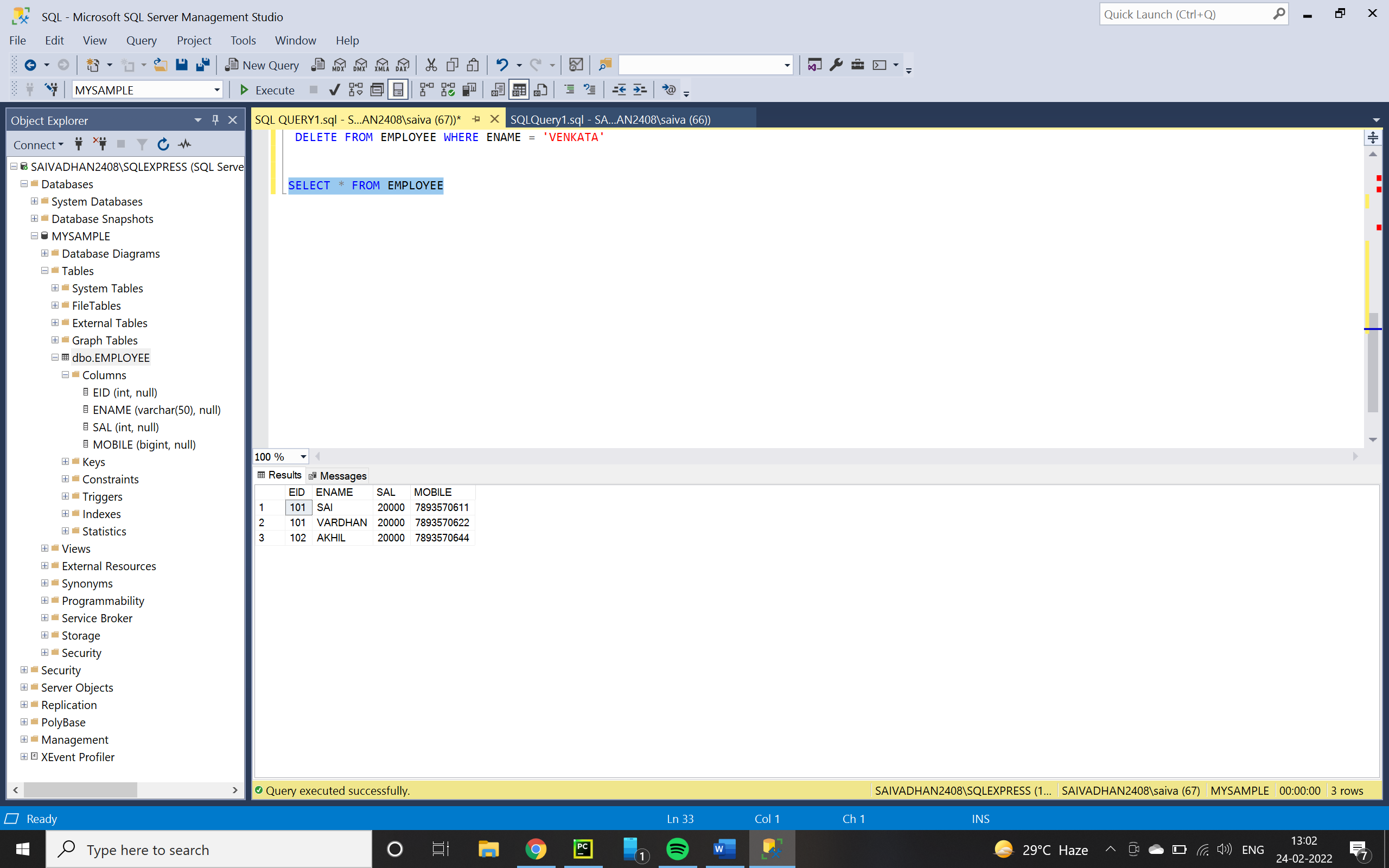
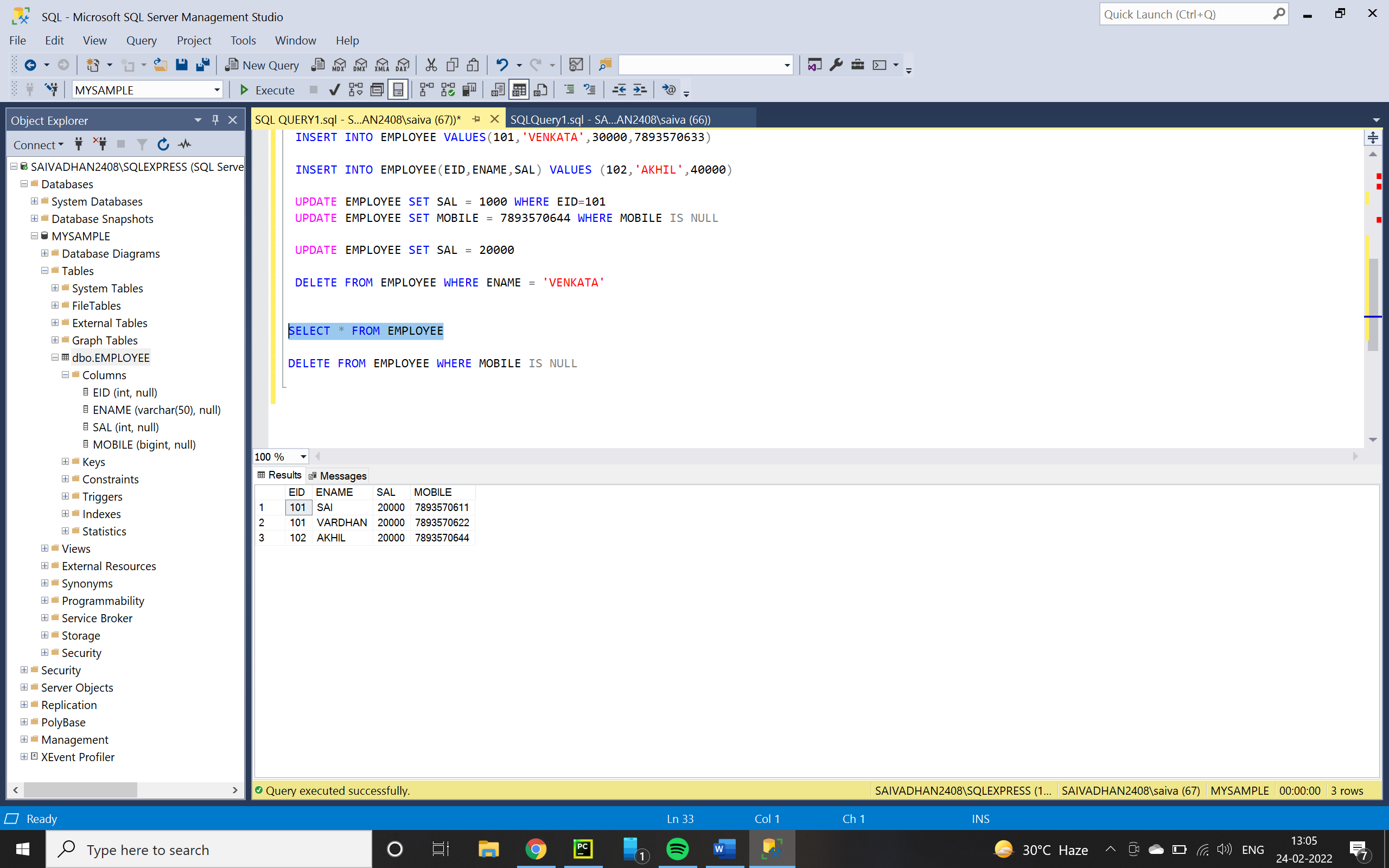
UPDATE <T N > SET <COLUMN NAME 1> = VALUE1,<COLUMN NAME 2> = VALUE2,…………[WHERE(CONDITION)];

EX 1 = UPDATE EMPLOYEE SET SAL = 1000 WHERE EID=101

EX2 = FILLING UNDEFINED VALUES

UPDATE EMPLOYEE SET MOBILE = 7893570644 WHERE MOBILE IS NULL



* + - * EX 3 : - UPDATION OF ALL RECORDS AT A TIME
      * UPDATE EMPLOYEE SET SAL = 20000
      * 
  + **DELETE : -**
  + WE CAN DELETE ALL ROWS OR SPECIFIC ROW FROM A TABLE
  + SYNTAX : -
    - DELETE FROM <TN> [WHERE(CONDITION)]
    - EX1 : - DELETE SPECIFIC DETAILS
    - DELETE FROM EMPLOYEE WHERE ENAME = 'VENKATA'
    - 
    - EX – 2 : - DELETING THE NULL ROW : -
    - DELETE FROM EMPLOYEE WHERE MOBILE IS NULL
    - 
    - EX 3 : - DELETING ALL RECORDS
    - DELETE FROM EMPLOYEE
    - 