

Hand in a hardcopy of your answers (printed or handwritten). Include your name, and be sure to clearly identify which problems you are answering and which homework the problems are from since problems from multiple homeworks will be collected at the same time. These problems will be collected on Monday 10/26.

*You may use MySQL Workbench to create/modify tables instead of writing CREATE TABLE statements from scratch. In that case, run the statement SHOW CREATE TABLE *tablename* in a query tab after you've created the table to get the CREATE TABLE statement.*

- *Use the `username_library` database (where `username` is your username) and connect to the database with your individual MySQL account (rather than `guest`) so that you have permission to create and edit tables. Your MySQL username and password will be handed out.*
- *See the “data definition in MySQL Workbench” link on Canvas for instructions on using Workbench for data definition.*

You may also use the MySQL database to develop and test your LOAD DATA INFILE statements.

- *Data files for #3 are available at <https://math.hws.edu/bridgeman/courses/343/f26/hw/library> for testing purposes.*
 - *You'll need to configure MySQL Workbench to allow LOAD DATA INFILE — see the “LOAD DATA INFILE workaround” link on Canvas.*
-

Use the following relational schema for questions 1–3:

```
BOOK(Book_id, Title, Publisher_name)
BOOK_AUTHORS(Book_id, Author_name)
BOOK_COPIES(Book_id, Branch_id, No_of_copies)
BOOK_LOANS(Loan_id, Book_id, Branch_id, Card_no, Date_out, Due_date,
           Returned_date)
BORROWER(Card_no, Name, Address, Phone)
LIBRARY_BRANCH(Branch_id, Branch_name, Address)
PUBLISHER(Name, Address, Phone)
```

- Identifiers (book, branch, and loan IDs; card numbers) are positive whole numbers. Loan IDs are generated automatically.
- Titles can be up to 100 characters long.
- Names (of authors, publishers, branches, borrowers) can be up to 50 characters long.
- Addresses (for publishers, branches, and borrowers) can be up to 100 characters long.
- Phone numbers are always exactly 12 characters.

The foreign keys: ($\rightarrow$ reads as “refers to”)

- BOOK.Publisher_name $\rightarrow$ PUBLISHER.Name
- BOOK_AUTHORS.Book_id $\rightarrow$ BOOK.Book_id
- BOOK_COPIES.Book_id $\rightarrow$ BOOK.Book_id
- BOOK_COPIES.Branch_id $\rightarrow$ LIBRARY_BRANCH.Branch_id
- BOOK_LOANS.Book_id $\rightarrow$ BOOK.Book_id
- BOOK_LOANS.Branch_id $\rightarrow$ LIBRARY_BRANCH.Branch_id
- BOOK_LOANS.Card_no $\rightarrow$ BORROWER.Card_no

1. Give `CREATE TABLE` statements for the following tables, including the primary key and foreign key constraints indicated. Restrict deletion but cascade updates. List the statements in a valid order of execution.
 - BOOK
 - BOOK_LOANS
 - BORROWER
 - LIBRARY_BRANCH
 - PUBLISHER
2. For each of the following constraints, identify how to implement it in the database (domain, default value, `NOT NULL`, `UNIQUE`, `CHECK` constraint, primary key, foreign key) and give the relevant part(s) of the `CREATE TABLE` statement(s). (You don't need to give the full `CREATE TABLE`.) If it is not possible to express the constraint using any of these mechanisms, explain why.

Treat each constraint separately; they are not cumulative.

- (a) By default, the number of copies of a book at a branch is 1.
- (b) By default, the check out date for a book is the date on which the `BOOK_LOANS` entry is recorded and the book is due two weeks after it is checked out.

`CURDATE()` returns the current date as a `DATE` value (no time of day), e.g., `SELECT CURDATE();` might return `2026-10-11`.

`DATE_ADD(date, INTERVAL n unit)` returns the date *n* units after *date*, where *unit* is `DAY`, `WEEK`, `MONTH`, `YEAR`, etc. The *date* argument can be any expression of type `DATE`. For example:

- `DATE_ADD('2026-10-11', INTERVAL 3 DAY)` returns `2026-10-14`
- `DATE_ADD(CURDATE(), INTERVAL 1 MONTH)` returns the date one month from today

- (c) The number of copies can't be negative.
- (d) Every book must have a title, and every loan must have a check out date and a due date.
- (e) Loan dates record a calendar day only, not a time of day, and must be valid dates (e.g., not February 30).
- (f) The combination of name and phone number also uniquely identifies particular borrowers.
- (g) Phone numbers must be of the form `XXX-XXX-XXXX`.
- (h) Neither the due date nor the return date can be before the date the book was checked out.

- (i) The same loan can't be recorded twice: there can't be two loan records for the same book, branch, and borrower with the same check out date.
 - (j) A book can only be loaned out from a branch that holds that book (i.e., that has a `BOOK_COPIES` entry for it).
 - (k) If a publisher is removed from the database, its books remain but no longer list a publisher.
 - (l) Deleting a book also removes its author and copy records, but a book that appears in any loan record cannot be deleted.
 - (m) A particular borrower can't have more than 5 books checked out at a time.
 - (n) A book's check out date cannot be in the future.
 - (o) Once a loan is recorded, its check out date cannot be changed.
3. (a) Write `LOAD DATA INFILE` statements to load the following data files. The file contents are shown for reference — see the instructions at the beginning of this handout to download files if you want to test your `LOAD DATA INFILE` statements.
- (i) `PUBLISHER.csv`

```
Name,Address,Phone
DAW,"375 Hudson St, New York, NY 10014",123-456-7890
Eos,"10 East 53rd St, New York, NY 10022",212-207-7000
```
 - (ii) `BOOK.csv`

```
Book_id,Title,Publisher_name
1,Foreigner,DAW
2,Invader,DAW
3,Mission Child,Eos
4,Fortress in the Eye of Time,Eos
```
 - (iii) `BOOK_LOANS.csv`

```
Book_id,Branch_id,Card_no,Date_out,Due_date
1,2,5,2006-08-05,2006-11-05
2,2,3,2006-08-06,2006-10-06
4,6,1,2006-10-07,2006-10-17
2,7,5,2006-09-08,2006-10-18
4,8,2,2006-10-05,2006-11-05
1,8,6,2006-10-16,2006-11-06
1,14,6,2006-11-07,2006-11-17
1,14,3,2006-11-08,2006-11-18
```

(iv) PUBLISHER.txt — the columns are separated by tabs

DAW	375 Hudson St, New York, NY 10014	123-456-7890
Eos	10 East 53rd St, New York, NY 10022	212-207-7000

(b) Give a legal order for loading data into all seven of the library tables. You don't need to write the full `LOAD DATA INFILE` statements, just list the table names.

Use the following schema for question 4:

```
EMPLOYEE(fname,minit,lname,ssn,bdate,address,sex,salary,superssn,  
         deptnum)  
DEPARTMENT(deptname,deptnum,mgrssn,mgrstartdate)  
DEPT_LOCATIONS(deptnum,deptloc)  
PROJECT(projname,projnum,projloc,deptnum)  
WORKS_ON(empssn,projnum,hours)  
DEPENDENT(empssn,fname,sex,bdate,relationship)
```

4. For each of the following constraints, identify how to implement it in the database (domain, default value, `NOT NULL`, `UNIQUE`, `CHECK` constraint, primary key, foreign key). You don't need to give actual SQL, but be specific — don't just say "domain", for example, but indicate which column and the specific data type. If it is not possible to express the constraint using any of these mechanisms, explain why.

Treat each constraint separately; they are not cumulative.

- (a) Employees can only work on projects belonging to the department they work for.
- (b) Managers do not have supervisors.
- (c) Managers must work for the department they manage.
- (d) Hours worked on a project are reported to the tenth of an hour.