|  |  |
| --- | --- |
| **Databases**  Diploma in IT / FI / CSF  Year 1 (2019/20) Semester 2 | Week **7** |
| **2** hours |
| **PRACTICAL 6**  **CREATE, INSERT, UPDATE and DELETE** | |

**OBJECTIVES**

At the end of this practical, you should know how to:

* construct SQL statement to insert, update and delete data

**REFERENCES**

Please refer to the following appendices in Database textbook.

* Appendix B: Tables in NP40 Book Rental System’s Database
* Appendix E: Data Dictionary for NP40 Book Rental System
* Database Textbook: pages 2-64 to 2-81
* PolyMall: Database Systems - Topic 7 Create Table

[7.1 Create Table Statement](https://polymall.polytechnic.edu.sg/webapps/scor-scormengine-BB5784d4c32fccb/delivery?action=launchPackage&course_id=_813_1&content_id=_36891_1)

* PolyMall: Database Systems - [Topic 3 Insert-Update-Delete](https://polymall.polytechnic.edu.sg/webapps/blackboard/content/listContentEditable.jsp?content_id=_30617_1&course_id=_813_1&mode=reset&courseTocLabel=Topic+3+Insert-Update-Delete)

Syntax:

**CREATE TABLE table (**

**column DATATYPE [ DEFAULT expression ] [ NULL | NOT NULL ] [ constraint ]**

**{, column DATATYPE [ DEFAULT expression ] [ NULL | NOT NULL ] [ constraint ] }**

**{, constraint }**

**)**

**INSERT INTO table [ ( column {, column} ) ]**

**[ VALUES ( expression {, expression} ) ] | [ sub\_select ]**

**UPDATE table1**

**SET column = expression**

**FROM table1 [ ,…n ]**

**[INNER JOIN table2 ON table1.field1 = table2.field2]**

**[INNER JOIN table3 ON table2.field2 = table3.field3]**

**[INNER JOIN tableX ON table\_name3.field3 = tableX.fieldX]**

**[WHERE search\_condition ]**

**DELETE FROM table**

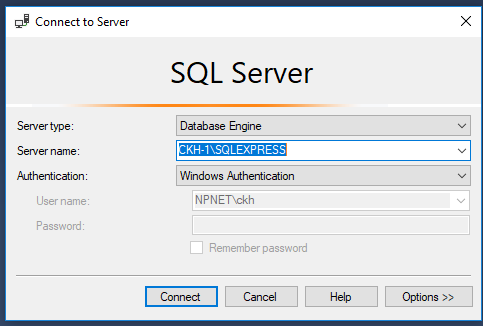
**[ WHERE search\_condition ]**

**QUESTIONS**

Construct SQL statements for the following questions.

**Connect to Local MS SQL Server**

1. Connect to your local MS SQL server as shown:



Note: **s101xxxxx** may be your computer name which you may be able to find out from **Control Panel -> System & Security -> System -> See the name of this computer -> Computer Name** (without the domain name e.g. npnet.np.edu.sg).

**Create Databases**

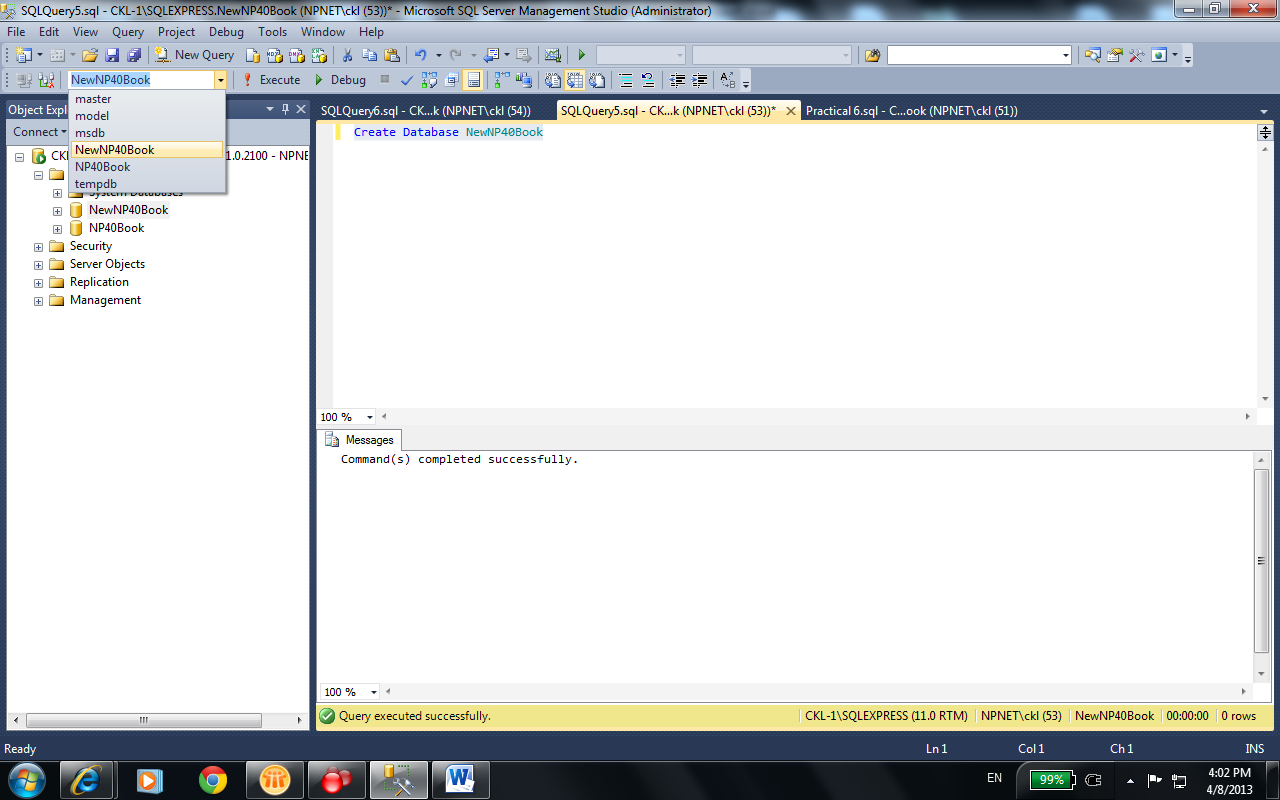
1. **This is to be done only if you have not done this in Week 1 or if the database has been corrupted**. Create the NP40Book database on your local MS SQL server by executing the database script file – **NP40Book\_Setup.sql** (downloadable from MeL under Week 1 Practical).
2. Create a new database on your local MS SQL server named as NewNP40Book by executing the following statement:

**CREATE DATABASE NewNP40Book**

You should see the following message.



Select the newly created database NewNP40Book as your working database from the drop down list of the available databases on your local machine, as shown:



**Create Tables**

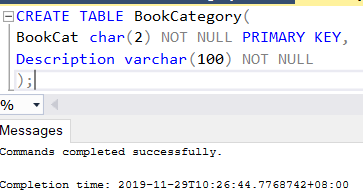
1. Create the table *BookCategory* as defined in the data dictionary on Appendix E (E-3). The table *BookCategory* is derived from the relation below:

*BookCategory* (BookCat, Description)

Note: Specify the primary key constraint for the table – *BookCategory*.

To remove a database table from a database owing to wrong specifications during its creation, you may issue the **DROP** SQL statement. For example:

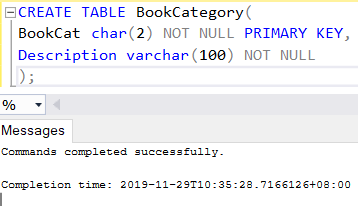
**DROP TABLE BookCategory**



1. Create the table *Publisher* as defined in the data dictionary on Appendix E (E-3). The table *Publisher* is derived from the relation below:

*Publisher* (PublisherID, Name)

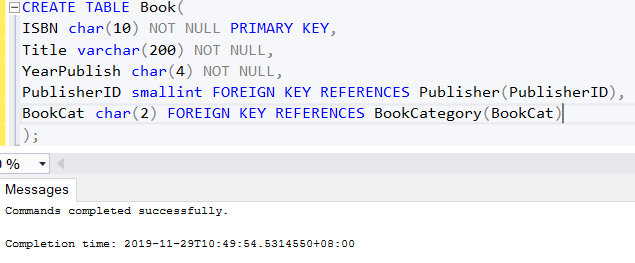
Note: Specify the primary key constraint for the table – *Publisher*.



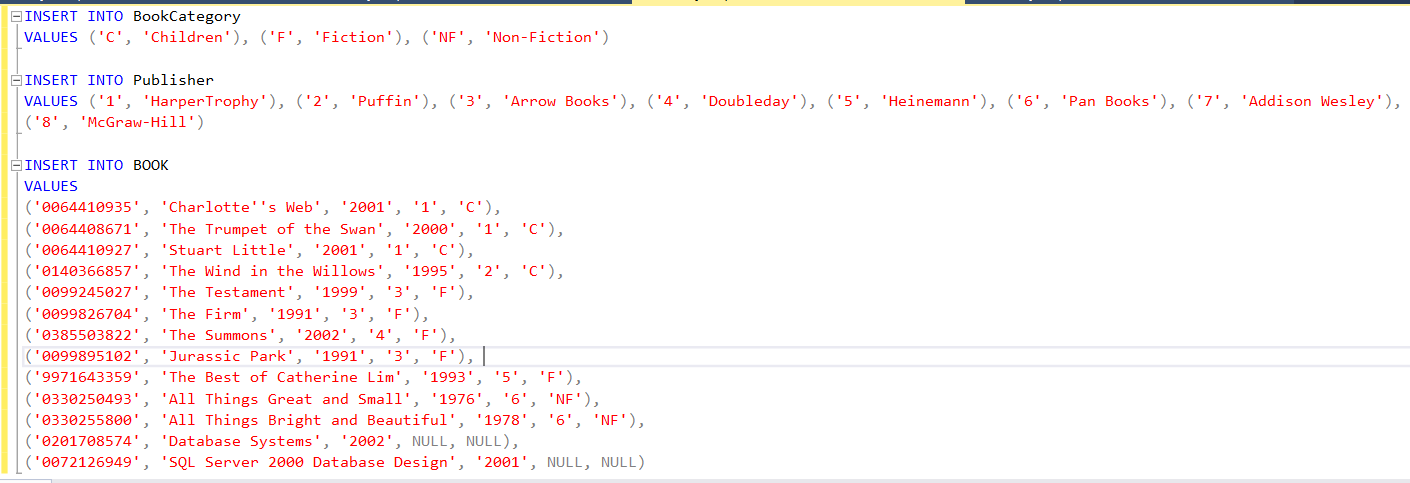
1. Create the table *Book* as defined in the data dictionary on Appendix E (E-3). The table *Book* is derived from the relation below:

*Book* (ISBN, Title, YearPublish, <PublisherID>, <BookCat>)

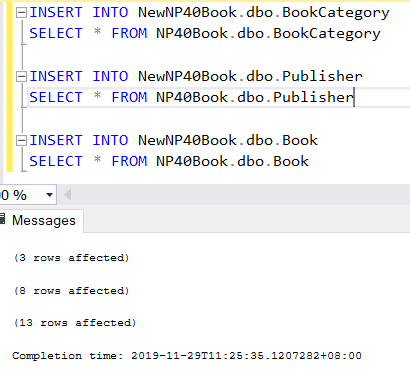
Note: Specify the primary key and foreign key constraints for the table - *Book*.



1. Load your tables with data from the following tables in the following sequence:
   1. NP40Book.dbo.BookCategory
2. NP40Book.dbo.Publisher
3. NP40Book.dbo.Book



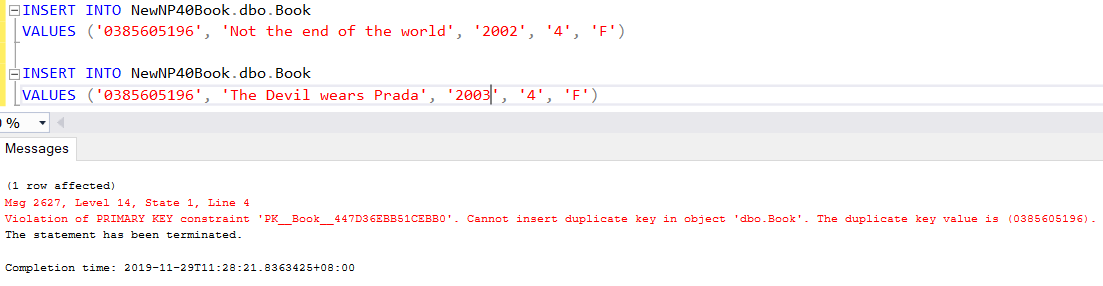
**OR**



1. Insert the following records into the *Book* table:

|  |  |  |
| --- | --- | --- |
| **ISBN** | 0385605196 | 0385605196 |
| **Title** | Not the end of the world | The Devil wears Prada |
| **YearPublish** | 2002 | 2003 |
| **PublisherID** | 4 | 4 |
| **BookCat** | F | F |

(Hint: Use the INSERT keyword)

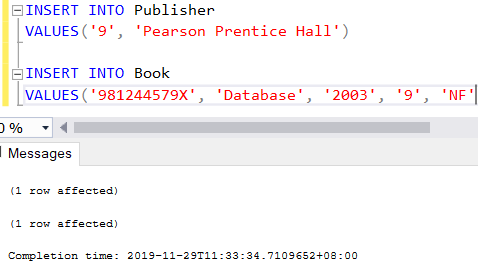


What happen when inserting the second book record? Explain why this happens?

The second book record failed to insert, because it has the same ISBN number as the first book, but ISBN must be unique.

1. Insert the following record into the *Publisher* and *Book* tables:

|  |  |
| --- | --- |
| **ISBN** | 981244579X |
| **Title** | Database |
| **YearPublish** | 2003 |
| **PublisherID** | 9 |
| **Name** | Pearson Prentice Hall |
| **BookCat** | NF |

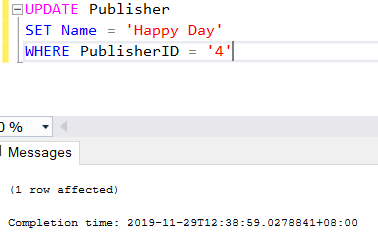


Which table should you insert the data into first? Why?

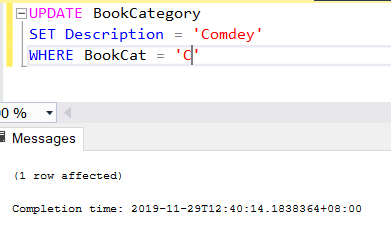
Need to insert to publisher table, as book have foreign key link to publisher.

1. Amend the publisher **name** of Publisher ID (4) from ‘Doubleday' to 'Happy Day'.

(Hint: Use the UPDATE keyword)



1. Amend the book category **description** of the Book category (‘C’) from ‘Children’ to 'Comedy'.

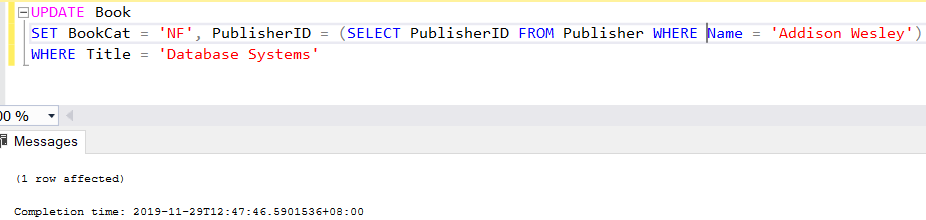


1. More data is available for the title 'Database Systems':

|  |  |
| --- | --- |
| **Publisher** | Addison Wesley |
| **BookCat** | NF |

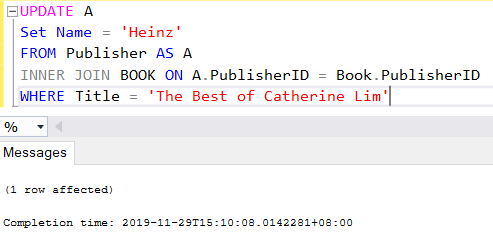
Amend the record to reflect these new data. You may assume that there is only one book in the database with this title. Verify that the amendment is done correctly.

(Hint: You have to use subquery (SELECT) in order to retrieve the PublisherID given only the name of the publisher.)



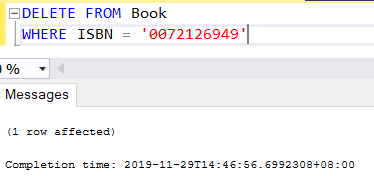
1. Amend the publisher for the book 'The Best of Catherine Lim' to 'Heinz'.

(Hint: Use the INNER JOIN keyword)

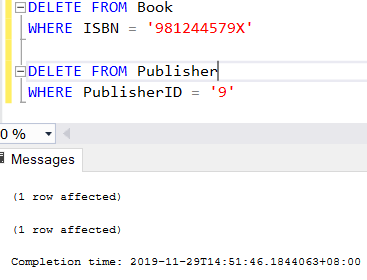


1. Remove *ISBN* – **'0072126949'** from the *Book* table.

(Hint: Use the DELETE keyword)



1. The rows that were inserted in Question 9 were not supposed to be there in the first place. Amend the database to reflect this. Row from which table should be removed first? Why?

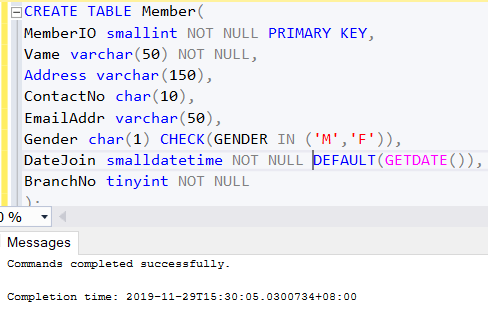


Should delete from book first as if you delete from publisher first, the book table table have error as it the foreign key that looking for publisherID ‘9’ cannot be found

1. Create the table *Member* as defined in the data dictionary on Appendix E (E-3). The table *Book* is derived from the relation below:

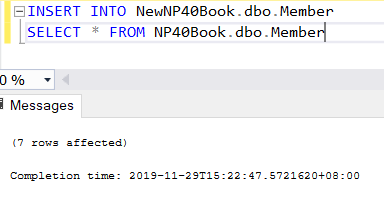
*Member* (MemberID, Name, Address, ContactNo, EmailAddr, Gender, DateJoin, <BranchNo>)

Note: Specify the primary key constraint but do not specify the foreign key constraint for the table - *Member*.



**\*there is a typo for MemberID, in the query I put MemberIO. I have fix the name in my database alr**

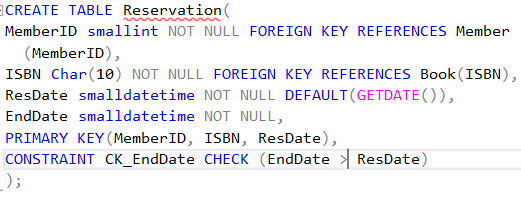
1. Load your table with data from NP40Book.dbo.Member table.



* + 1. Figure 1 is the Data Dictionary for the Reservation relation. Write the SQL code to create this table in the NEWNP40Book database, include any necessary constraints.

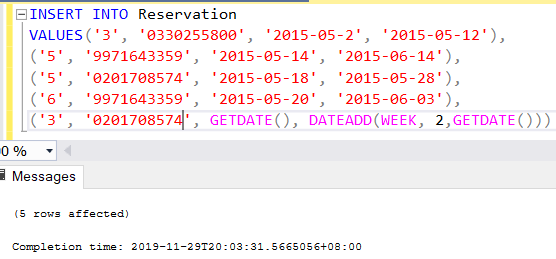
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Attribute***  ***Name*** | ***Description*** | ***Data Type and Length*** | ***Constraint*** | ***Null value*** |
| MemberID | Unique identifier for each member | Int | Primary Key  Foreign Key -> Member(MemberID) | No |
| ISBN | Unique identifier for each book | Char(10) | Primary Key  Foreign Key -> Book(ISBN) | No |
| ResDate | The date the reservation is made | Datetime | Primary Key, Default to today’s date | No |
| EndDate | The date after which the reservation is not required | Datetime | Not earlier than ResDate | No |

Figure 1: Data Dictionary for Reservation relation



19. Load the table created above with the following data:

|  |  |  |  |
| --- | --- | --- | --- |
| **MemberID** | **ISBN** | **ResDate** | **EndDate** |
| 3 | 0330255800 | 2 May 2015 | 12 May 2015 |
| 5 | 9971643359 | 14 May 2015 | 14 June 2015 |
| 5 | 0201708574 | 18 May 2015 | 28 May 2015 |
| 6 | 9971643359 | 20 May 2015 | 3 June 2015 |
| 3 | 0201708574 | today | 2 weeks from today |



1. List the details of all books and reservations that have been reserved from 15th May 2015 to 30th May 2015. You are required to display the ResDate and EndDate as MM/YYYY format.

