Assumptions:

- **BNO** are unique book identifiers.
- **PNO** are unique publisher identifiers.
- The date in the *books* relation represents the year the book was originally issued. The date in the *publishes* table represents the copyright date when that particular publisher issued it.
Problems

1. Give the SQL statements to create all tables in this database. Remember to set all integrity constraints, including primary and foreign keys.

2. A CS student screwed up entering all the books’ editions (thinking 0 stood for the first edition when normal people use a 1 to indicate the first). Increase every book edition by 1 using a single SQL statement.

3. ** Delete all books that have never been published (do not have an entry in the `publishes` relation).
   ```sql
   DELETE FROM books WHERE BNO NOT IN
   (SELECT BNO FROM publishes);
   ```

4. Retrieve all distinct book titles and authors that published in New York.
   ```sql
   SELECT DISTINCT Title, DISTINCT Author
   FROM books NATURAL JOIN publishes NATURAL JOIN publishers
   WHERE Publisher='New York';
   ```

5. Retrieve the titles, publishers, and copyright dates of all first edition books using subqueries only and no joins.
   ```sql
   SELECT Title, Publisher, Copyright FROM books, publishes, publishers
   WHERE books.BNO=publishes.BNO AND publishers.PNO=publishes.PNO AND BNO IN (SELECT BNO FROM books
   WHERE edition=1);
   ```

6. Per each copyright year, find the maximum and minimum number of pages published.
   ```sql
   SELECT Copyright,min(Pages),max(Pages) FROM publishes GROUP BY Copyright;
   ```

7. What year did Adam Smith publish his first book?

8. ** Who published Adam Smith’s most recent book?
   ```sql
   SELECT Copyright,min(Pages),max(Pages) FROM publishes GROUP BY Copyright;
   ```

9. As mentioned in the lectures, outer join expressions in SQL (and relational algebra) are short-hands for more involved queries. Rewrite the following statement without using the outer join expression.

   ```sql
   SELECT * FROM books natural left outer join publishes;
   Solution:
   select * from books natural join publishes
   union
   select books.BNO,Title,Author,Date,Edition,NULL,NULL,NULL
   from books, publishes
   where NOT EXISTS (select * from books where books.BNO = publishes.BNO);
   ```