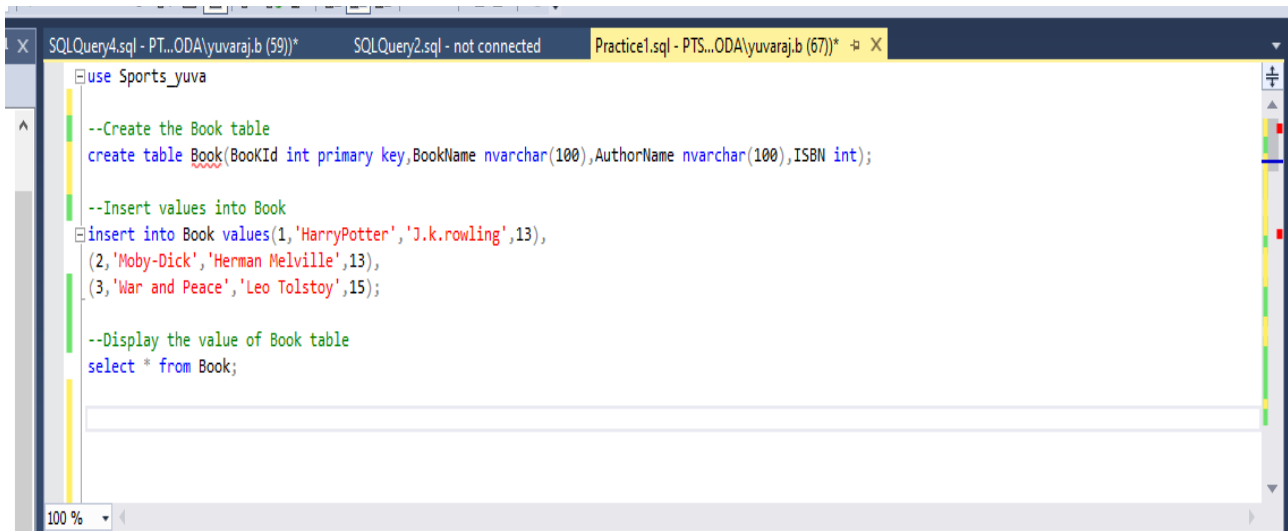


Date : 24/07/2024

1. Create Book as table with columns BookID, BookName, AuthorName, ISBN



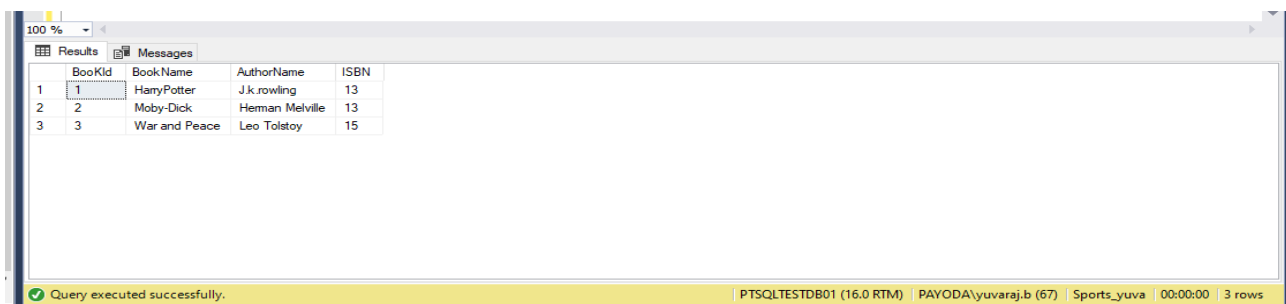
```
use Sports_yuva

--Create the Book table
create table Book(BookID int primary key, BookName nvarchar(100), AuthorName nvarchar(100), ISBN int);

--Insert values into Book
insert into Book values(1, 'HarryPotter', 'J.k.rowling', 13),
(2, 'Moby-Dick', 'Herman Melville', 13),
(3, 'War and Peace', 'Leo Tolstoy', 15);

--Display the value of Book table
select * from Book;
```

Output:



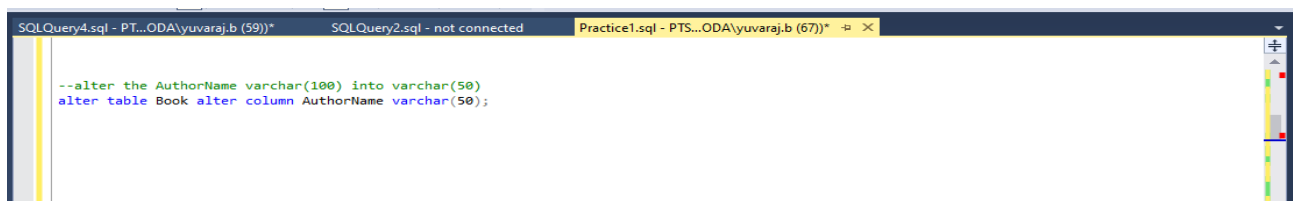
	BookID	BookName	AuthorName	ISBN
1	1	HarryPotter	J.k.rowling	13
2	2	Moby-Dick	Herman Melville	13
3	3	War and Peace	Leo Tolstoy	15

Query executed successfully. | PTSQLESTDB01 (16.0 RTM) | PAYODA\yuvaraj.b (67) | Sports_yuva | 00:00:00 | 3 rows

2. BookID should be the primary key

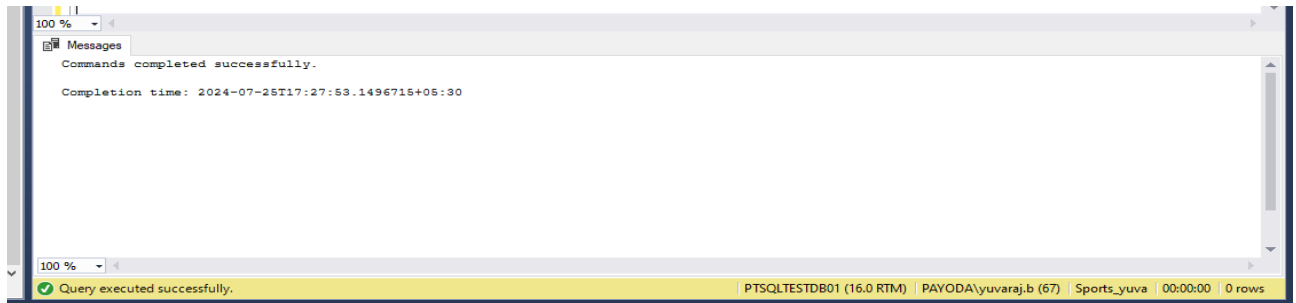
```
--Create the Book table
create table Book(BookID int primary key, BookName nvarchar(100), AuthorName nvarchar(100), ISBN int);
```

3. Alter Type from NVARCHAR(100) to NVARCHAR(50)

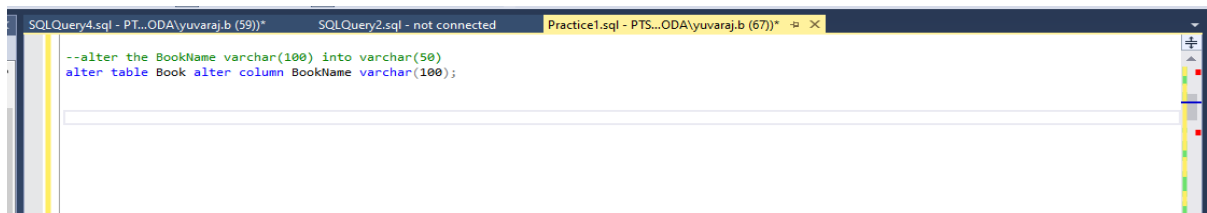


```
--alter the AuthorName varchar(100) into varchar(50)
alter table Book alter column AuthorName varchar(50);
```

output:



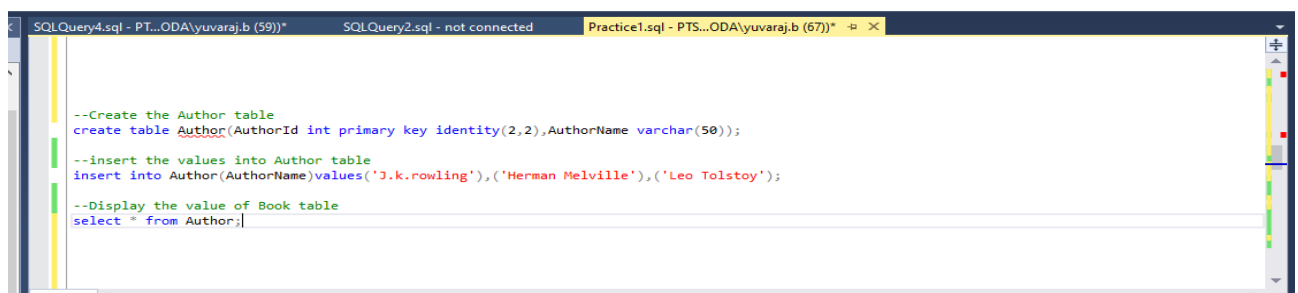
4.Alter Type from NVARCHAR(100) to NVARCHAR(150)



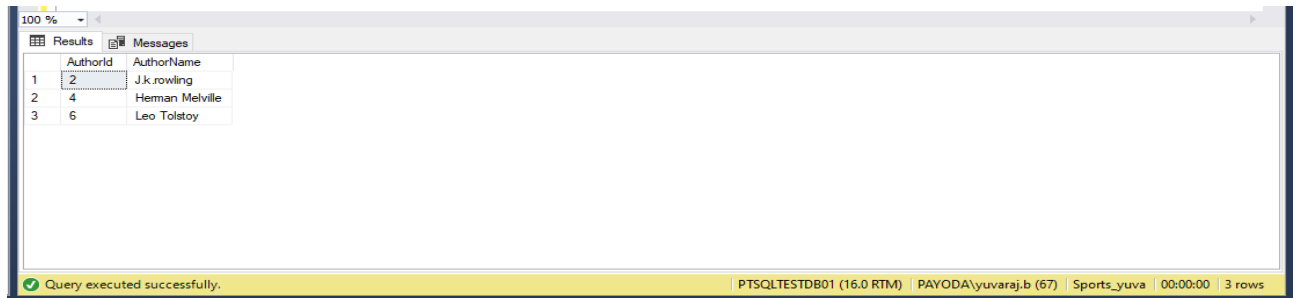
Output:



6. Create Author table with author_id,authorName



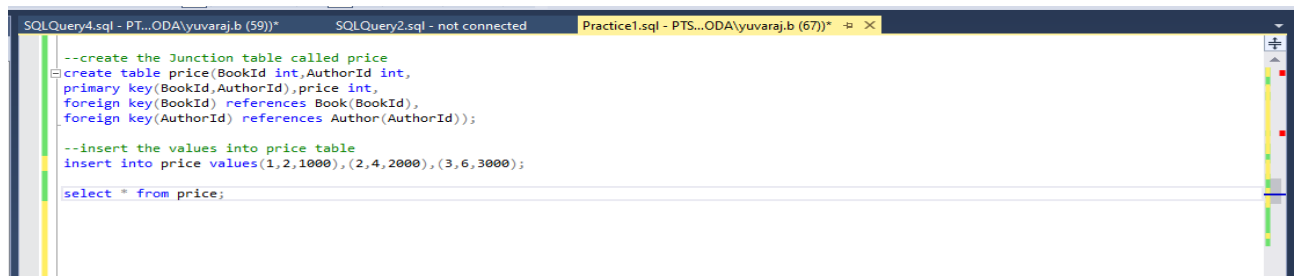
Output:



The screenshot shows a SQL Server Enterprise Manager window with a query results grid. The grid has two columns: 'AuthorId' and 'AuthorName'. There are three rows of data. The status bar at the bottom indicates 'Query executed successfully.' and '3 rows'.

	AuthorId	AuthorName
1	2	J.k.rowling
2	4	Herman Melville
3	6	Leo Tolstoy

7.Create Junction table for Books and Authors



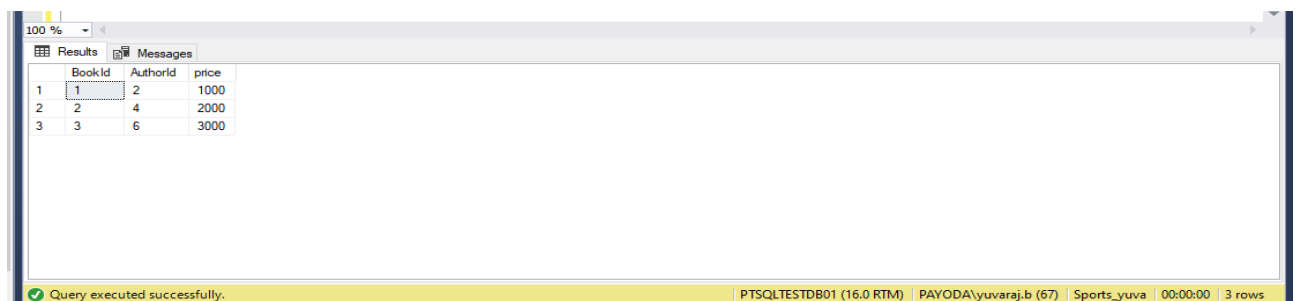
The screenshot shows a SQL Server Enterprise Manager window with a query editor. The code defines a junction table named 'price' with columns 'BookId', 'AuthorId', and 'price'. It includes primary and foreign key constraints. The code also inserts three rows of data into the 'price' table and selects all data from it.

```
--create the Junction table called price
create table price(BookId int,AuthorId int,
primary key(BookId,AuthorId),price int,
foreign key(BookId) references Book(BookId),
foreign key(AuthorId) references Author(AuthorId));

--insert the values into price table
insert into price values(1,2,1000),(2,4,2000),(3,6,3000);

select * from price;
```

Output:



The screenshot shows a SQL Server Enterprise Manager window with a query results grid. The grid has four columns: 'BookId', 'AuthorId', and 'price'. There are three rows of data. The status bar at the bottom indicates 'Query executed successfully.' and '3 rows'.

	BookId	AuthorId	price
1	1	2	1000
2	2	4	2000
3	3	6	3000