EMORY UNIVERSITY

Managing Big Data

Homework #3

Due: turned in by Wed 10/11/2017 06:59 am

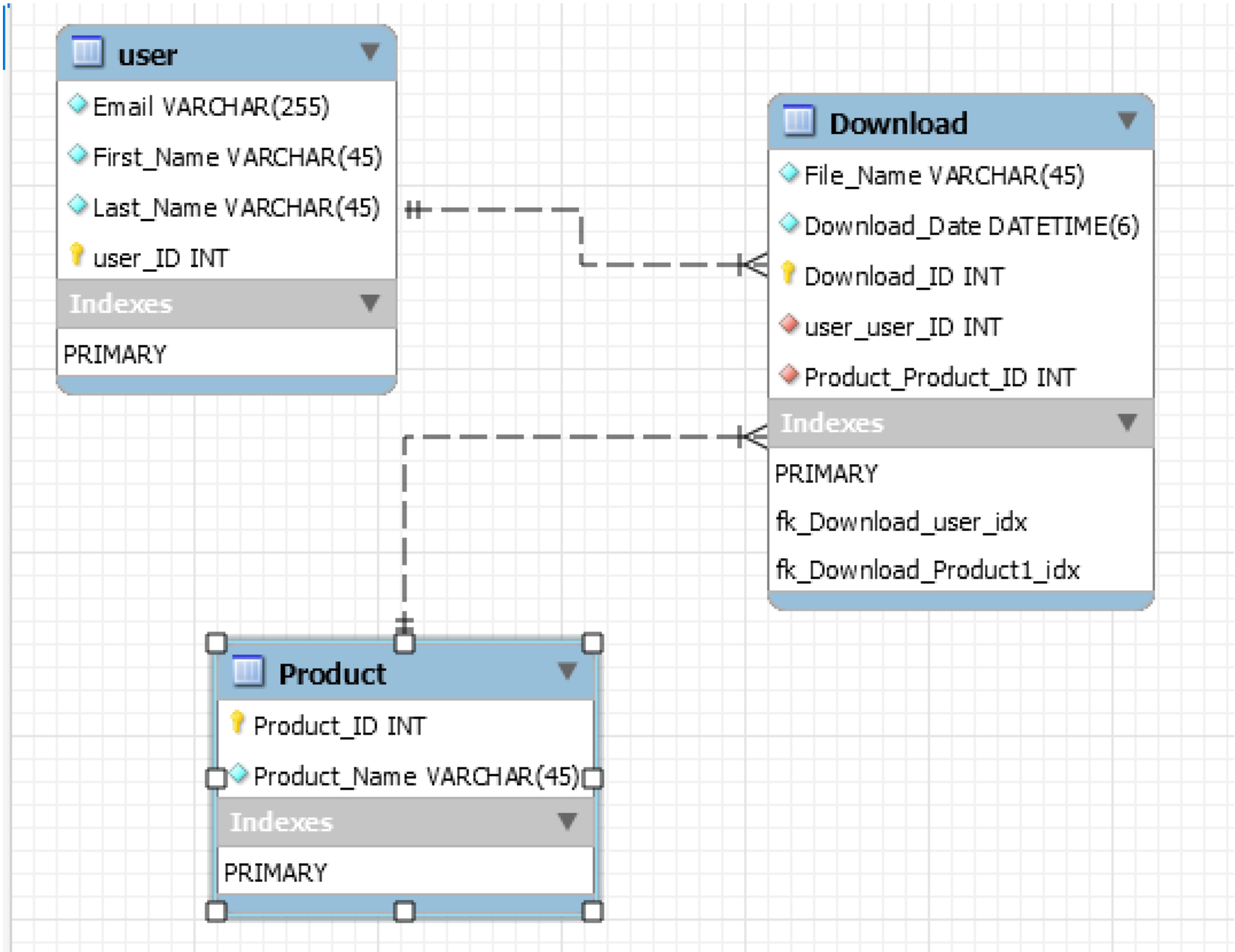
___Wenbo Liu____

(put your name above)

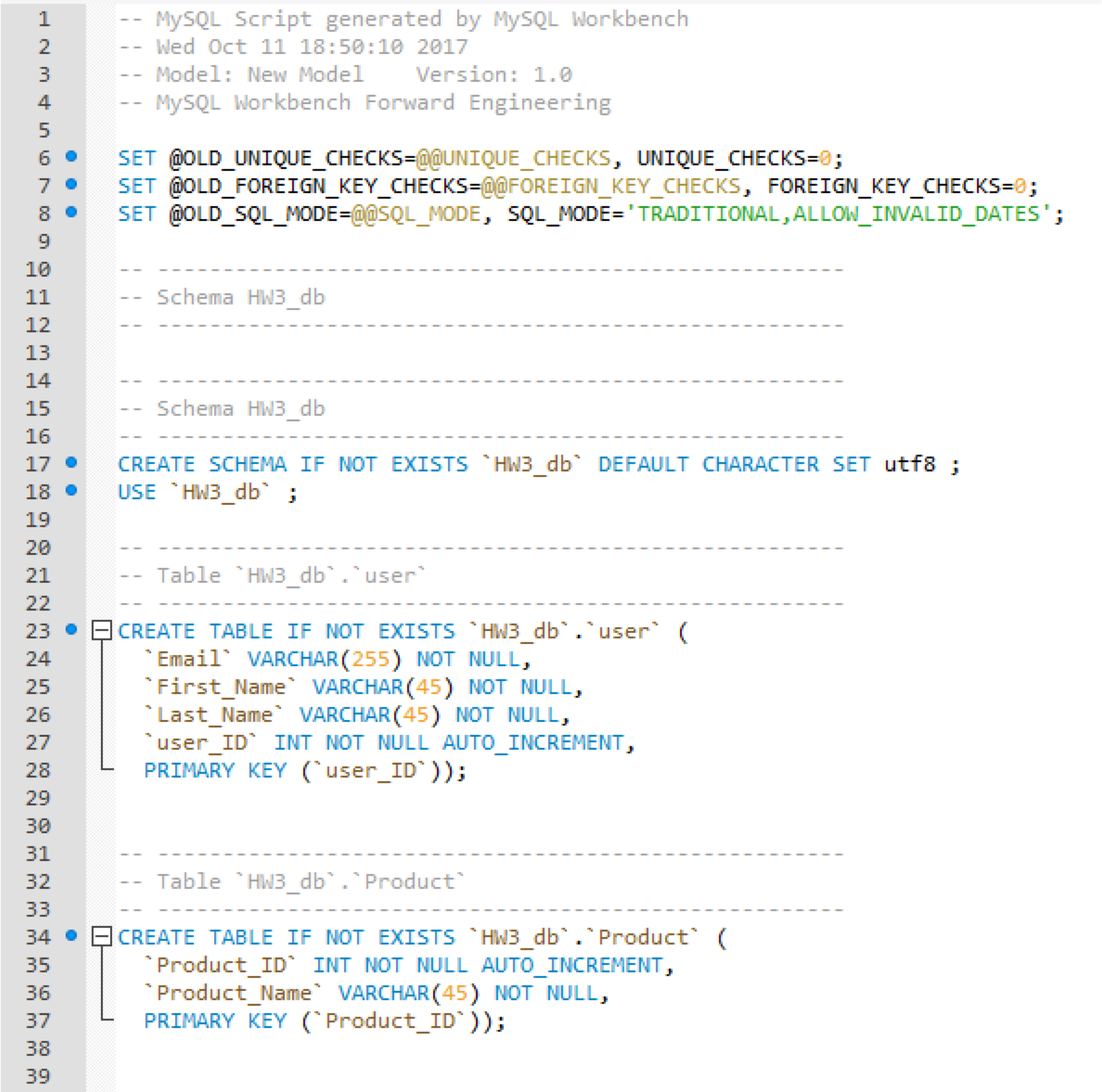
Total grade: ________ out of ___100___ points

There are 5 numbered questions. Please answer them all and submit your assignment as a single PDF or Word file by uploading it to the HW1 drop-box on the course website. You should provide: SQL statements, results of the SQL statement (typically copy first 10 rows), and answers to questions, if any.

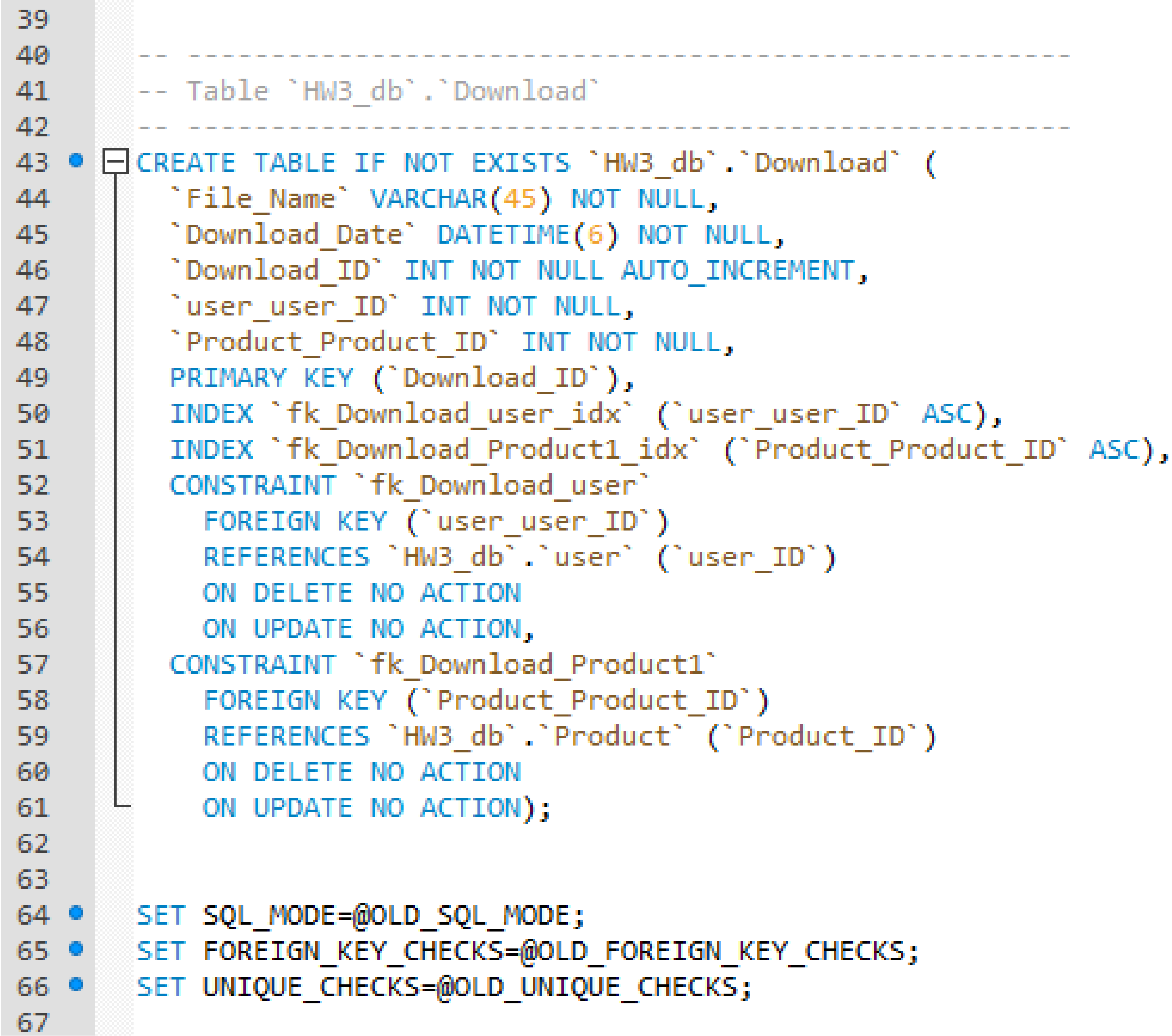
1. **Use MySQL Workbench to create an EER diagram for a database that stores information about the downloads that users make.**
   - **Each user must have an email address, first name, and last name.**
   - **Each user can have one or more downloads.**
   - **Each download must have a filename and download date/time.**
   - **Each product can be related to one or more downloads.**
   - **Each product must have a name.**



2. Use MySQL Workbench to open the EER diagram that you created in exercise 1. Then, export a script that creates the database and save this script in a file named ex3-2.sql. Next, use MySQL Workbench to open this file and review it. Report the script here.



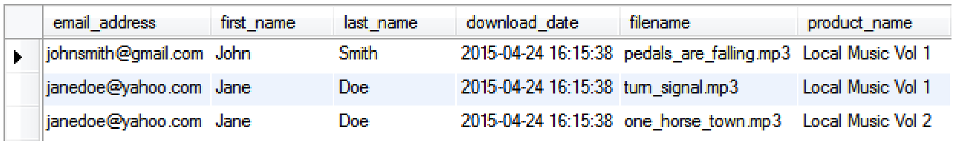
```sql
-- MySQL Script generated by MySQL Workbench
-- Wed Oct 11 18:50:10 2017
-- Model: New Model    Version: 1.0
-- MySQL Workbench Forward Engineering

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='TRADITIONAL,ALLOW_INVALID_DATES';

-- -----------------------------------------------------
-- Schema HW3_db
-- -----------------------------------------------------

-- -----------------------------------------------------
-- Schema HW3_db
-- -----------------------------------------------------
CREATE SCHEMA IF NOT EXISTS `HW3_db` DEFAULT CHARACTER SET utf8 ;
USE `HW3_db` ;

-- -----------------------------------------------------
-- Table `HW3_db`.`user`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `HW3_db`.`user` (
  `Email` VARCHAR(255) NOT NULL,
  `First_Name` VARCHAR(45) NOT NULL,
  `Last_Name` VARCHAR(45) NOT NULL,
  `user_ID` INT NOT NULL AUTO_INCREMENT,
  PRIMARY KEY (`user_ID`));


-- -----------------------------------------------------
-- Table `HW3_db`.`Product`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `HW3_db`.`Product` (
  `Product_ID` INT NOT NULL AUTO_INCREMENT,
  `Product_Name` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`Product_ID`));


```



```

-- -----------------------------------------------------
-- Table `HW3_db`.`Download`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `HW3_db`.`Download` (
  `File_Name` VARCHAR(45) NOT NULL,
  `Download_Date` DATETIME(6) NOT NULL,
  `Download_ID` INT NOT NULL AUTO_INCREMENT,
  `user_user_ID` INT NOT NULL,
  `Product_Product_ID` INT NOT NULL,
  PRIMARY KEY (`Download_ID`),
  INDEX `fk_Download_user_idx` (`user_user_ID` ASC),
  INDEX `fk_Download_Product1_idx` (`Product_Product_ID` ASC),
  CONSTRAINT `fk_Download_user`
    FOREIGN KEY (`user_user_ID`)
    REFERENCES `HW3_db`.`user` (`user_ID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Download_Product1`
    FOREIGN KEY (`Product_Product_ID`)
    REFERENCES `HW3_db`.`Product` (`Product_ID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION);


SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;

```

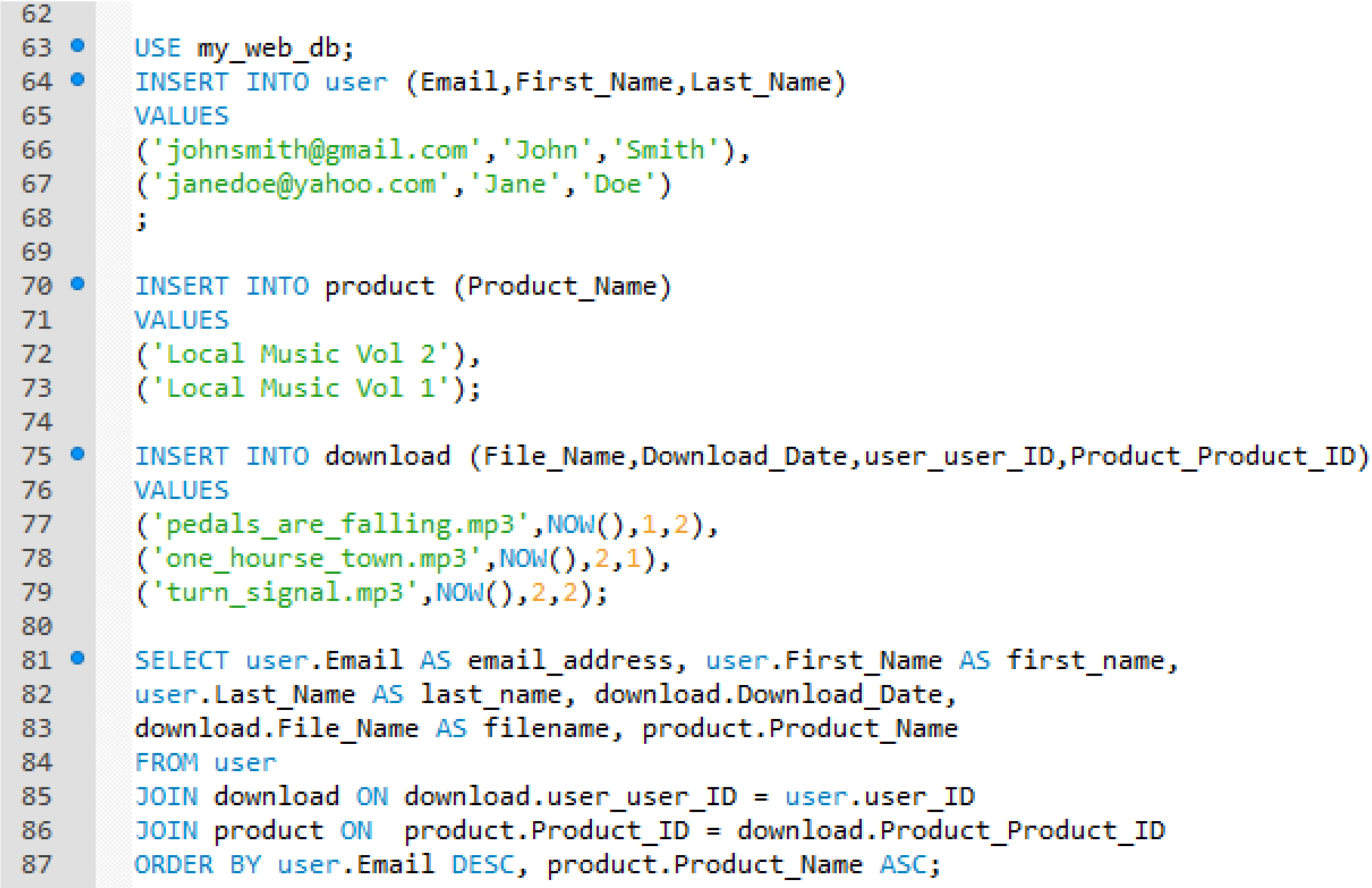
3. **Run the script you created in exercise 2 to create the database under the name my_web_db. Write a script that adds rows to the database. In particular,**
   - **Add two rows to the Users and Products tables.**
   - **Add three rows to the Downloads table:**
     - **one row for user 1 and product 2;**
     - **one row for user 2 and product 1;**
     - **and one row for user 2 and product 2.**
     - **Use the NOW function to insert the current date and time into the download_date column.**

   Write a SELECT statement that joins the three tables and retrieves the data from these tables like this:

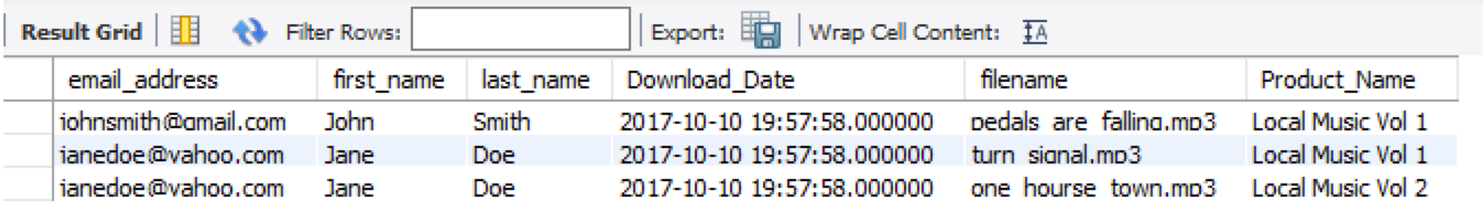


email_address	first_name	last_name	download_date	filename	product_name
johnsmith@gmail.com	John	Smith	2015-04-24 16:15:38	pedals_are_falling.mp3	Local Music Vol 1
janedoe@yahoo.com	Jane	Doe	2015-04-24 16:15:38	turn_signal.mp3	Local Music Vol 1
janedoe@yahoo.com	Jane	Doe	2015-04-24 16:15:38	one_horse_town.mp3	Local Music Vol 2

Sort the results by the email address in descending sequence and the product name in ascending sequence.

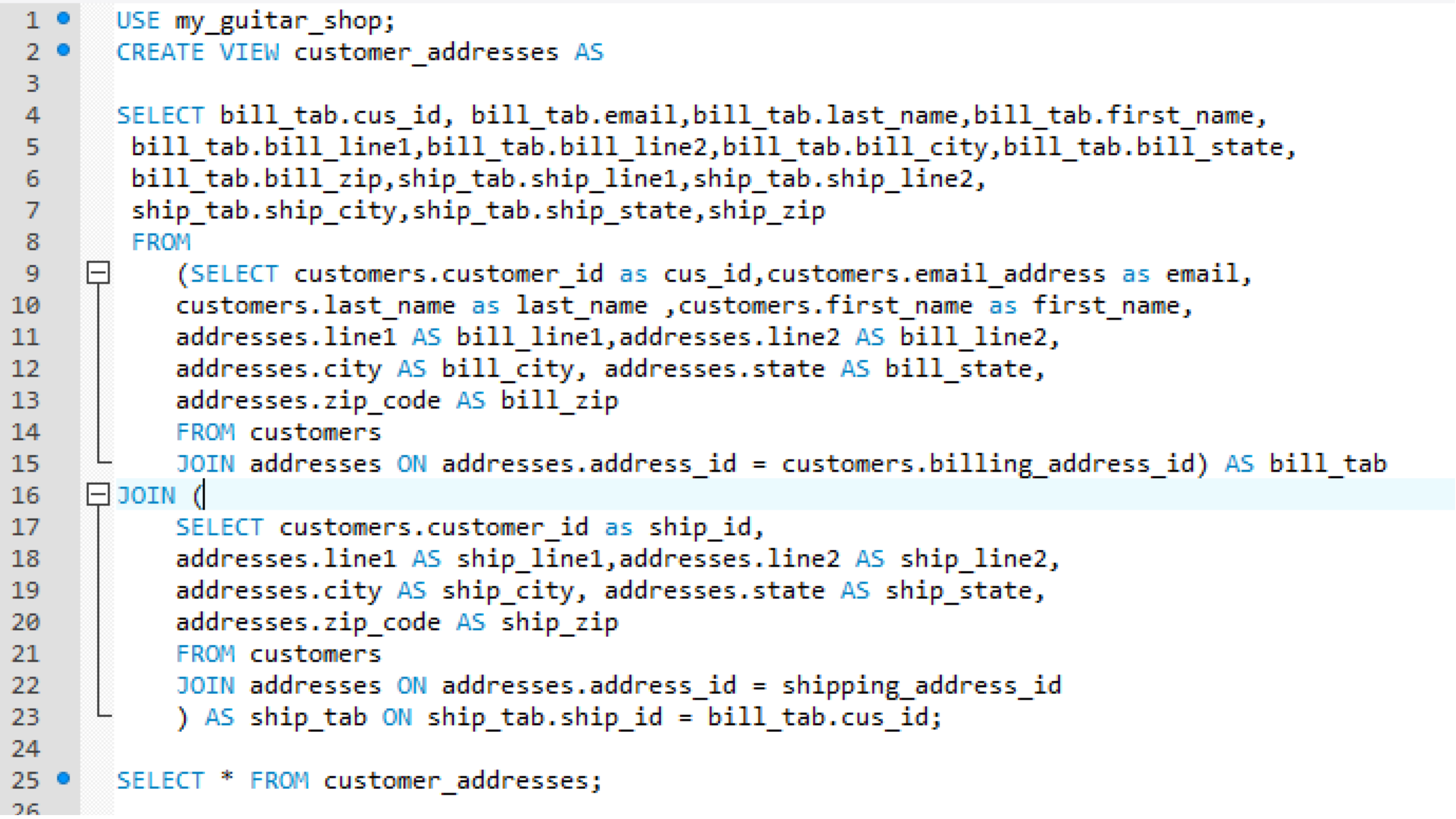


```

USE my_web_db;
INSERT INTO user (Email,First_Name,Last_Name)
VALUES
('johnsmith@gmail.com','John','Smith'),
('janedoe@yahoo.com','Jane','Doe')
;

INSERT INTO product (Product_Name)
VALUES
('Local Music Vol 2'),
('Local Music Vol 1');

INSERT INTO download (File_Name,Download_Date,user_user_ID,Product_Product_ID)
VALUES
('pedals_are_falling.mp3',NOW(),1,2),
('one_hourse_town.mp3',NOW(),2,1),
('turn_signal.mp3',NOW(),2,2);

SELECT user.Email AS email_address, user.First_Name AS first_name,
user.Last_Name AS last_name, download.Download_Date,
download.File_Name AS filename, product.Product_Name
FROM user
JOIN download ON download.user_user_ID = user.user_ID
JOIN product ON  product.Product_ID = download.Product_Product_ID
ORDER BY user.Email DESC, product.Product_Name ASC;
```

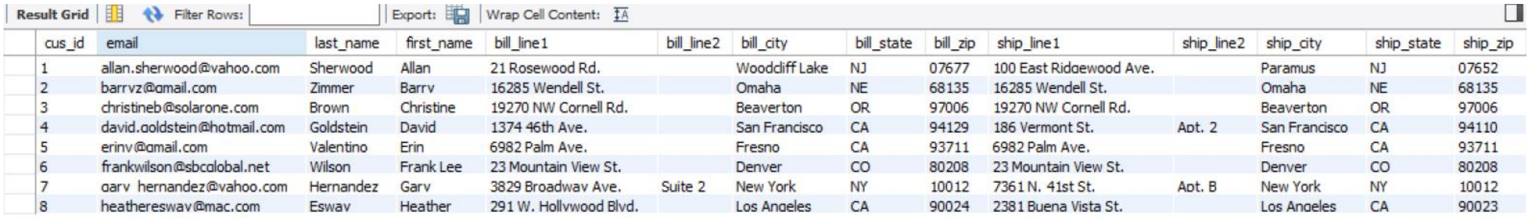


email_address	first_name	last_name	Download_Date	filename	Product_Name
johnsmith@gmail.com	John	Smith	2017-10-10 19:57:58.000000	pedals_are_falling.mp3	Local Music Vol 1
janedoe@yahoo.com	Jane	Doe	2017-10-10 19:57:58.000000	turn_signal.mp3	Local Music Vol 1
janedoe@yahoo.com	Jane	Doe	2017-10-10 19:57:58.000000	one_hourse_town.mp3	Local Music Vol 2

4. **Create a view named customer_addresses that shows the shipping and billing addresses for each customer. This view should return these columns from the Customers table: customer_id, email_address, last_name, and first_name. This view should also return these columns from the Addresses table: bill_line1, bill_line2, bill_city, bill_state, bill_zip, ship_line1, ship_line2, ship_city, ship_state, and ship_zip. The rows in this view should be sorted by the last_name and then first_name columns.**



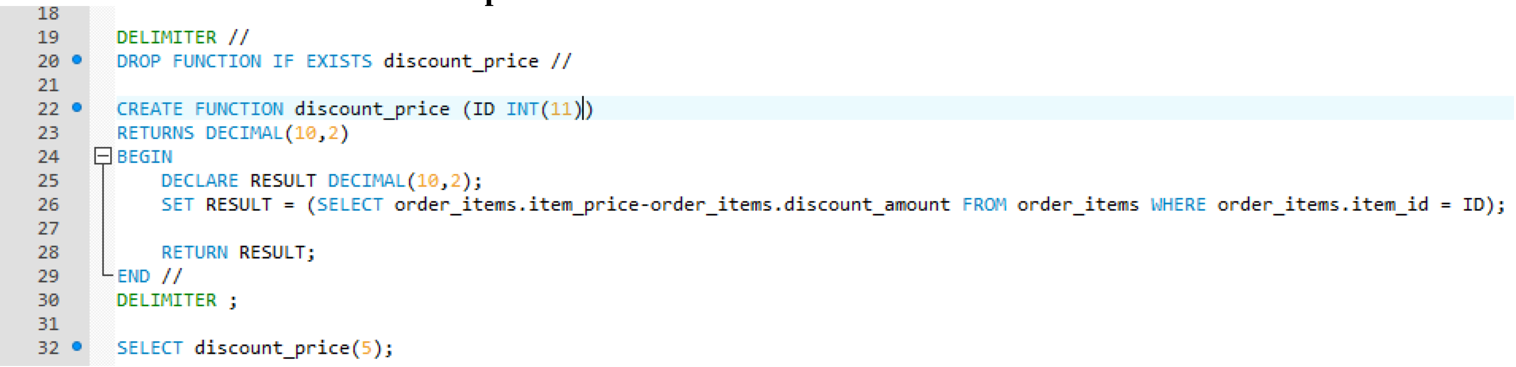
```
USE my_guitar_shop;
CREATE VIEW customer_addresses AS

SELECT bill_tab.cus_id, bill_tab.email,bill_tab.last_name,bill_tab.first_name,
 bill_tab.bill_line1,bill_tab.bill_line2,bill_tab.bill_city,bill_tab.bill_state,
 bill_tab.bill_zip,ship_tab.ship_line1,ship_tab.ship_line2,
 ship_tab.ship_city,ship_tab.ship_state,ship_zip
 FROM
    (SELECT customers.customer_id as cus_id,customers.email_address as email,
    customers.last_name as last_name ,customers.first_name as first_name,
    addresses.line1 AS bill_line1,addresses.line2 AS bill_line2,
    addresses.city AS bill_city, addresses.state AS bill_state,
    addresses.zip_code AS bill_zip
    FROM customers
    JOIN addresses ON addresses.address_id = customers.billing_address_id) AS bill_tab
JOIN (
    SELECT customers.customer_id as ship_id,
    addresses.line1 AS ship_line1,addresses.line2 AS ship_line2,
    addresses.city AS ship_city, addresses.state AS ship_state,
    addresses.zip_code AS ship_zip
    FROM customers
    JOIN addresses ON addresses.address_id = shipping_address_id
    ) AS ship_tab ON ship_tab.ship_id = bill_tab.cus_id;

SELECT * FROM customer_addresses;
```



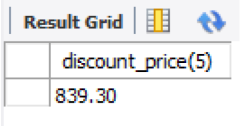
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

cus_id	email	last_name	first_name	bill_line1	bill_line2	bill_city	bill_state	bill_zip	ship_line1	ship_line2	ship_city	ship_state	ship_zip
1	allan.sherwood@yahoo.com	Sherwood	Allan	21 Rosewood Rd.		Woodcliff Lake	NJ	07677	100 East Ridgewood Ave.		Paramus	NJ	07652
2	barryz@gmail.com	Zimmer	Barry	16285 Wendell St.		Omaha	NE	68135	16285 Wendell St.		Omaha	NE	68135
3	christineb@solarone.com	Brown	Christine	19270 NW Cornell Rd.		Beaverton	OR	97006	19270 NW Cornell Rd.		Beaverton	OR	97006
4	david.goldstein@hotmail.com	Goldstein	David	1374 46th Ave.		San Francisco	CA	94129	186 Vermont St.	Apt. 2	San Francisco	CA	94110
5	erinv@gmail.com	Valentino	Erin	6982 Palm Ave.		Fresno	CA	93711	6982 Palm Ave.		Fresno	CA	93711
6	frankwilson@sbcglobal.net	Wilson	Frank Lee	23 Mountain View St.		Denver	CO	80208	23 Mountain View St.		Denver	CO	80208
7	gary_hernandez@yahoo.com	Hernandez	Gary	3829 Broadway Ave.	Suite 2	New York	NY	10012	7361 N. 41st St.	Apt. B	New York	NY	10012
8	heatheresway@mac.com	Esway	Heather	291 W. Hollywood Blvd.		Los Angeles	CA	90024	2381 Buena Vista St.		Los Angeles	CA	90023

5. **Write a script that creates and calls a stored function named discount_price that calculates the discount price of an item in the Order_Items table of the *my_guitar_shop* database (discount amount subtracted from item price). To do that, this function should accept one parameter for the item ID, and it should return the value of the discount price for that item.**



```

DELIMITER //
DROP FUNCTION IF EXISTS discount_price //

CREATE FUNCTION discount_price (ID INT(11))
RETURNS DECIMAL(10,2)
BEGIN
    DECLARE RESULT DECIMAL(10,2);
    SET RESULT = (SELECT order_items.item_price-order_items.discount_amount FROM order_items WHERE order_items.item_id = ID);

    RETURN RESULT;
END //
DELIMITER ;

SELECT discount_price(5);
```



Result Grid

discount_price(5)
839.30