2. Using the CREATE DATABASE statement, create a new database named **test\_db** with explicit specifications for database and transaction log files. The database file with the logical name **test\_db\_dat** is stored in the file C:\tmp\test\_db.mdf and the initial size is 5MB, the maximum size is unlimited, and the file growth is eight percent. The log file called **test\_db\_log** is stored in the file C:\tmp\test\_db\_log.ldf and the initial size is 2MB, the maximum size is 10MB, and the file growth is 500KB.

USE master;

CREATE DATABASE test\_db

ON (NAME=test\_db\_dat,

FILENAME = 'C:\Program Files\Microsoft SQL Server\MSSQL12.SQLEXPRESS\MSSQL\DATA\test\_db1.mdf', SIZE = 5,

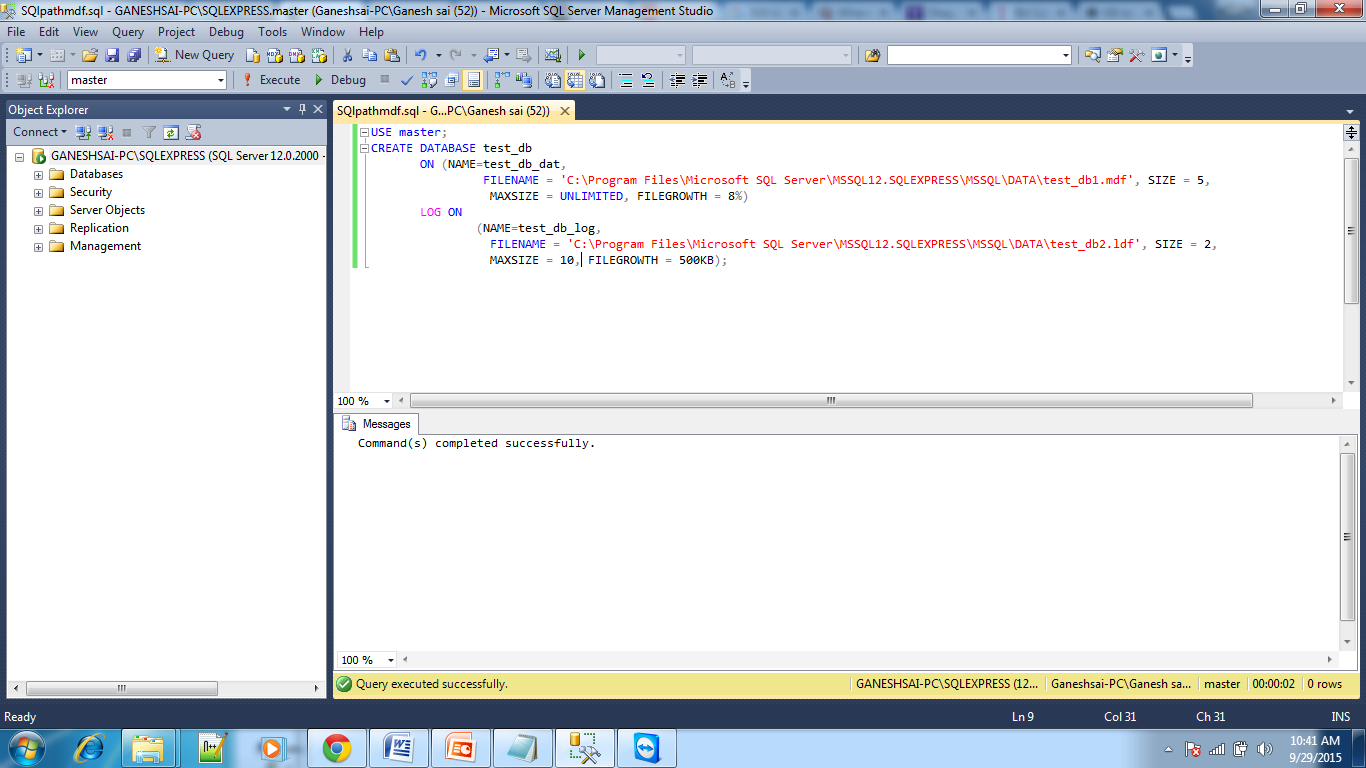
MAXSIZE = UNLIMITED, FILEGROWTH = 8%)

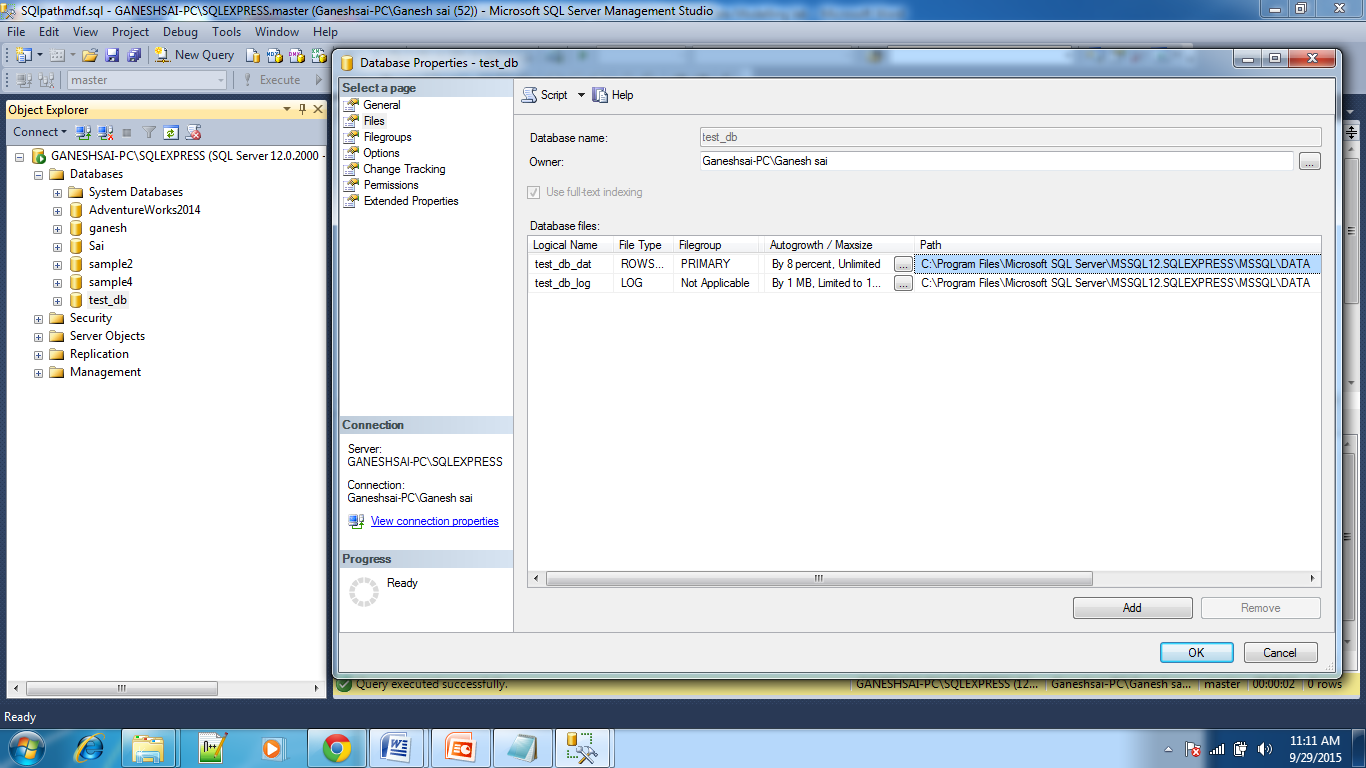
LOG ON

(NAME=test\_db\_log,

FILENAME = 'C:\Program Files\Microsoft SQL Server\MSSQL12.SQLEXPRESS\MSSQL\DATA\test\_db2.ldf', SIZE = 2,

MAXSIZE = 10, FILEGROWTH = 500KB);



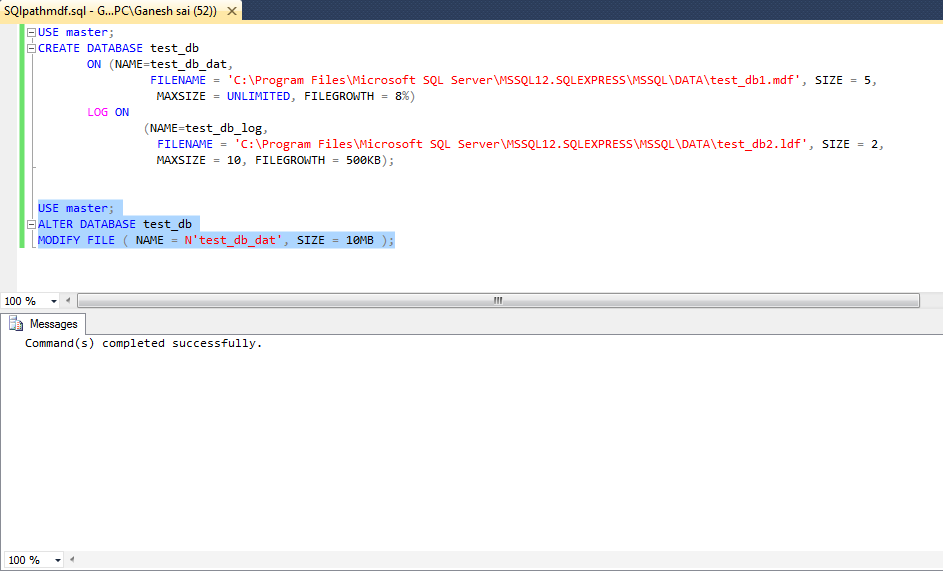


1. Using the ALTER DATABASE statement, change the file size of the **test\_db** database to 10MB.

USE master;

ALTER DATABASE test\_db

MODIFY FILE ( NAME = N'test\_db\_dat', SIZE = 10MB );



1. Create the tables **customers** and **orders** with the following columns (Use CREATE TABLE)

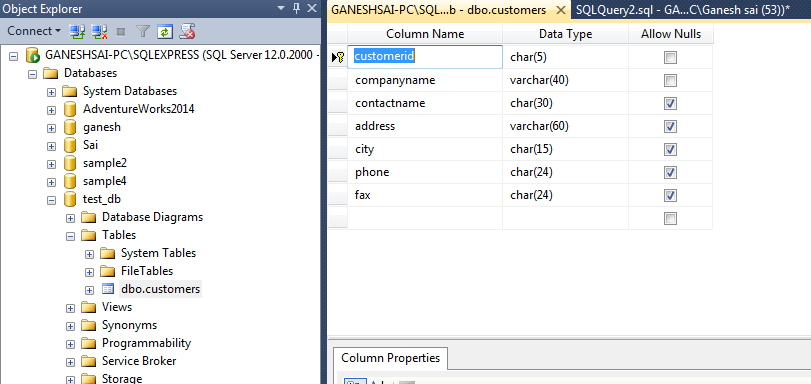
|  |  |
| --- | --- |
| **customers** | **orders** |
| customerid char(5) not null | orderid integer not null |
| companyname varchar(40) not null | customerid char(5) not null |
| contactname char(30) null | orderdate date null |
| address varchar(60) null | shippeddate date null |
| city char(15) null | freight money null |
| phone char(24) null | shipname varchar(40) null |
| fax char(24) null | shipaddress varchar(60) null |
|  | quantity integer null |

The primary keys are **customerid** for customers and **orderid** for orders

ANS:

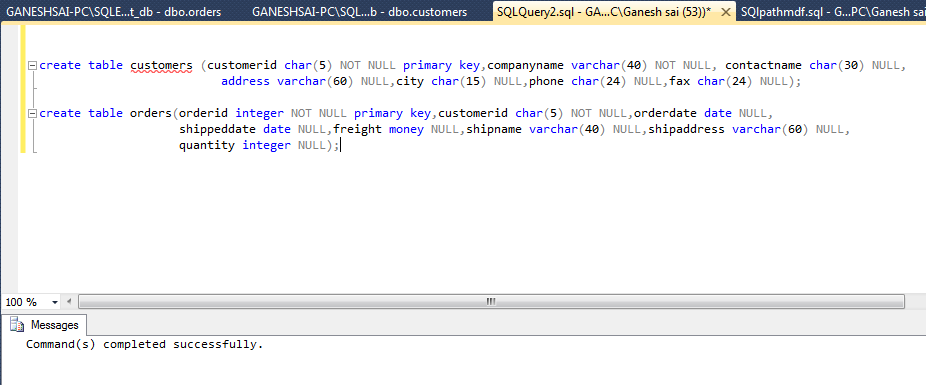
FIRST TABLE:

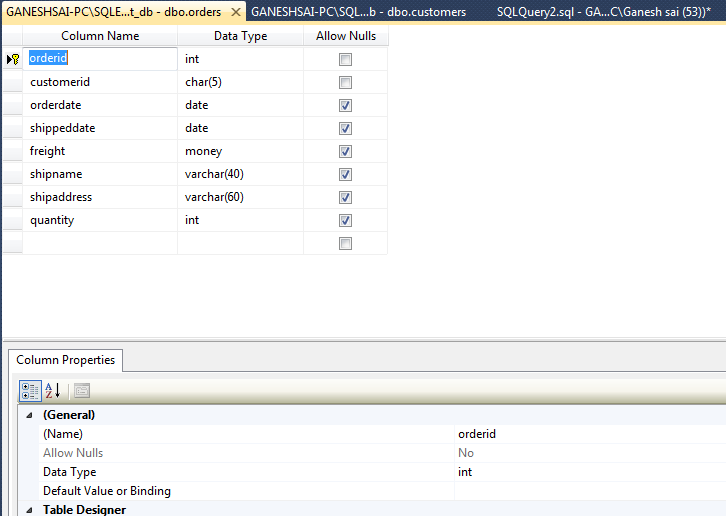
create table customers (customerid char(5) NOT NULL primary key,companyname varchar(40) NOT NULL, contactname char(30) NULL,address varchar(60) NULL,city char(15) NULL,phone char(24) NULL,fax char(24) NULL);



SECOND TABLE:

create table orders(orderid integer NOT NULL primary key,customerid char(5) NOT NULL,orderdate date NULL,shippeddate date NULL,freight money NULL,shipname varchar(40) NULL,shipaddress varchar(60) NULL,quantity integer NULL);



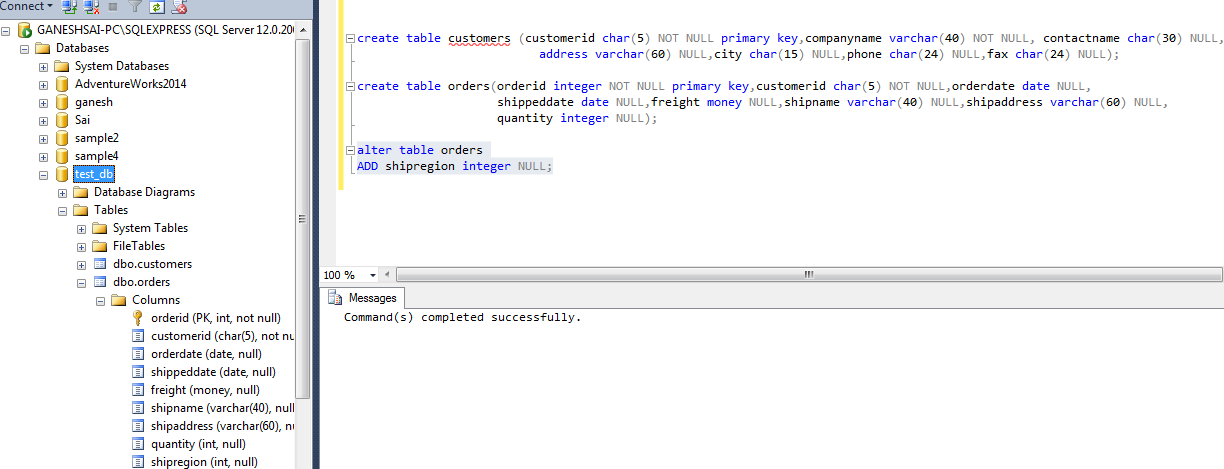


1. Using the ALTER TABLE statement, add a new column named **shipregion** to the **orders** table. The fields should be nullable and contain integers.

ANS:

alter table orders

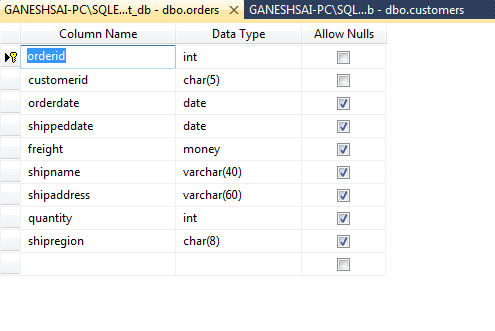
ADD shipregion integer NULL;



1. Using the ALTER TABLE statement, change the data type of the column **shipregion** from INTEGER to CHARACTER with length 8. The fields may contain NULL values.

ANS:

ALTER TABLE orders ALTER COLUMN shipregion character(8) ;

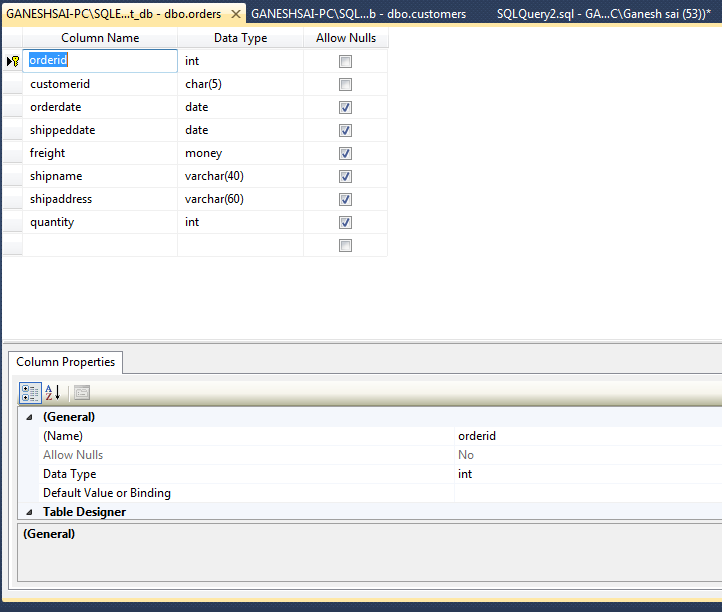


1. Delete the formerly created column **shipregion**.

ANS:

ALTER TABLE orders

DROP COLUMN shipregion;



1. Using the ALTER TABLE statement, create an integrity constraint that limits the possible values of the **quantity** column in the **orders** table to values between 1 and 30.

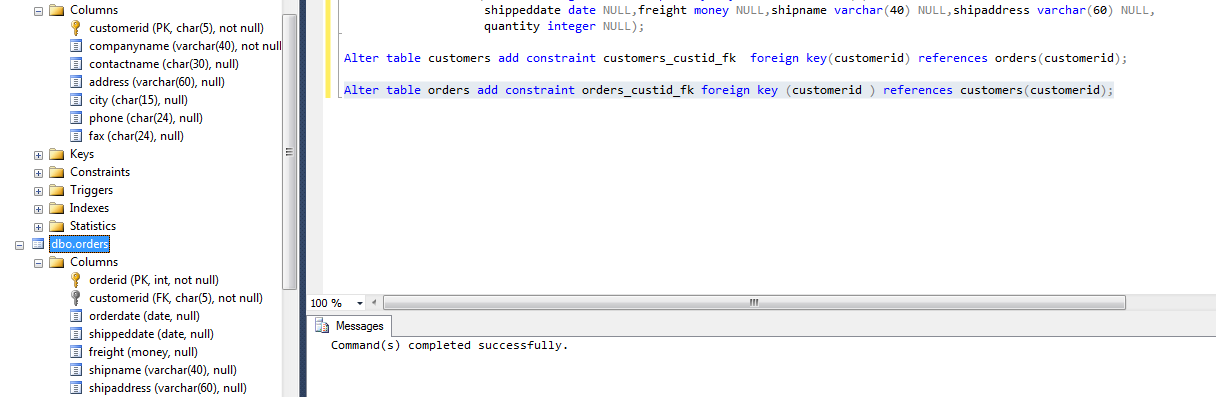
ANS:

ALTER TABLE orders ADD CONSTRAINT quantity CHECK( quantity BETWEEN 1 AND 30 );

1. Add a foreign key constraint in the **orders** table on **customerid** which references the **customerid** of **customer**s table

**ANS:**

Alter table orders add constraint orders\_custid\_fk foreign key (customerid ) references customers(customerid);



1. (25 points) Create a database **Library** based on the following diagram. Choose the appropriate action (NO ACTION,CASCADE,SET NULL, SET DEFAULT) for each referential integrity constraint, both for ON DELETE and ON UPDATE. Justify your choice. In the diagram, underlined attributes represent the primary key and links represent referential integrity.

****

ON Delete, ON Update done graphically using tool:

Integrity constraints between book(book\_id) and book\_authors(book\_id)

create table book(book\_id int primary key,title varchar(10) ,publishername varchar(10));

create table book\_authors(book\_id int,authorname varchar(10) primary key);

alter table book\_authors

add constraint bookauthor\_bookid\_fk foreign key(book\_id) references book(book\_id);

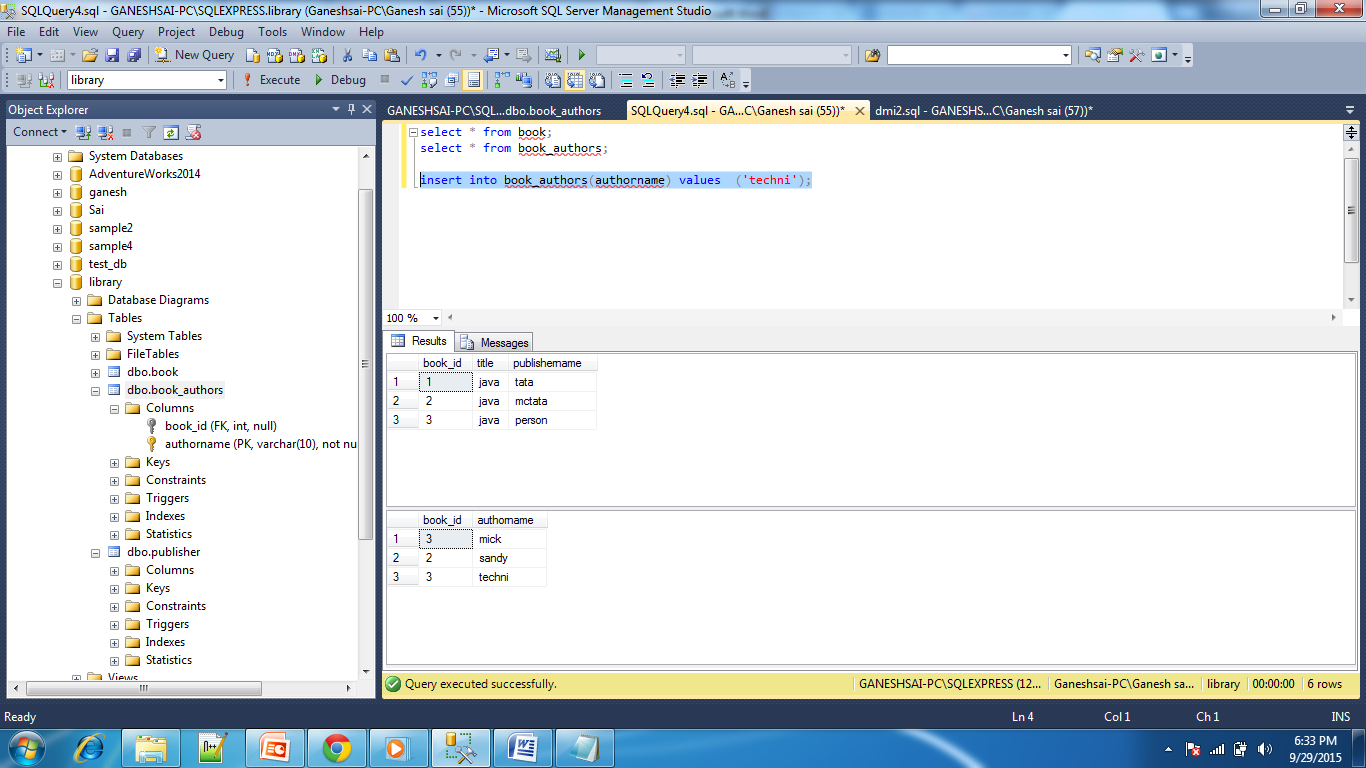
alter table book\_author

add constraint df\_book\_bookid

default 3 for book\_id;

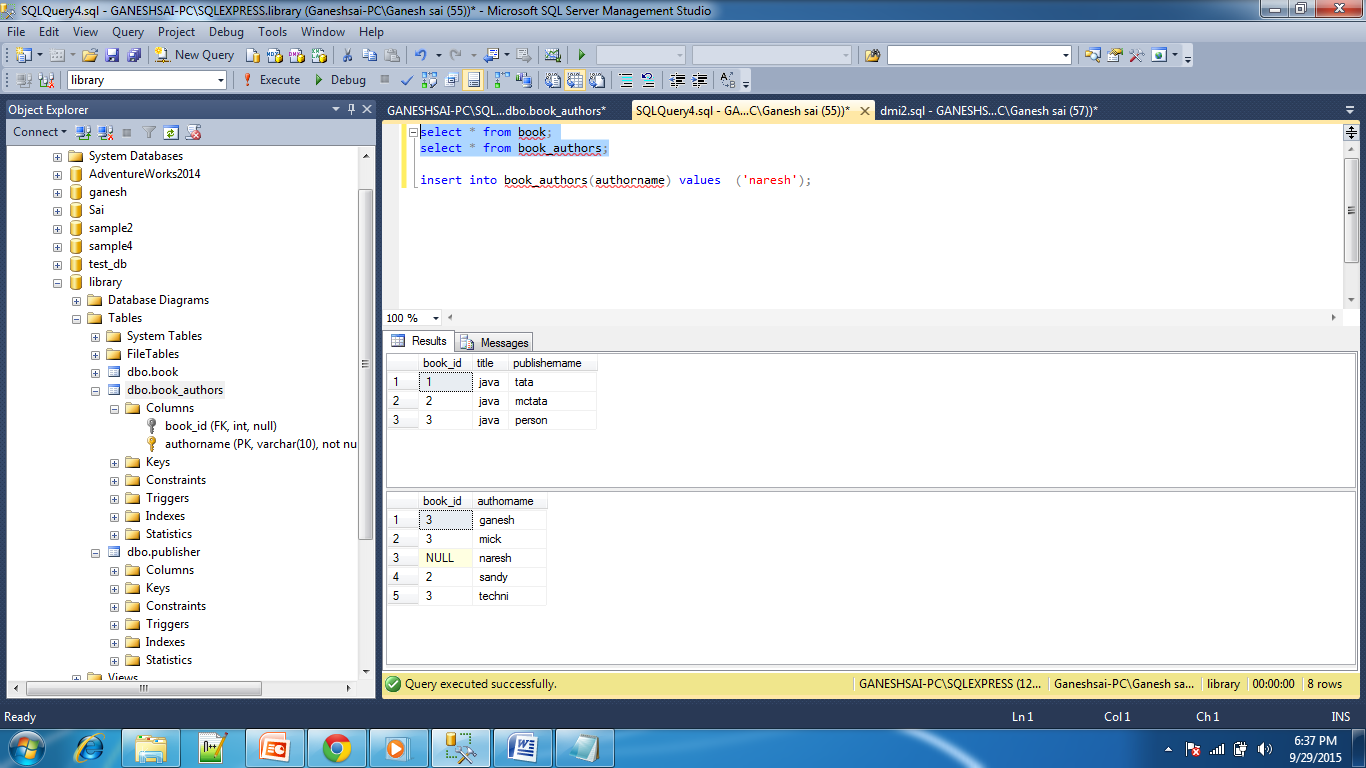
After setting default value to 3

insert into book\_authors(authorname) values ('techni');

****

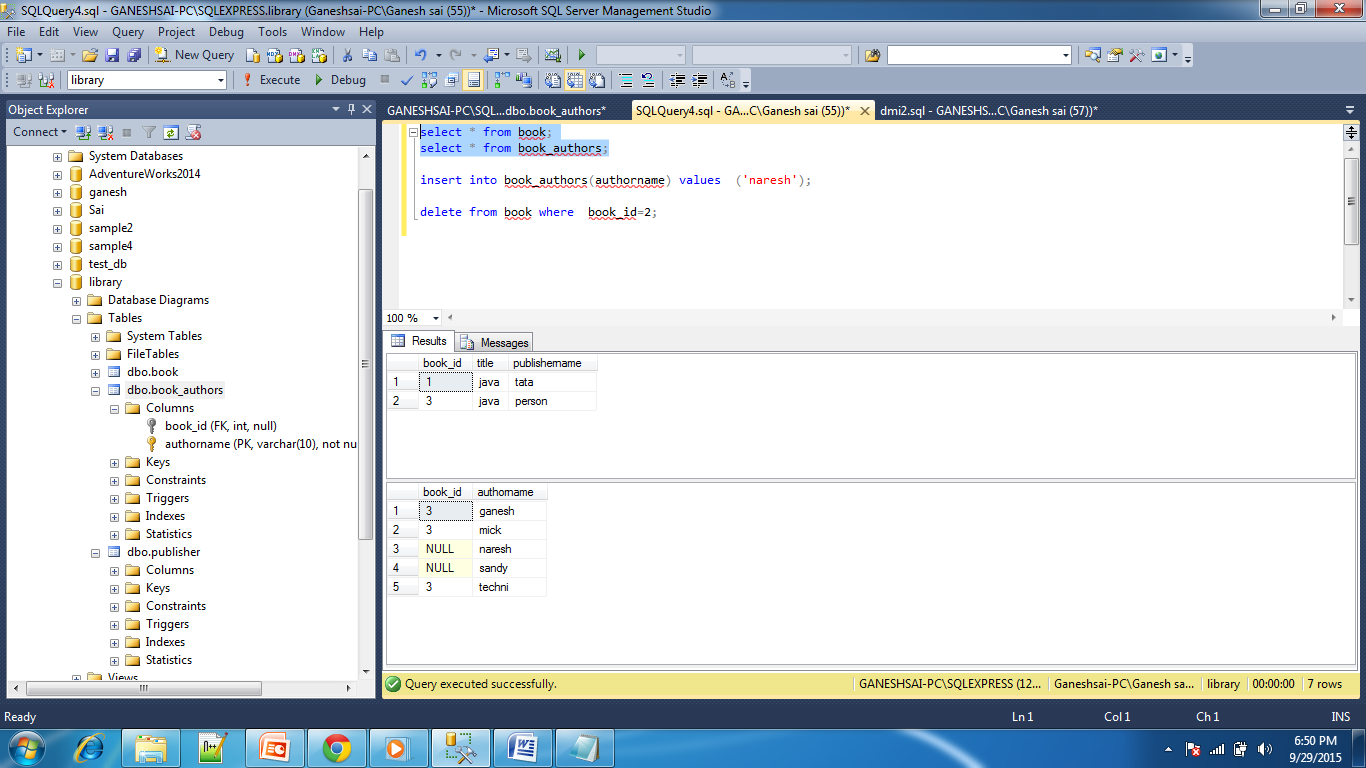
After setting to null

insert into book\_authors(authorname) values ('naresh');



CASCADE:

delete from book where book\_id=2;



Integrity constraints between book(publisher\_name) and publisher(name);

create table book(book\_id int primary key,title varchar(10) ,publishername varchar(10));

create table publisher(name varchar(10) primary key ,addres varchar(10),phone int);

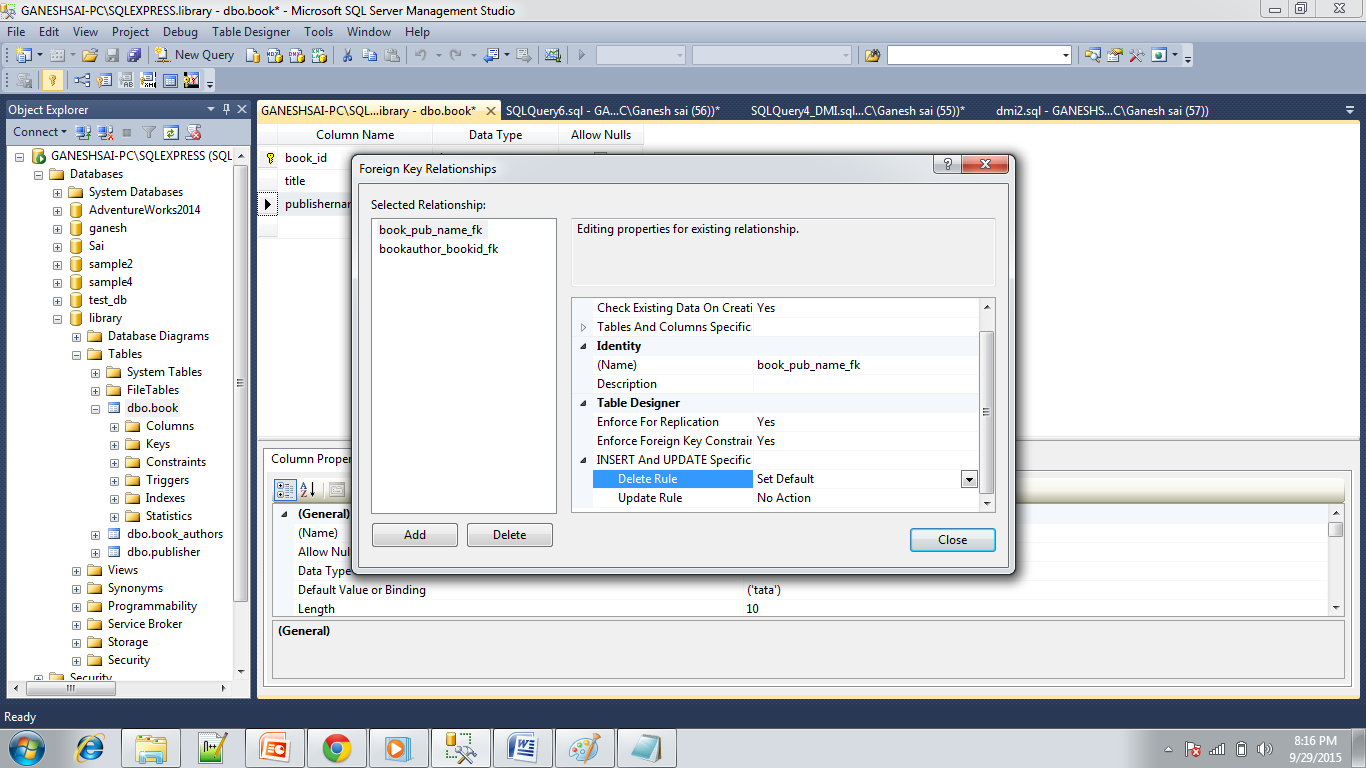
alter table book

add constraint book\_pub\_name\_fk foreign key(publishername) references publisher(name);

set default:

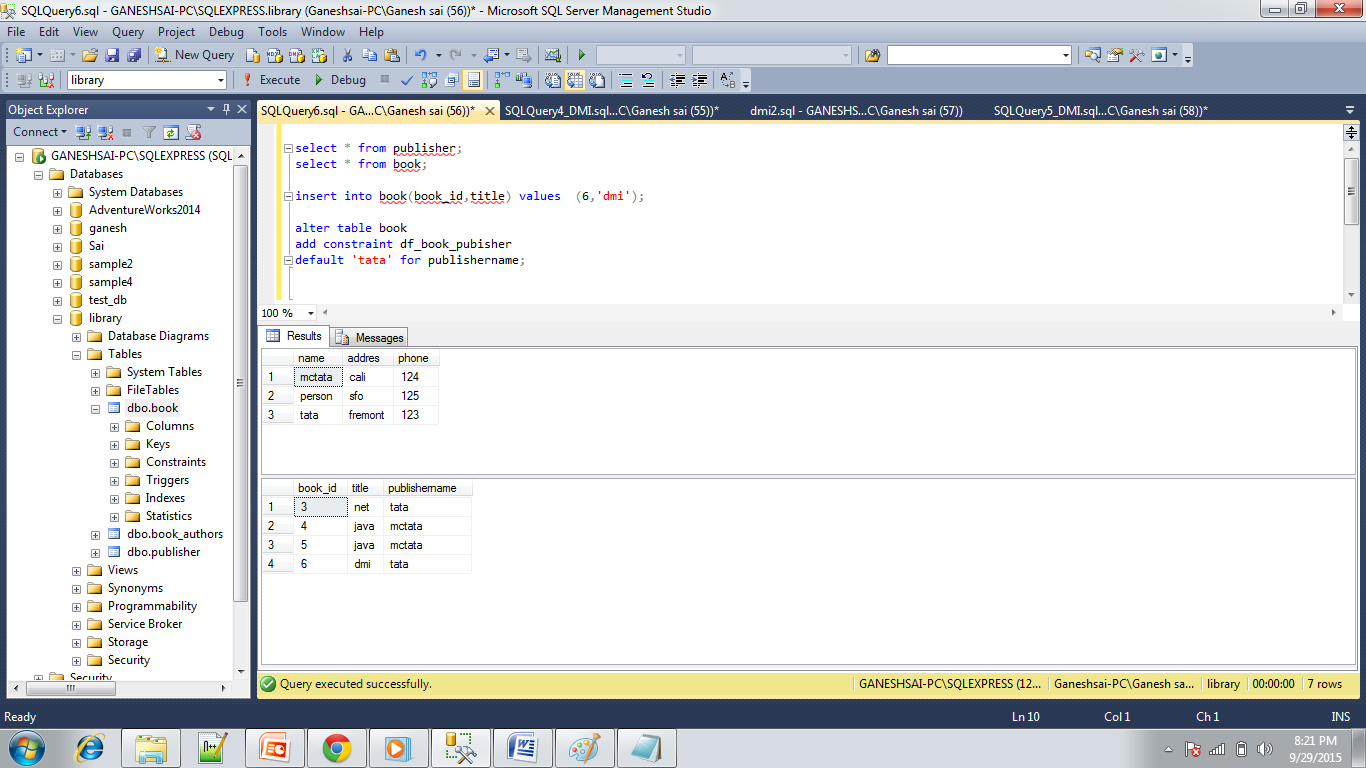
alter table book add constraint df\_book\_pubisher default 'tata' for publishername;

alter table book drop constraint df\_book\_pubisher;

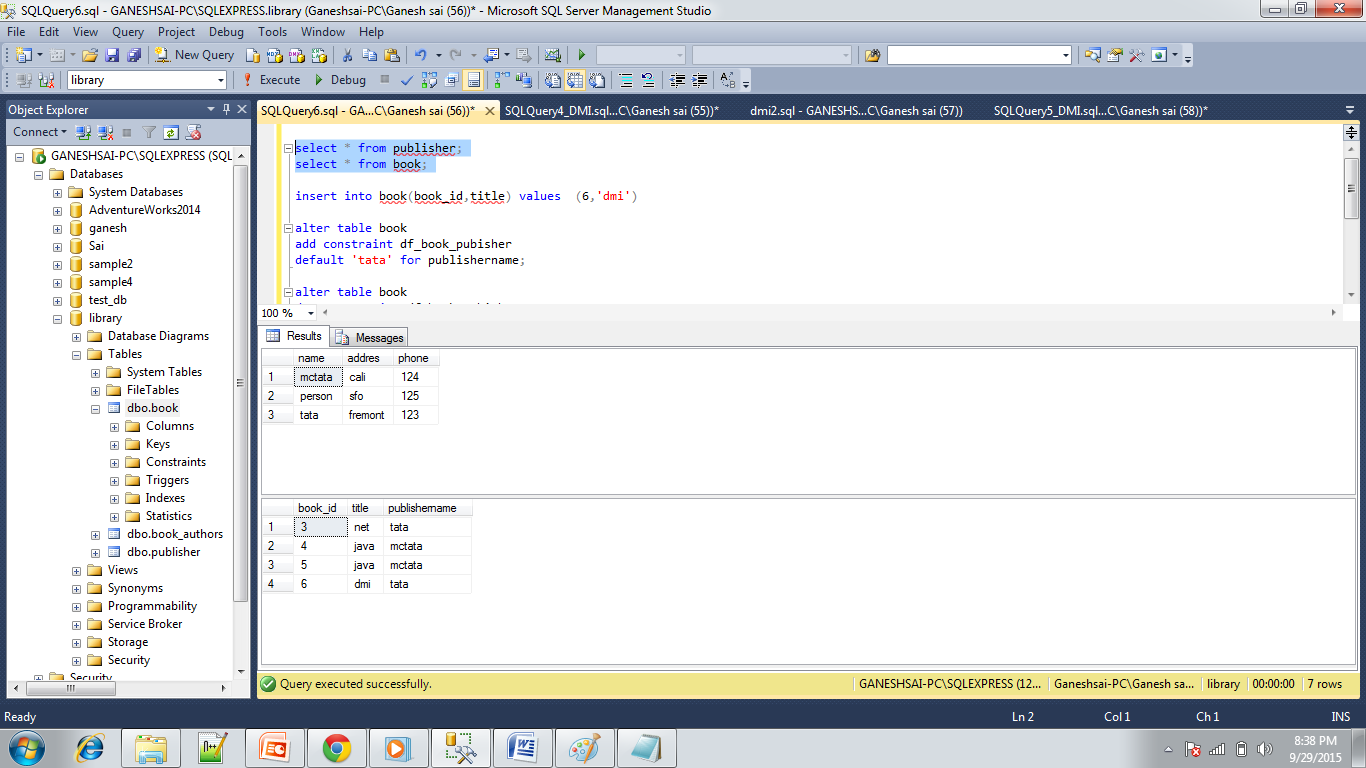


Default value will be displayed

insert into book(book\_id,title) values (6,'dmi');

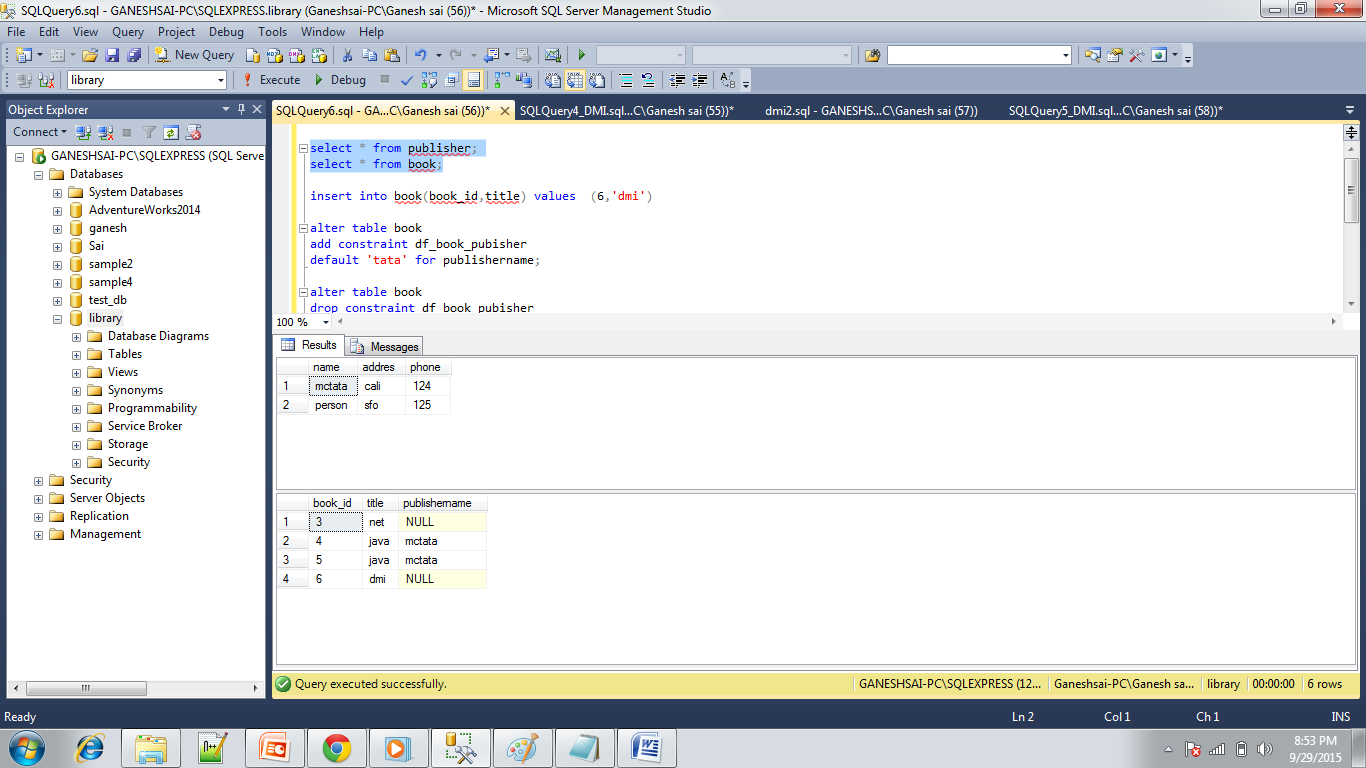


alter table book add constraint df\_book\_pubisher default 'tata' for publishername;



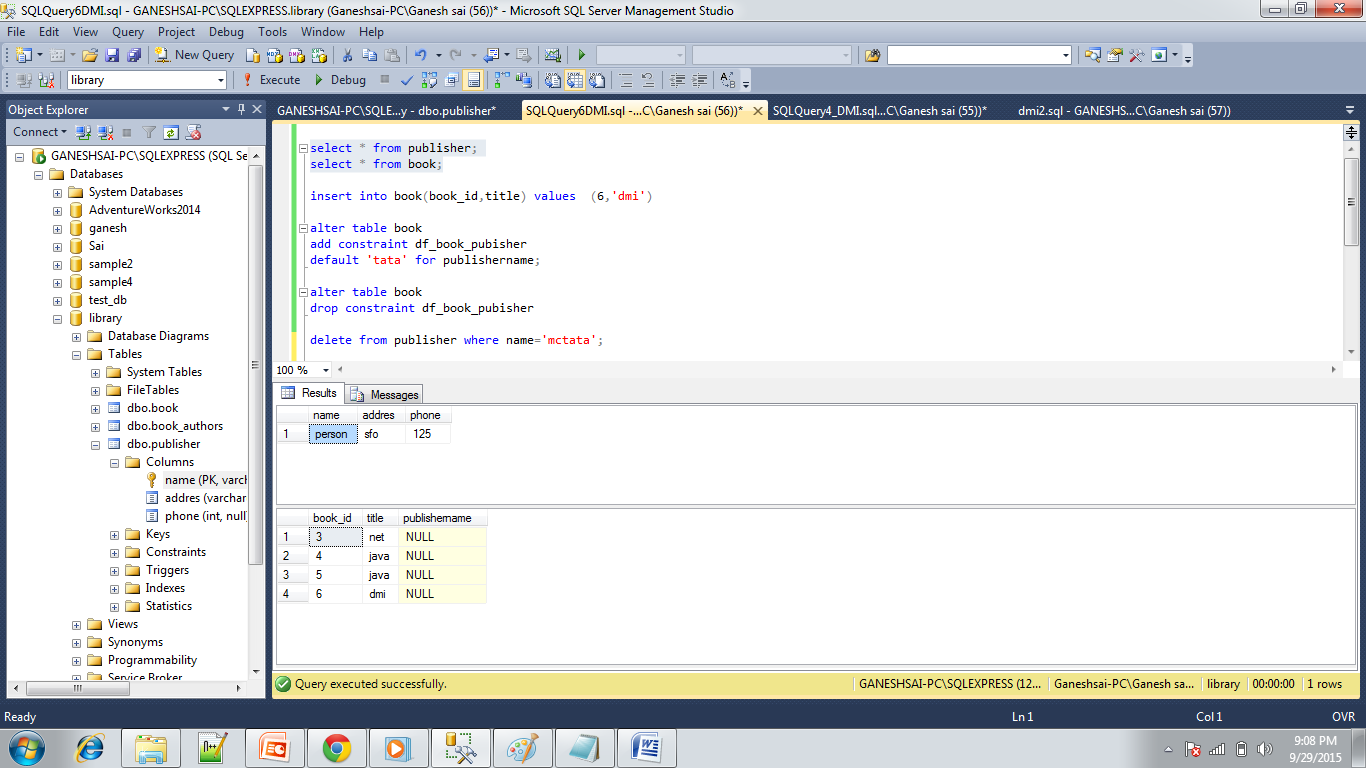
SET NULL:

delete from publisher where name='tata';



ON UPDATE CASCADE

delete from publisher where name='mctata';



Integrity constraints between book(book\_id) and book\_copies(book\_id);

create table book(book\_id int primary key,title varchar(10) ,publishername varchar(10));

Create table book\_copies(book\_id int,branch\_id int,noofcopies int);

alter table book\_copies

add constraint con\_bookcopies\_bookid

foreign key(book\_id) references

book(book\_id);

Between library\_branch(branch\_id) and book\_copies(branch\_id)

Create table book\_copies(book\_id int,branch\_id int,noofcopies int);

create table library\_branch (branch\_id int primary key, branch\_name varchar(10), addres varchar(10));

alter table book\_copies

add constraint con\_brnch\_id

foreign key(branch\_id) references library\_branch (branch\_id);