Exercise

1. We assume that there are no data in all the three tables. So the primary id of the following tables start from 1.
2. ALTER command (you are free to use either GUI or SQL command):
    - Add a new column in users table with following properties: registration_timestamp (type: timestamp, default: current_timestamp)
    - Add a new column in products table with following properties: purchase_timestamp (type: timestamp, default: current_timestamp)

    NOTE:

    - When we assign a default value to a column, then that default value is stored if none other value is forced to be stored using INSERT/UPDATE. When default of a timestamp column is CURRENT_TIMESTAMP, it enters the date & time of that very instance when the record/row was being created. Thus, registration_timestamp in users table will now store the date & time whenever a particular user registers. Similarly, in users_products table we will now store exactly when was it when a user purchased a particular product.

3. Now, we will insert records of all the users who registered, all the products that we sell, and order history of every user who has made at least one purchase in the past. INSERT the following records in users table, products table, and users_products table:

    users table:
    data below is in the following sequence:
    email, first_name, last_name, phone, registration_timestamp
    Data:'unknown' ('venu@xyz.com', 'Venu', 'Sharma', 547903927, '2016-09-28 21:30:04'),
    ('shubham@xyz.com', 'Shubha', NULL, 538915694, CURRENT_TIMESTAMP),
    ('disha@xyz.com', 'Disha', 'Kaur', 557825323, '2016-09-30 00:02:14'),
    ('ankit@xyz.com', 'Ankit', 'Kumar', 561322116, '2016-09-30 15:05:18'),
    ('mrinal@xyz.com', 'Mrinal', 'Joy', 517918670, '2016-10-02 09:08:06'),
    ('abhilash@xyz.com', 'Abhilash', 'Jalsani', 509841902, '2016-10-01 10:30:00'),
    ('hardik@xyz.com', 'Hardik', 'Arora', 595452568, '2016-09-30 13:20:45'),
    ('yesha@xyz.com', 'Yesha', 'Krishna', 534532216, '2016-09-30 13:20:45'),
    ('rushit@xyz.com', 'Rushit', NULL, 534359370, '2016-09-29 11:46:37'),
    ('jessy@xyz.com', 'Jessy', 'Joseph', 591053100, '2016-09-29 00:02:14'),
    ('jasper@xyz.com', 'Jaspreet', NULL, 515078235, '2016-09-29 23:50:12'),
    ('prachi@xyz.com', 'Prachi', NULL, 530670640, '2016-09-29 12:12:12');

    NOTE:

    - Registration timestamps are shared for each user. Please notice yourself that for Shubham (2nd user on the list above), the argument CURRENT_TIMESTAMP will store

the time of running the INSERT query. We recommend you to cross-check this fact by yourself.

- Few entries are NULL, that is. The data for these entries was not provided.
- You do not have to take care of the id column explicitly, provided it is already defined as primary key and with auto-increment.

products table:
data below is in the following sequence:
(name, category)
Data: ('harry_potter', 'book'),
('deception_point', 'book'),
('drunkards_walk', 'book'),
('the_kite_runner', 'book'),
('animal_farm', 'book'),
('night_lamp', 'other'),
('pencil_box', 'stationery'),
('fountain_pen', 'stationery'),
('study_bed_table', 'other'),
('personal_diary', 'stationery');

NOTE:

- You do not have to take care of the id column explicitly, provided it is already defined as primary key and with auto-increment.

users_products table:
data below is in the following sequence:
(user_id, product_id, purchase_timestamp)
Data:
(12, 3, '2016-10-10 18:10:12'),
(12, 6, '2016-10-12 18:10:12'),
(8, 6, '2016-10-29 12:12:12'),
(4, 6, '2016-10-15 13:05:04'),
(10, 9, '2016-10-10 02:15:16'),
(7, 4, '2016-10-10 08:34:42'),
(5, 1, '2016-10-19 16:58:54'),
(4, 5, '2016-10-14 11:35:32'),
(8, 5, '2016-10-23 56:45:23'),
(7, 7, '2016-10-04 13:12:35'),
(12, 2, '2016-10-14 12:23:41'),
(12, 7, '2016-10-21 14:15:13'),
(7, 2, '2016-10-05 16:08:02'),
(7, 8, '2016-10-06 17:34:49'),
(1, 2, '2016-10-13 15:37:51'),
(11, 6, '2016-10-15 10:03:01');

Now we have the order history of every user who has made at least one purchase.

NOTE:

- Purchase timestamp are shared for each user.
- You do not have to take care of the id column explicitly, provided it is already defined as primary key and with auto-increment.

4. Now, you received a mail from one of your users that all the emails that he receives from us are addressed to a wrong name. His name is Shubham, but the mails are addressed to Shubha. So you figured out that probably he made a typo while typing his name during registering on your website (all the emails are addressed to the first_name column). Your task is to update users name to Shubham using UPDATE command.

   Now, you received another mail from another email from yesha@xyz.com who says that she bought a book, and just wanted to confirm whether the order was successfully placed or not. Your task is to find out whether a user with email id yesha@xyz.com has ordered anything. If yes, what.

   Next, you figured out she did purchase a book. While checking for her orders, you also figured out that there's a typo in the name of the book she has ordered (just by looking at the entry). Your task is to UPDATE the book's name to its correct name. (Search in Google if you don't know the correct spelling of that book).