MLDS-413 Introduction to Databases and Information Retrieval

Homework 2: Data modeling: Data Sets, Normalization, and ER Diagrams

Name 1: ____

NetID 1: ____

Name 2: ____

NetID 2: ____

Instructions

You should submit this homework assignment via Canvas. Acceptable formats are word files, text files, and pdf files. Paper submissions are not allowed and they will receive an automatic zero.

As explained during lecture and in the syllabus, assignments are done in groups. The groups have been created and assigned. Each group needs to submit only one assignment (i.e., there is no need for both partners to submit individually the same homework assignment).

Each group can submit solutions multiple times (for example, you may discover an error in your earlier submission and choose to submit a new solution set). We will grade only the last submission and ignore earlier ones.

Make sure you submit your solutions before the deadline. The policies governing academic integrity, tardiness and penalties are detailed in the syllabus.

Question 1. Dataset Exploration (6 points)

Download the data sets using the links below and import them to either Excel, Numbers, or another spreadsheet processing program of your choice. Below you can find a tutorial on how to download a data set and import it to Excel or Numbers. Then, proceed to answer the following questions:

- **a.** **(3 points)** Did you encounter any problems in importing any of these datasets into a spreadsheet? If yes, describe which dataset(s) you encountered the problem with, and explain the reasons you believe it failed to be imported.
- **b.** For the dataset(s) that were successfully imported, please answer the following questions:
  - **i.** **(1 point)** What is the data set's name? You have to be really precise with the name; after all, there may be multiple datasets at Kaggle.com with similar names. Find the name that uniquely identifies it.
  - **ii.** **(1 points)** How many rows does it have?
  - **iii.** **(1 points)** How many columns does it have?

Link to Project in Kaggle	CSV file to download
https://www.kaggle.com/benhamner/sf-bay-area-bike-share	status.csv
https://www.kaggle.com/abcsds/pokemon	pokemon.csv

How to download a data set from Kaggle

The links above point to two data predictive modelling and analytics projects on Kaggle. You can read the project and data description on the corresponding project overview page. If you click the project data page, you will see a list of comma-separated values (CSV) files and other file types on the left hand side. When you click on any file, you can preview the first 100 columns, and read the column metadata or column metrics. There is a download button to download that CSV file to your computer. Once downloading is done, you can follow the guidelines below to import the dataset into the spreadsheet program of your choice.

How to import a CSV file into Excel

Go to the "File" main menu and select the submenu "import" to import a csv file in Excel. Choose the appropriate file type to import the file (CSV), and then click the import button. Next, select which file to import, and click "Get Data". Most of the CSV files are delimited by commas to separate each column, and you can import it starting at row *n* if that file is too large. After that, there are a few options to opt for a proper column data format, and a spreadsheet which you want to put the data. It is recommended import the new data into a new sheet. When you are done with all file configurations, click "Finish". Excel will try to import the CSV file, or throw an exception if any error happens.

How to open a CSV file in Number

For Numbers, you can choose which CSV file to open, and the program will do it for you. That's it.

Solution

1. **Status.csv**
   - **a.** **The table has 71,984,435 rows. Excel can handle only up to 1,048,576 rows x 16,384 columns on a worksheet. This table is too large for a spreadsheet program to load. Consequently, we need to use databases in order to store, access and manipulate big data. Note: students do not have to find the exact number of columns; answering "*the table is too large*" will get a full score.**
   - **b. i.** **(answering this is optional) Dataset Name: "*SF Bay Area Bike Share*", Table Name: "*status*"**
   - **b. ii.** **(answering this is optional) 71,984,435 rows**
   - **b. iii.** **(answering this is optional) 4 columns**
2. **Pokemon.csv**
   - **a.** **The Excel can load this table without any problem**
   - **b. i.** **Dataset Name: "*Pokemon with stats*", Table Name: "*pokemon*"**
   - **b. ii.** **801 rows including the header**
   - **b. iii.** **13 columns**

Question 2. Data Type Exploration (6 points)

Assume the datasets provided below. Consider the following data formats:

- **a. (1 point)** 32-bit integer
- **b. (1 point)** 64-bit integer
- **c. (1 point)** fixed point (and specify the number of decimal places)
- **d. (1 point)** floating point (either single or double precision)
- **e. (1 point)** date and time in epoch seconds
- **f. (1 point)** date and time in epoch microseconds

For each one of the data formats above, please answer the following:

- i. Is there a column in one of these datasets that would be **best** stored in that format? (yes/no)
- ii. If yes, please provide
  1. the data set
  2. the table name
  3. the column name
  4. a one-sentence description of the column
  5. an example of the data in the column
  6. the reason why your chosen data type is appropriate
- iii. If no, explain why not (1-2 sentences)

Data sets:

1. https://www.kaggle.com/benhamner/sf-bay-area-bike-share
2. https://www.kaggle.com/datasf/san-francisco

Solution

There are many possible solutions to the question above. This solution provides one such set.

- **a. 32-bit integer**
  - **i. Yes**
  - **ii. 1. San Francisco Open Data**
  - **ii. 2. Bikeshare_stations**
  - **ii. 3. Dockcount**
  - **ii. 4. Number of total docks at station**
  - **ii. 5. 11**
  - **ii. 6. The maximum number of dock count is only 35 from the preview data, and it should not be more than the maximum value of the 32-bit integer.**

- **b. 64-bit integer**
  - **i. Yes**
  - **ii. 1. San Francisco Open Data**
  - **ii. 2. Sfpd_incidents**
  - **ii. 3. pdid**
  - **ii. 4. Unique Identifier for use in update and insert operations**
  - **ii. 5. 12008803126080**
  - **ii. 6. Most of the numbers in this column require a 64-bit integer, and the 32-bit integer is not enough to represent the data. Moreover, this is a unique identifier of an operation; thus, it needs more bits to identify and distinguish a row.**

c. Fixed Point (1 decimal point)

i. Yes

ii. 1. SF Bay Area Bike Share

ii. 2. weather

ii. 3. max_temperature_f

ii. 4. There is no description of the data (as is the case in many datasets). Most likely, each data point represents the maximum temperature in degrees Fahrenheit on that day.

ii. 5. 74.0

ii. 6. The decimal point of most data points is zero, and this column requires only one decimal point. Fixed point is the most space-efficient representation with sufficient accuracy.

d. Floating Point

i. Yes

ii. 1. San Francisco Open Data

ii. 2. Sfpd_incidents

ii. 3. latitude

ii. 4. The latitude of the incident

ii. 5. 37.7803522156893

ii. 6. The latitude data is very accurate, and in general it requires 10 decimal points. For this reason, the floating point is more suitable than the fixed point.

e. date and time in epoch seconds

i. Yes

ii. 1. SF Bay Area Bike Share

ii. 2. trip

ii. 3. end_date

ii. 4. the end date of the trip

ii. 5. 8/29/2013 9:11

ii. 6. This data contains date and time in seconds of the end date of the trip. Without the microsecond, using this data type can save more space/memory space.

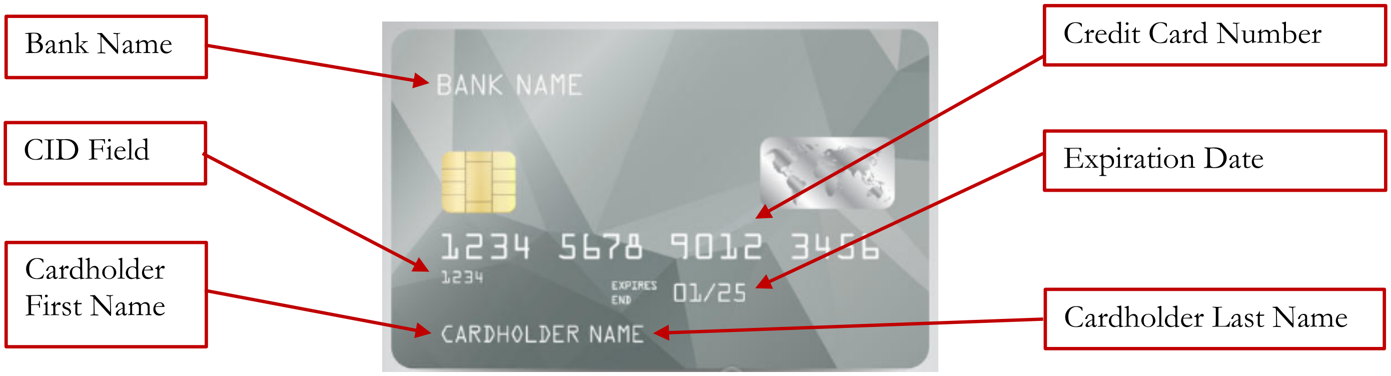
f. date and time in epoch microseconds

i. No

iii. Among all columns of these two datasets, the columns that include time require a maximum resolution of seconds. Thus, these columns would be better serviced by type (e). There is no column that requires the high resolution of microseconds (f). Using (f) for the existing columns that include time would be suboptimal, as microseconds (f) require 64-bits to allow for counter increments sufficiently far into the future (the microseconds since 1970 already need 51 bits to be represented), while seconds (e) with 32-bit integers can run for another 20 years before the counter wraps around.

Note aside: Remember the Y2K problem? Well, we have a similar problem ahead of us. Many systems store the epoch in seconds in signed 32-bit integers. Such implementations cannot encode times after 03:14:07 UTC on 19 January 2038. The next second after that time overflows the 32-bit integer, resulting in a negative number. This is called the "Year 2038 problem".

Question 3. Data Types (8 points)



You are building a database for a credit card company. You need to select the best data types for the various parts of a credit card shown above. The database software you use supports the following types:

- *32-bit signed integer*: can store all integers between -2,147,483,648 and 2,147,483,647.
- *32-bit floating point*: can store numbers in the scientific notation with about 7 decimal digits of precision and exponent between 10^{-38} and 10^{+38}. It can precisely store all integers $\leq$ 16,777,215.
- *Epoch seconds*: a 32-bit unsigned integer that represents data and time by the number of seconds since midnight on Jan 1, 1970 in London.
- *UTF-8*: text, of any length.

You must use one of these four data types to store each of the values below. Each value should be stored in a single data element. Which of these types is the best to store:

a. **(1 point)** The bank name?
Answer: UTF-8

b. **(1 point)** The CID field (4-digit decimal number)?
Answer: 32-bit integer

c. **(1 point)** The cardholder last name?
Answer: UTF-8

d. **(1 point)** The expiration date?
Answer: Epoch seconds

e. **(2 points)** The credit card number (16-digit decimal number)?
Answer: UTF-8 (32-bit integers max out at 10 digits, they cannot represent 16 digits)

f. **(2 points)** The balance of the credit card?
NOTE: the balance is in USD, it is guaranteed to stay between -1,000,000.00 and +1,000,000.00 (negative values indicate credit), and must be accurate down to one cent (i.e., 1/100th of a dollar).
Answer: 32-bit integer showing the balance in cents (this is basically a fixed-point type)

Question 4. Database Normalization (10 points)

You work as a data analyst at a fishing/outdoors company. To identify new talents in database design, the company hosts an annual database schema competition. The winner takes home a commemorative statue known as the *Data Bass*. You won the competition last year, so a friend asked you to review his submission. Unfortunately, your friend did not take MLDS-413 and put all his data in a single table, shown below:

Employees_Database					
Empl. ID	*Name*	*Favorite Record*	*Department*	*Dept. Manager ID*	*Fav. Record's Artist*
1	Nancy	Abbey Road	Sales	1	The Beatles
2	John	Porgy and Bess	Accounting	2	Gershwin
3	Bill	Kind of Blue	Operations	6	Miles Davis
4	Tracy	A Night At The Opera	Sales	1	Queen
5	Muji	La Revancha del Tango	Sales	1	Gotan Project
6	Ohana	Ka 'Ano'i	Operations	6	Kamakawiwo'ole
7	Jill	Porgy and Bess	Accounting	2	Gershwin
8	Gloria	La Revancha del Tango	Operations	6	Gotan Project
9	Frank	Abbey Road	Accounting	2	The Beatles

a. **(6 points)** Help him by normalizing the database to remove redundancy. Show the normalized database **schema**.

Solution

Employees	
Employee ID	**integer**
Name	**varchar(256)**
Favorite Record ID	**integer**
Department ID	**integer**

Departments	
Department ID	**integer**
Name	**varchar(256)**
Manager ID	**integer**

Records	
Record ID	**integer**
Record Name	**varchar(256)**
Artist Name	**varchar(256)**

b. (4 points) Show the current **instance** of the database in the normalized schema.

Solution

Employees			
Employee ID	*Name*	*Favorite Record ID*	*Department ID*
1	Nancy	1	1
2	John	2	2
3	Bill	3	3
4	Tracy	4	1
5	Muji	5	1
6	Ohana	6	3
7	Jill	2	2
8	Gloria	5	3
9	Frank	1	2

Departments		
Department ID	*Name*	*Manager ID*
1	Sales	1
2	Accounting	2
3	Operations	6

Records		
Record ID	*Record Name*	*Artist Name*
1	Abbey Road	The Beatles
2	Porgy and Bess	Gershwin
3	Kind of Blue	Miles Davis
4	A Night At The Opera	Queen
5	La Revancha del Tango	Gotan Project
6	Ka 'Ano'i	Kamakawiwo'ole

Question 5. ER Diagram (20 points)

The main entities that participate in an online bookstore enterprise are as follows:

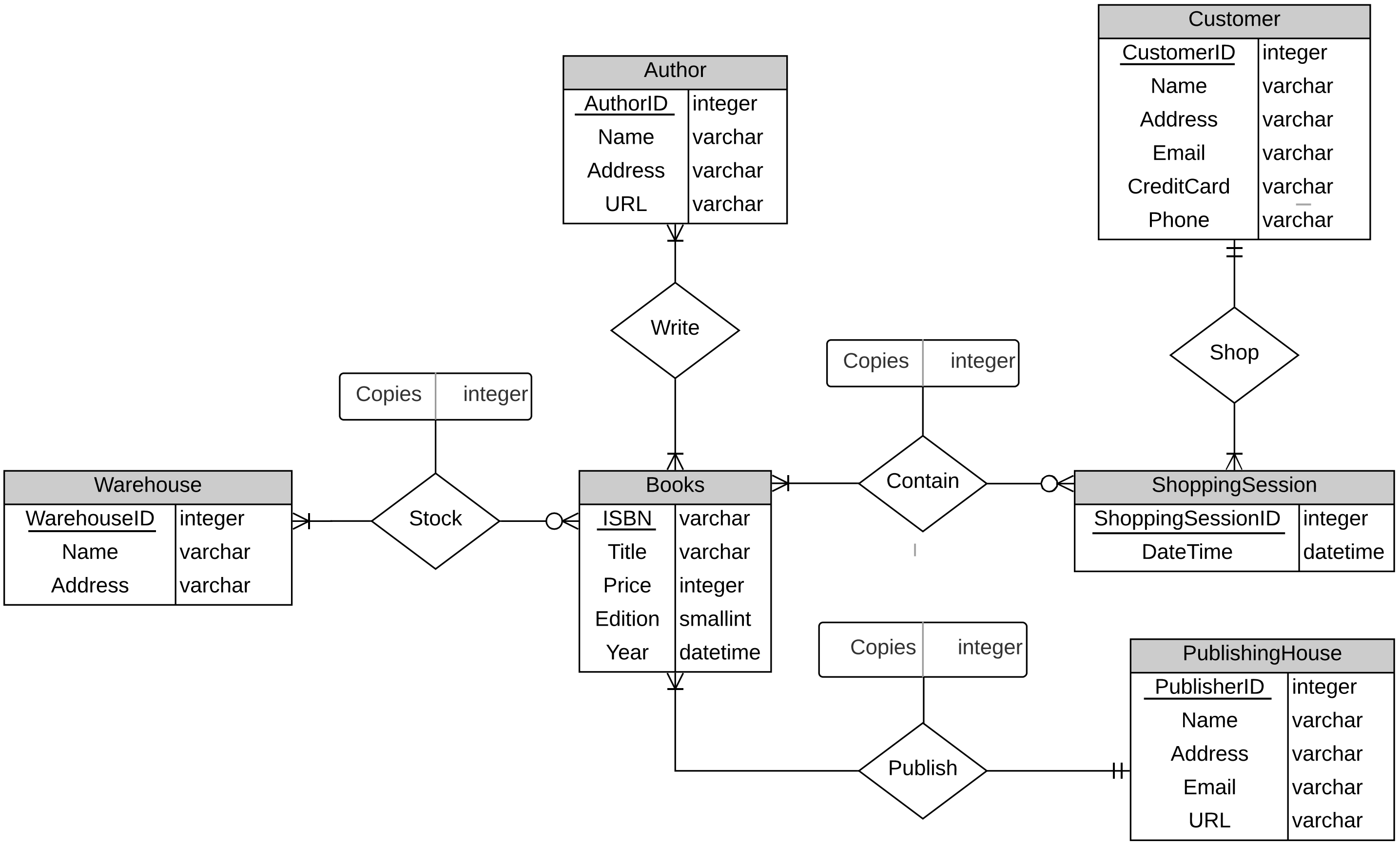
- A book has the information about the year that it was published, its title, the (current) price and its ISBN number. **Assume that ISBN is the unique number assigned to each edition/version of the book**.
- An author has information which includes his/her name and contact-address, along with a URL.
- Each publishing house/company has a name, postal address, phone number, email and URL.
- Each customer has a customer ID, name, address, email, credit card, and phone number, and **each customer must provide only one set of information**.
- A particular "shopping session" is typically recorded as a shopping basket, which is assigned a unique basket ID and has the information about the date of the given shopping session.
- Since this is an online bookstore, there must be physical locations where (copies of) the books are stored. A given warehouse has its address, name, and phone number available.

The associations among the various entities listed above are as follows:

- Each book is written by some author(s).
- Each book is published by a particular publishing house and information is kept about the publishing date, the edition number, and the number of copies.
- Each shopping basket is associated with a particular customer.
- Each shopping basket may contain several books and even several copies of a particular book.
- Each warehouse keeps/stocks different books, and for each book it also records the number of copies that it currently has.

Please draw an ER Diagram that models the online bookstore according to the information and rules provided above. If you make any assumptions along the way, please write them down. Note: there are a number of online tools to draw ER diagrams (e.g., https://www.lucidchart.com).

Solution



The following assumptions are made in drawing this diagram:

- **ISBN is unique for each version/edition of the book.**
- **The *contain* relation indicates which books and how many of them are in the shopping basket. The table of this relation has both the *shopping_session_ID* and *ISBN* as a composite key.**
- **The *shop* relation is a result of a customer doing a shopping session. The relation table contains *shopping_session_ID* and *customer_ID* as a composite key and the number of copies of a book.**

Relationship Descriptions and Assumptions:

- **A book is stocked in one or many warehouses, and a warehouse stocks zero, one, or many books.**
- **A publishing house publishes one or many books, but a book must be published by one and only one publishing house.**
- **A customer shops one or many shopping sessions, and one shopping session has one and only one customer.**
- **A shopping session contains one or many books, and a book is contained by zero, one or many shopping sessions. Note that there are no rules or constraints that a shopping session cannot have zero books, so zero, one, or many participation is also fine.**
- **An author writes one or many books, and a book is written by one or many authors.**

Note that there are many ways to draw the ER diagram of these entities and relationships, and a lot depends on additional assumptions made by the database designer. This solution shown here is only one of many possible solutions.