Part 1. Creating Tables:

Book

```
CREATE TABLE dbo.Book
(
   BookID INT NOT NULL
   ,Title VARCHAR(100) NOT NULL
   ,PublisherID INT NOT NULL
);
```

BookAuthors

```
CREATE TABLE dbo.BookAuthors
(
   BookID INT NOT NULL
   ,AuthorName VARCHAR(100) NOT NULL
);
```

BookCopies

```
CREATE TABLE dbo.BookCopies
(
   BookID INT NOT NULL
   ,BranchID INT NOT NULL
   ,NumOfCopies INT NOT NULL
);
```

BookLoans

```
CREATE TABLE dbo.BookLoans
(
   BookID INT NOT NULL
   ,BranchID INT NOT NULL
   ,CardNo INT NOT NULL
,DateOut DATE NULL
,DueDate DATE NULL
);
```

Publisher

```
CREATE TABLE dbo.Publisher
(
   PublisherID INT NOT NULL
   ,PublisherName VARCHAR(150) NOT NULL
   ,PubisherAddress VARCHAR (200) NOT NULL
   ,PublisherPhone INT NULL
);
```

LibraryBranch

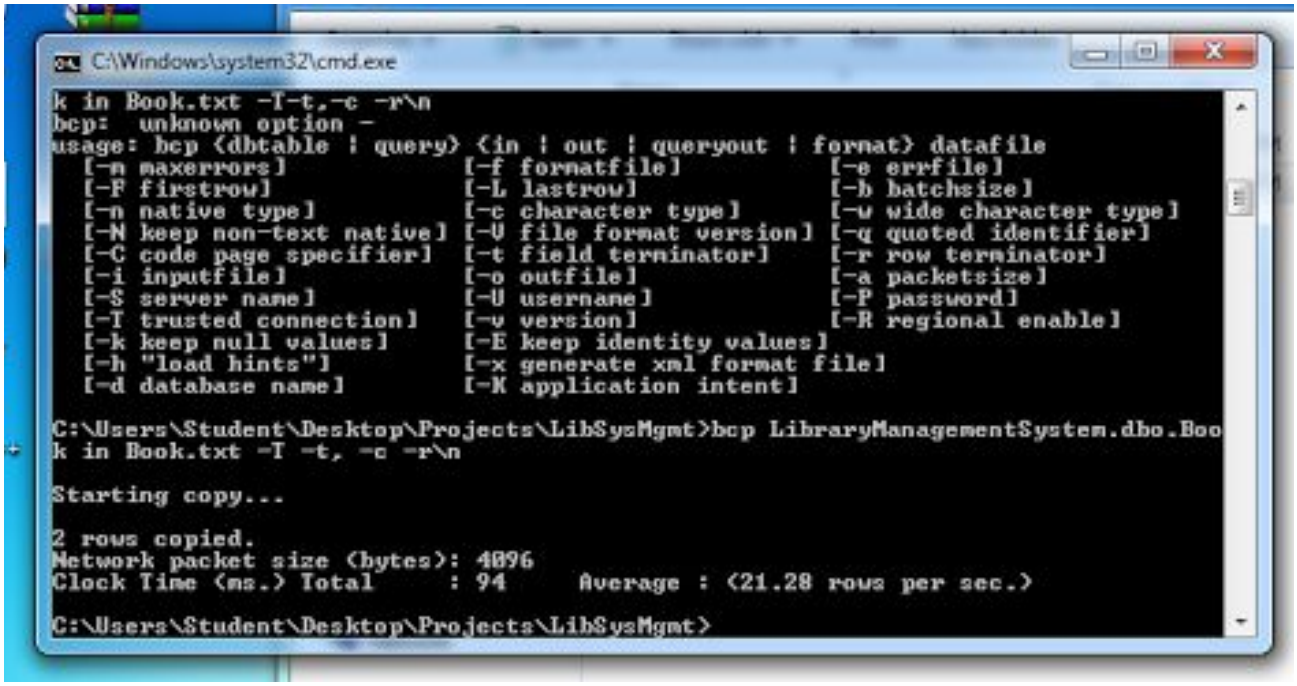
```
CREATE TABLE dbo.LibraryBranch
(
   BranchID INT NOT NULL
   ,BranchName VARCHAR(100) NOT NULL
   ,BranchAddress VARCHAR (200) NOT NULL
);
```

Borrower

```
CREATE TABLE dbo.Borrower
(
   CardNo INT NOT NULL
   ,BorrowerName VARCHAR(100) NOT NULL
   ,BorrowerAddress VARCHAR(200) NOT NULL
   ,BorrowerPhone INT NULL
);
```

Part 2. Inserting Data into Tables

A. BCP command that works to import data to tables:



B. SQL command for putting data in table from New Query:

```
INSERT INTO BookLoans (BookID, BranchID, CardNo, DateOut, DueDate)
VALUES (15,3,1256,'2016-06-24','2016-07-24');
```

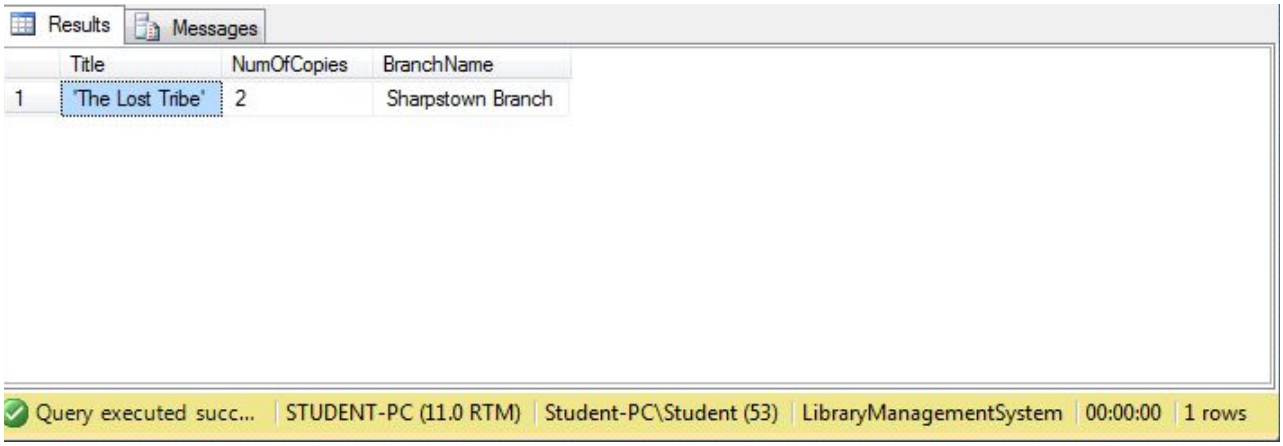
Part 3. Questions

1. How many copies of the book titled The Lost Tribe are owned by the library branch whose name is "Sharpstown"?

Query:

```
SELECT B.Title, C.NumOfCopies, L.BranchName
FROM LibraryBranch AS L
        INNER JOIN BookCopies AS C
        ON L.BranchID = C.BranchID
        INNER JOIN Book AS B
        ON C.BookID = B.BookID
WHERE L.BranchID = 1
AND B.BookID = 1
```

Returns:

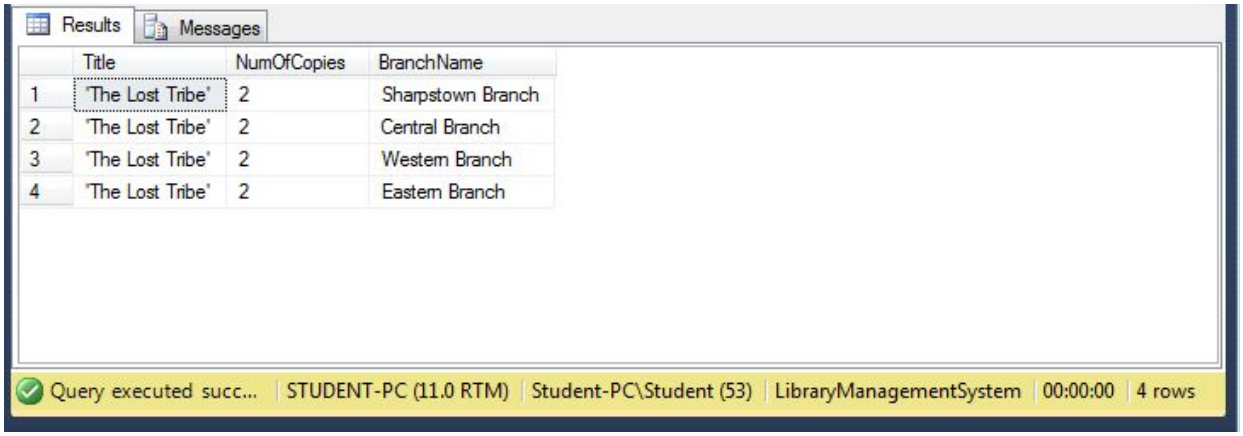


2. How many copies of the book titled The Lost Tribe are owned by each library branch?

Query:

```
SELECT B.Title, C.NumOfCopies, L.BranchName
FROM LibraryBranch AS L
        INNER JOIN BookCopies AS C
        ON L.BranchID = C.BranchID
        INNER JOIN Book AS B
        ON C.BookID = B.BookID
WHERE L.BranchID <=4
AND B.BookID = 1
```

Returns:

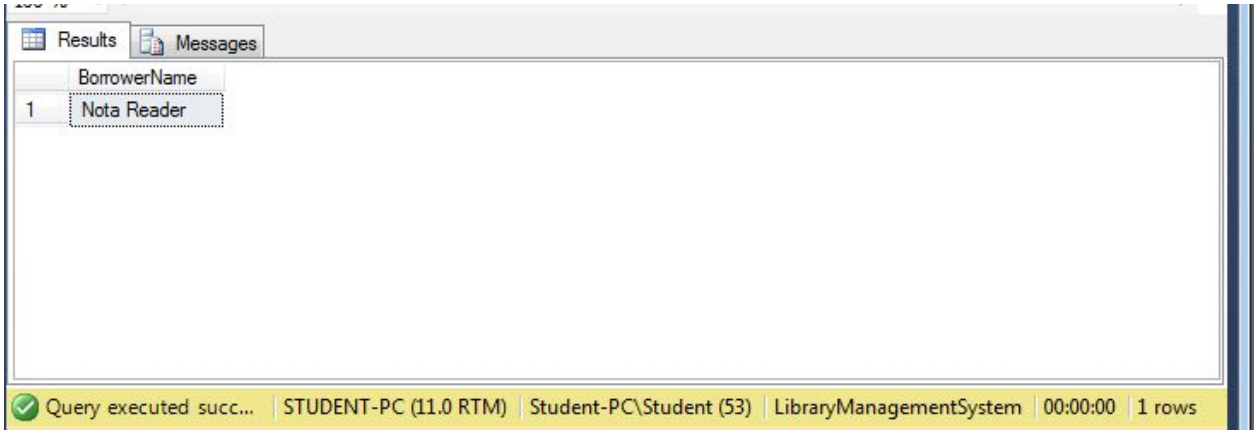


3. Retrieve the names of all borrowers who do not have any books checked out.

Query:

```
SELECT R.BorrowerName
FROM BookLoans AS L
        RIGHT OUTER JOIN Borrower AS R
        ON L.CardNo=R.CardNo
WHERE L.DateOut IS NULL
```

Returns:



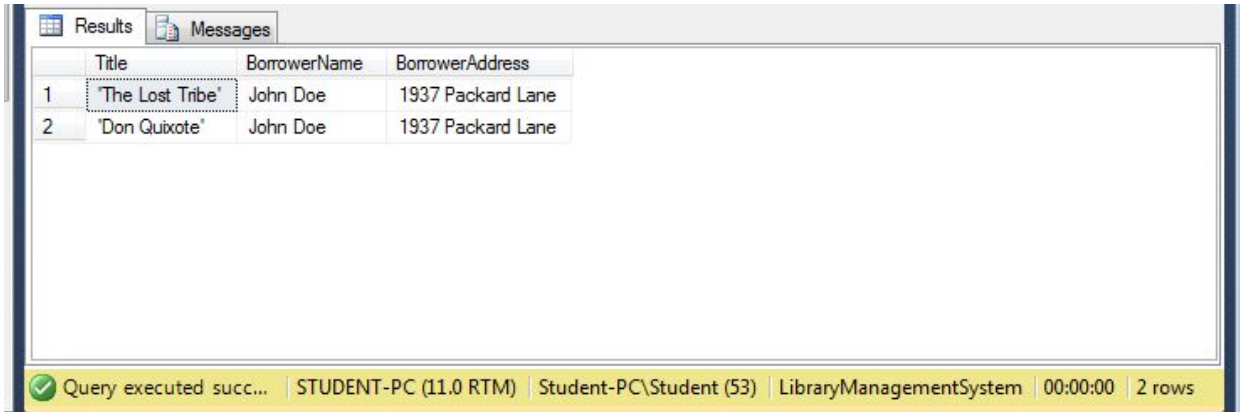
4. For each book that is loaned out from the "Sharpstown" branch and whose DueDate is today, retrieve the book title, the borrower's name, and the borrower's address.

Query:

```
SELECT B.Title, R.BorrowerName, R.BorrowerAddress
FROM BookLoans AS L
        INNER JOIN Book AS B
        ON L.BookID = B.BookID
        INNER JOIN Borrower AS R
        ON L.CardNo = R.CardNo
WHERE BranchID = 1
AND DueDate = '2016-11-21'
```

NOTE: I didn't enter any return dates of today, so I pretended 'today' was 2016-11-21 so I had some data to show.

Returns:



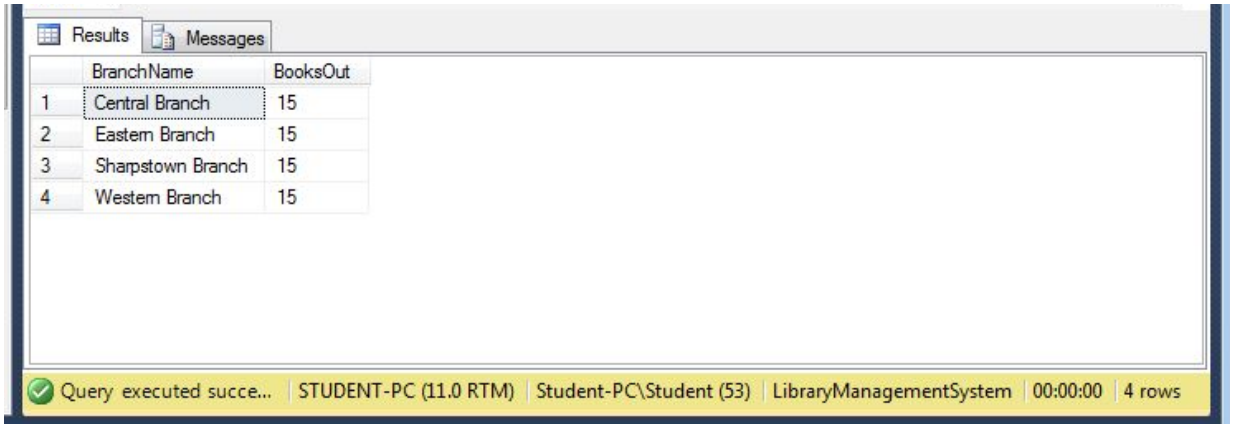
	Title	BorrowerName	BorrowerAddress
1	'The Lost Tribe'	John Doe	1937 Packard Lane
2	'Don Quixote'	John Doe	1937 Packard Lane

5. For each library branch, retrieve the branch name and the total number of books loaned out from that branch.

Query:

```
SELECT R.BranchName, COUNT(R.BranchName)
AS BooksOut
FROM LibraryBranch AS R
        INNER JOIN BookLoans AS L
        ON R.BranchID = L.BranchID
GROUP BY R.BranchName
```

Returns:



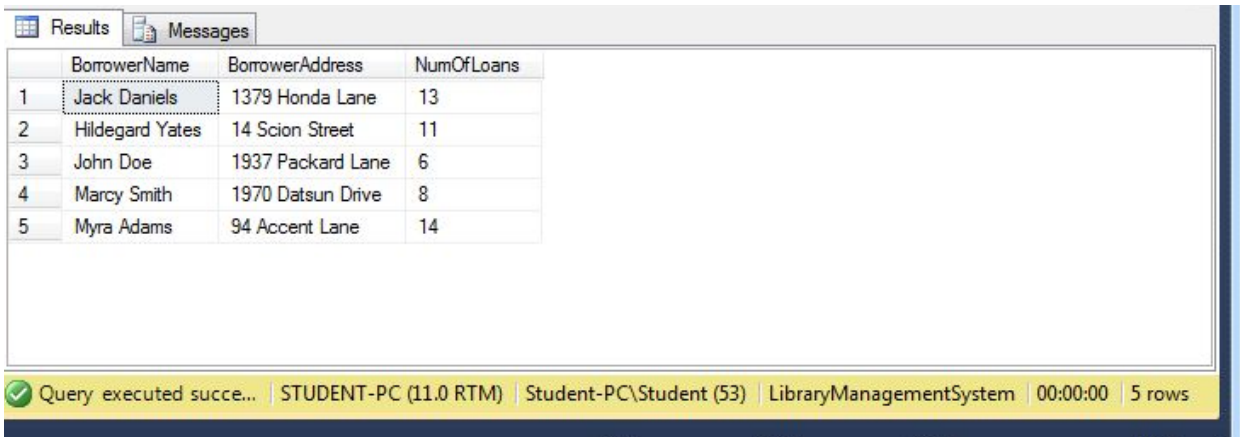
	BranchName	BooksOut
1	Central Branch	15
2	Eastern Branch	15
3	Sharpstown Branch	15
4	Western Branch	15

6. Retrieve the names, addresses, and number of books checked out for all borrowers who have more than five books checked out.

Query:

```
SELECT B.BorrowerName, B.BorrowerAddress,
COUNT(L.CardNo)
AS NumOfLoans
FROM (Borrower AS B
        INNER JOIN BookLoans AS L
        ON B.CardNo = L.CardNo)
GROUP BY B.BorrowerName, B.BorrowerAddress
HAVING COUNT(L.CardNo) > 5
```

Returns:

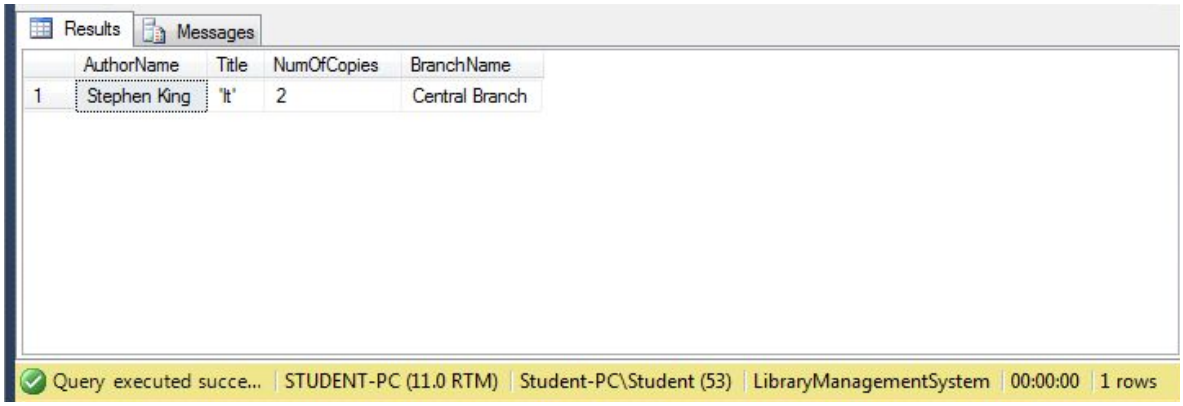


7. For each book authored (or co-authored) by "Stephen King", retrieve the title and the number of copies owned by the library branch whose name is "Central."

Query:

```
SELECT A.AuthorName, B.Title, C.NumOfCopies, L.BranchName
FROM BookAuthors AS A
        INNER JOIN BookCopies AS C
        ON A.BookID = C.BookID
        INNER JOIN LibraryBranch AS L
        ON C.BranchID = L.BranchID
        INNER JOIN Book AS B
        ON A.BookID = B.BookID
WHERE A.AuthorName = 'Stephen King'
AND L.BranchName = 'Central Branch'
```

Returns:



Part 4. Stored Proecdure

My stored procedure lists the books due on a given date at a given branch. I thought this would be a useful report for a librarian who needed to follow up with patrons whose books were due. It requires the librarian to input the name of branch and the date (in YYYY-MM-DD format) and returns the books due on that date, who has the books, and how to contact that borrower.

Here's the code I wrote to create the procedure:

```
CREATE PROC [dbo].[DailyDueByBranch] @Branch varchar(100), @Today date
AS
SELECT B.Title, N.BranchName, R.BorrowerName, R.BorrowerAddress, R.BorrowerPhone
FROM BookLoans AS L
        INNER JOIN LibraryBranch AS N
        ON L.BranchID = N.BranchID
        INNER JOIN Book AS B
        ON L.BookID = B.BookID
        INNER JOIN Borrower AS R
        ON L.CardNo = R.CardNo
WHERE N.BranchName = @Branch
AND DueDate = @Today
```

To test this, I entered the following code:

```
EXEC dbo.DailyDueByBranch 'Western Branch','2016-03-15'
```

Which should return information on the books due on the Ides of March, 2016.

The results show that the Ides of March is potentially an unlucky day for Myra Adams - she won't be laughing at 'The Divine Comedy' if she gets a fine: