**CSC263 Database Systems**

**Laboratory Assignment #4**

**Goals:**

* To complete the creation of the Publication database with all necessary business rules.
* To learn how to generate a SQL script to create a database schema.

**Objectives:** After successfully completing the lab, one should be able to

* + write a script file to create a database schema, including the declaration of all necessary enforcement, that guard the entity integrity and referential integrity for the database,
  + use DML to query the database.

**Requirements:**

Based on the discussions we had and the three labs you have done, modify and enhance the SQL script to create a database for the following “Publication” mini-world.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Title** | **Year** | **Edition** | **ISBN** | **Authors** |
| Visual Basic 2005 How to Program | 2006 | 3rd | 0131869000 | Harvey Deitel; Paul Deitel; |
| Visual C# 2005 How to Program | 2006 | 2nd | 0131525239 | Harvey Deitel; Paul Deitel; |
| Java How to Program | 2007 | 7th | 0132222205 | Harvey Deitel; Paul Deitel; |
| C++ How to Program | 2005 | 5th | 0131857576 | Harvey Deitel; Paul Deitel; |
| C How to Program | 2007 | 5th | 0132404168 | Harvey Deitel; Paul Deitel; |
| Internet & World Wide Web How to Program | 2004 | 3rd | 0131450913 | Harvey Deitel; Paul Deitel; Andrew Goldberg |
| Operating Systems | 2004 | 3rd | 0131828274 | Harvey Deitel; Paul Deitel; David Choffnes |
| Android How to Program | 2017 | 3rd | 0134444302 | Harvey Deitel; Paul Deitel; |
| Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud | 2016 | 1st | 0134175395 | William Stallings |
| Wireless Communication Networks and Systems | 2016 | 1st | 0133594173 | Cory Beard; William Stallings |
| Computer Organization and Architecture | 2016 | 10th | 0134101618 | Williams Stallings |
| Operating Systems | 2012 | 7th | 013230998X | Williams Stallings |
| Data and Computer Communications | 2014 | 10th | 0133506487 | Williams Stallings |
| Cryptography and Network Security | 2014 | 6th | 0133354695 | Williams Stallings |
| Network Security Essentials | 2014 | 5th | 0133370437 | Williams Stallings |
| Cryptography and Network Security | 2017 | 7th | 0134444280 | Williams Stallings |
| Database Systems: Design, Implementation, & Management | 2014 | 11th | 1285196147 | Carlos Coronel, Steven Morris |
| Database Systems: Design, Implementation, & Management | 2002 | 7th | 1418835935 | Peter Rob, Carlos Coronel |
| Fundamentals of Database Systems | 2017 | 7th | 0133970779 | Ramez Elmasri, Sham Navathe |
| Fundamentals of Database Systems | 2006 | 5th | 0321369572 | Ramez Elmasri, Sham Navathe |
| Fundamentals of Database Systems | 2003 | 4th | 0321122267 | Ramez Elmasri, Sham Navathe |

* Submit a lab report that contains the following items:

1. List ALL necessary constraints (domain, entity, and referential) and business rules.

Domain Constraints:

Author:

A\_ID (PRIMARY KEY

Book:

ISBN (PRIMARY KEY)

A\_writes\_B:

PRIMARY KEY is the composite Key of Author and Book

Entity Constraints:

Author:

A\_ID (PRIMARY KEY NOT NULL/UNIQUE)

lastName (NOT NULL)

firstName(NOT NULL)

Book:

ISBN (PRIMARY KEY/UNIQUE/NOT NULL)

Title (NOT NULL)

Edition(NOT NULL)

Year(NOT NULL)

A\_writes\_B:

A\_ID(FOREIGN KEY/NOT NULL/UNIQUE)

ISBN(FOREIGN KEY/NOT NULL/UNIQUE)

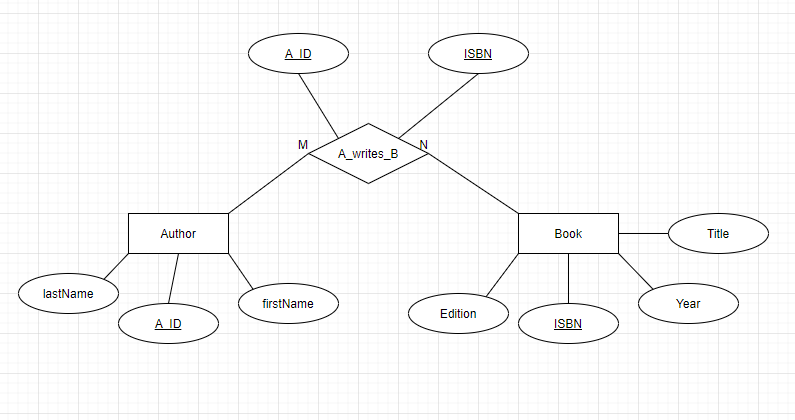
Referential Constraints:

A\_WRITES\_B

A\_ID (REFERENTIAL INTEGRITY CONSTRAINT references Author (A\_ID))

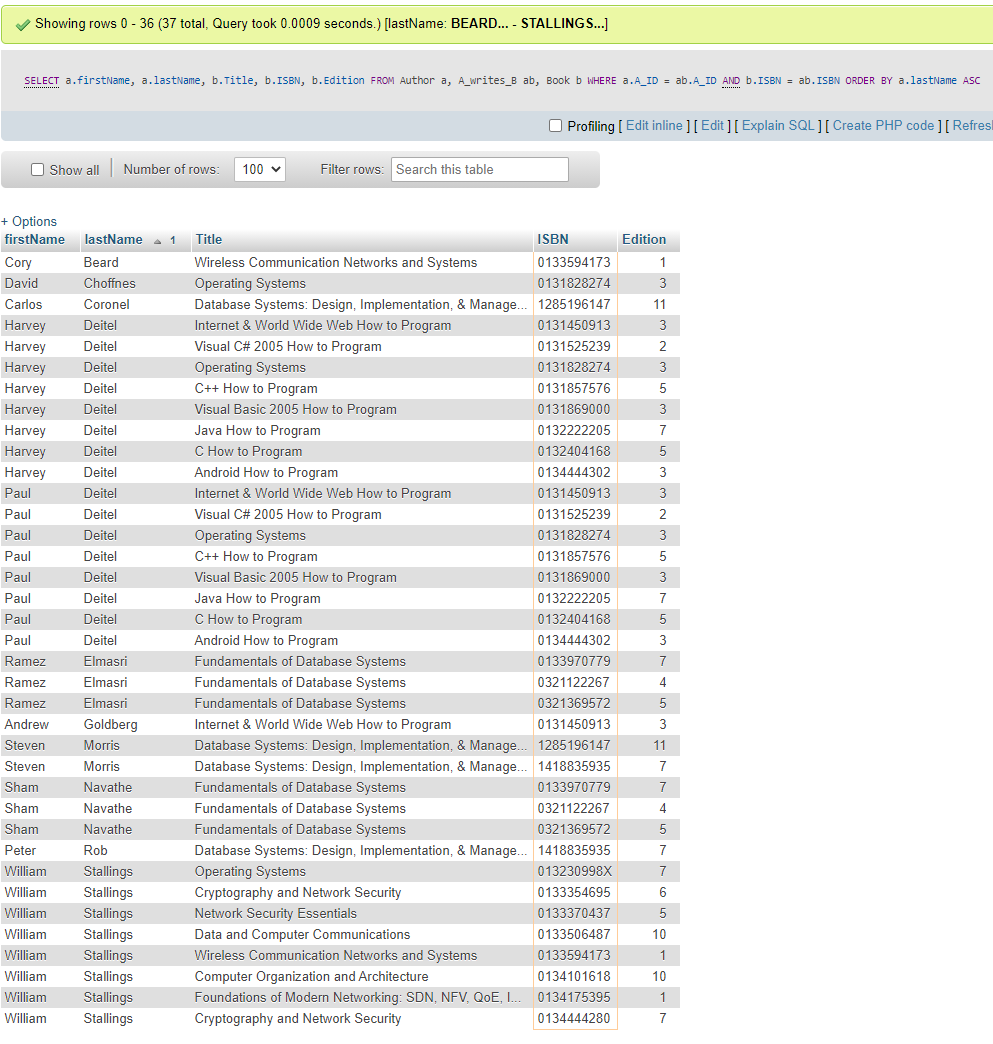
ISBN (REFERENTIAL INTEGRITY CONSTRAINT references ISBN (ISBN))

1. A complete ERD that shows ALL necessary constraints and rules.



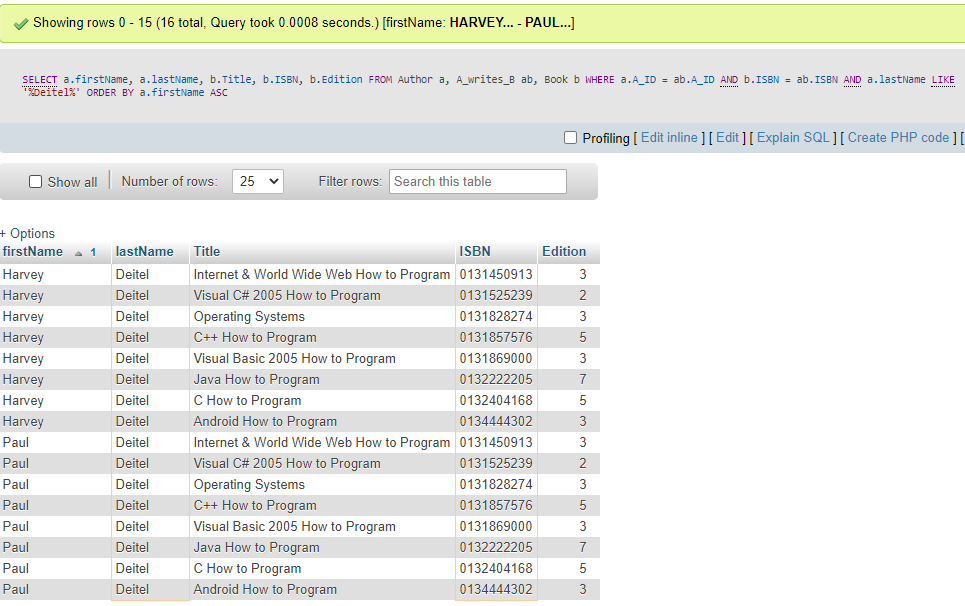
1. A complete schema for the database that shows all necessary constraints specified in point 1 above.
2. A complete listing of the database creation SQL script.

* Submit two SQL script files:
  1. SQL script that creates all tables and constraints (no script that populating the database)
  2. SQL script that populates the database (every table)
* Generate the following SQL queries and report your queries and the execution results
  1. List every author and his/her book information in the following format. The list must be in the alphabetical order of the authors’ last names.

**Query Used**: “ SELECT a.firstName, a.lastName, b.Title, b.ISBN, b.Edition FROM Author a, A\_writes\_B ab, Book b WHERE a.A\_ID = ab.A\_ID AND b.ISBN = ab.ISBN ORDER BY a.lastName ASC ”

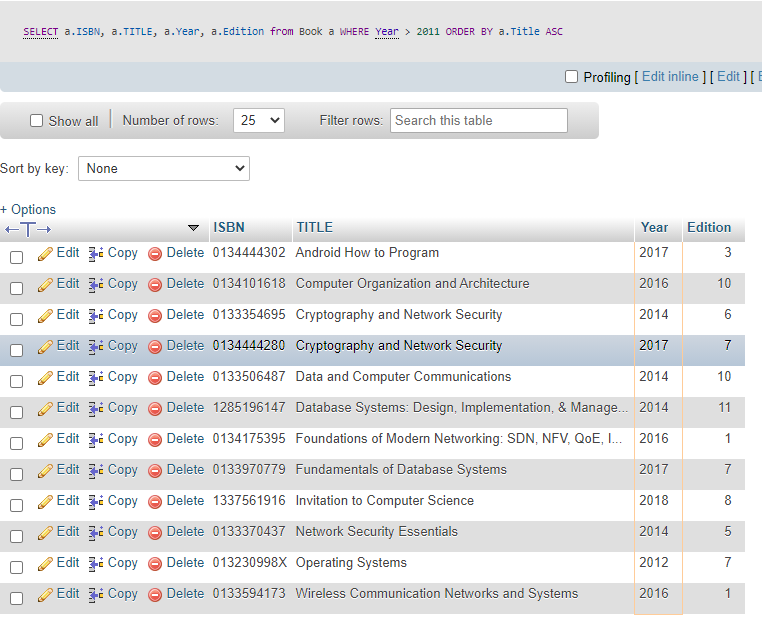
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| First Name | Last Name | Title | ISBN | Edition |

* 1. List every author whose last name is Deitel in the following format. The list must be in the alphabetical order of the authors’ first names.

**Query Used:** “ SELECT a.firstName, a.lastName, b.Title, b.ISBN, b.Edition FROM Author a, A\_writes\_B ab, Book b WHERE a.A\_ID = ab.A\_ID AND b.ISBN = ab.ISBN AND a.lastName LIKE '%Deitel%' ORDER BY a.firstName ASC ”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| First Name | Last Name | Title | ISBN | Edition |

* 1. List every book that was published in 2011 and later. The list must be in the alphabetical order of the books’ titles.

**Query Used:** “ SELECT a.ISBN, a.TITLE, a.Year, a.Edition from Book a WHERE Year > 2011 ORDER BY a.Title ASC ”

|  |  |  |  |
| --- | --- | --- | --- |
| ISBN | Title | Year | Edition |

* 1. List every book that was authored or co-authored by “Williams Stallings”. The list must be in the decreasing order of the books’ publication years.

**Query Used:** “ SELECT b.ISBN, b.Title, b.Year, b.Edition FROM Author a, A\_writes\_B ab, Book b WHERE a.A\_ID = ab.A\_ID AND b.ISBN = ab.ISBN AND a.lastName LIKE '%Stallings%' and firstName LIKE '%William%' ORDER BY b.Year DESC ”

|  |  |  |  |
| --- | --- | --- | --- |
| ISBN | Title | Year | Edition |

* 1. Generate and execute necessary SQL queries to add the following information to the database.

***Invitation to Computer Science*** 8th Edition

by Michael Schneider and Judith L. Gersting Cengage, 2019

ISBN-10: 1337561916

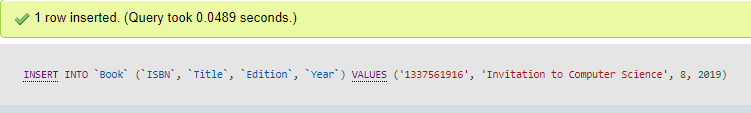
Note: There are multiple queries involved in this task.

1. Search the database to see if the book has already existed. **(DOES NOT EXIST)**

**Query Used:** “ SELECT a.Title, a.ISBN, a.Year, a.Edition from Book a WHERE a.Title LIKE '%Invitation to Computer Science%' AND a.ISBN LIKE '%1337561916%' ”

1. If it does not exist in the database, add it to the database; otherwise you can stop here.

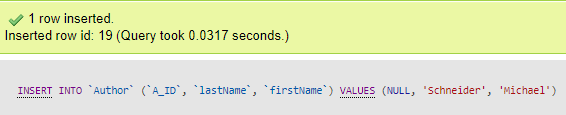
Query Used: “ INSERT INTO `Book` (`ISBN`, `Title`, `Edition`, `Year`) VALUES ('1337561916', 'Invitation to Computer Science', 8, 2019) ”



1. Search the database to see if “Michael Schneider” already exist. If it does, find the author’s A\_ID. And add one row in A\_Writes\_B table; if the author does not exist, add the author to the database, obtain the A\_ID and add a row in A\_Writes\_B table.

**Query Used to find if he exists:** “ SELECT a.firstName, a.lastName FROM Author a WHERE a.firstName LIKE '%Michael%' AND a.lastName LIKE '%Schneider%' ”

**Query Used to add him:** “ INSERT INTO `Author` (`A\_ID`, `lastName`, `firstName`) VALUES (NULL, 'Schneider', 'Michael') “



**Query Used to add row to A\_writes\_B:** “ INSERT INTO `A\_writes\_B` (`A\_ID`, `ISBN`) VALUES (19, '1337561916') “

1. Repeat (3) for “Judith L. Gersting”

**Query Used to find if he exists:** “ SELECT a.firstName, a.lastName FROM Author a WHERE a.firstName LIKE '%Judith%' AND a.lastName LIKE '%Gersting%' ”

**Query Used to add her:** “ INSERT INTO `Author` (`A\_ID`, `lastName`, `firstName`) VALUES (NULL, ‘Gersting’, ‘Judith') “

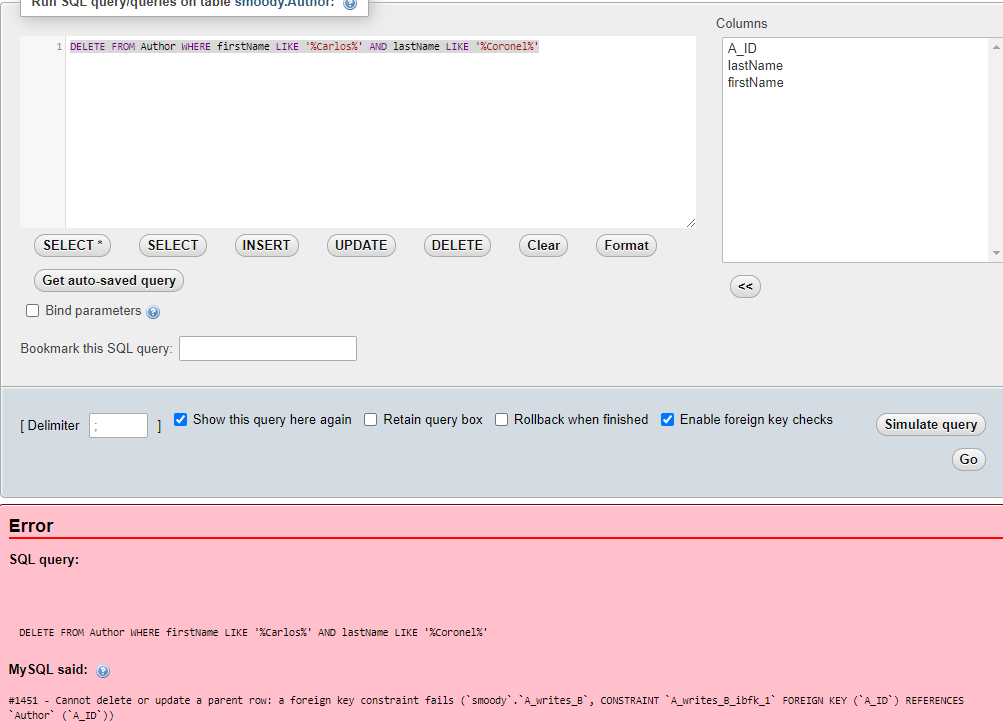
**Query Used to add row to A\_writes\_B:** “ INSERT INTO `A\_writes\_B` (`A\_ID`, `ISBN`) VALUES (20, '1337561916') ”

* 1. Delete the following author from the database.

Carlos Coronel

Note: If you successfully deleted this author from the database, what happened to all the books this author might have written and exist in the database? You can issue queries to prove your answer.

**Answer**: I have a foreign Key constraint that prevents an Author from being deleted if the Author is tied to an A\_ID as that is a key used for another table.

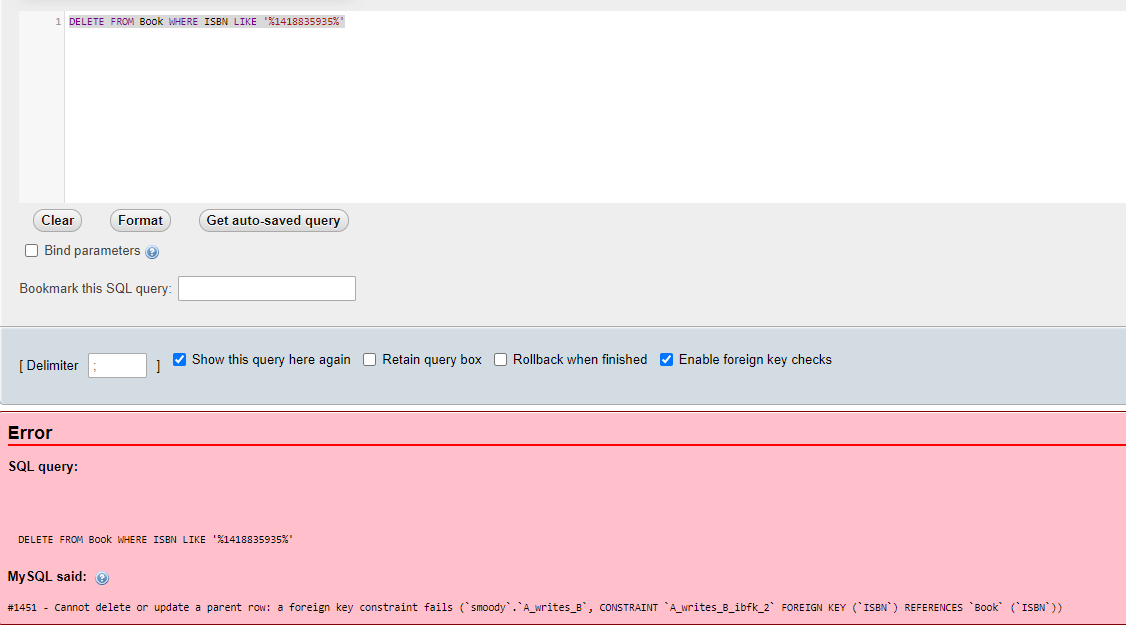


* 1. Delete the book with the following ISBN from the database.

1418835935

Note: Similarly, if you successfully deleted this book from the database, what happened to the information in A\_Writes\_B table that this book were related to? You can issue queries to prove your answer.

**Answer**: Similarly to above, I have a foreign key constrain due to the fact that ISBN is used as a Key for another Table.



* 1. Give the number of publications for each year. e.g.,

Year Number of Publications

2002 1

2004 2

. . . . . . . . . . . . . . .

Note: For this query, you need to use “COUNT( )” and “GROUP BY” in a “SELECT”

statement.

**Query Used:** “ SELECT COUNT(Title), Year FROM Book GROUP BY Year ORDER BY COUNT(Year) ASC ”

