Homework #2 **Due: turned in by Mon 01/27/2020 before class**

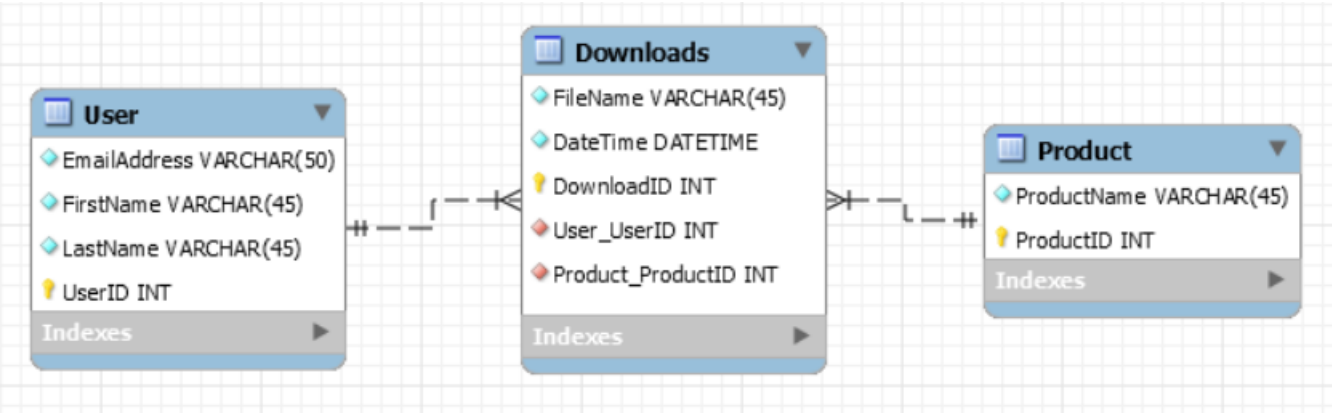
Adil Ashish Kumar

(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 HW2 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.**

```
-- MySQL Script generated by MySQL Workbench
-- Sun Jan 26 21:48:50 2020
-- 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='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DA
TE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';

-- -----------------------------------------------------
-- Schema my_web_db
-- -----------------------------------------------------

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

```
USE `my_web_db` ;

-- -----------------------------------------------------
-- Table `my_web_db`.`User`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `my_web_db`.`User` (
  `EmailAddress` VARCHAR(50) NOT NULL,
  `FirstName` VARCHAR(45) NOT NULL,
  `LastName` VARCHAR(45) NOT NULL,
  `UserID` INT NOT NULL,
  PRIMARY KEY (`UserID`))
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `my_web_db`.`Product`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `my_web_db`.`Product` (
  `ProductName` VARCHAR(45) NOT NULL,
  `ProductID` INT NOT NULL,
  PRIMARY KEY (`ProductID`))
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `my_web_db`.`Downloads`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `my_web_db`.`Downloads` (
  `FileName` VARCHAR(45) NOT NULL,
  `DateTime` DATETIME NOT NULL,
  `DownloadID` INT NOT NULL,
  `User_UserID` INT NOT NULL,
  `Product_ProductID` INT NOT NULL,
  PRIMARY KEY (`DownloadID`),
  INDEX `fk_Downloads_User_idx` (`User_UserID` ASC) VISIBLE,
  INDEX `fk_Downloads_Product1_idx` (`Product_ProductID` ASC) VISIBLE,
  CONSTRAINT `fk_Downloads_User`
    FOREIGN KEY (`User_UserID`)
    REFERENCES `my_web_db`.`User` (`UserID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Downloads_Product1`
    FOREIGN KEY (`Product_ProductID`)
    REFERENCES `my_web_db`.`Product` (`ProductID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


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
values ('johnsmith@gmail.com','John','Smith',1),
('janedoe@yahoo.com','Jane','Doe',2);

INSERT into product
values('Local Music Vol 1',100),
('Local Music Vol 2',101);

INSERT INTO downloads
values('pedals_are_falling.mp3', now(),401,1,100),
('turn_signal.mp3', now(),402,2,100),
('one_horse_down',  now(),403,2,101);

select emailaddress as email_address,firstname as first_name,lastname as last_name,datetime as
download_date,filename, productname as product_name
from user join downloads on userID = user_userID
join product on productid=product.productid
order by emailaddress desc,productname;
```

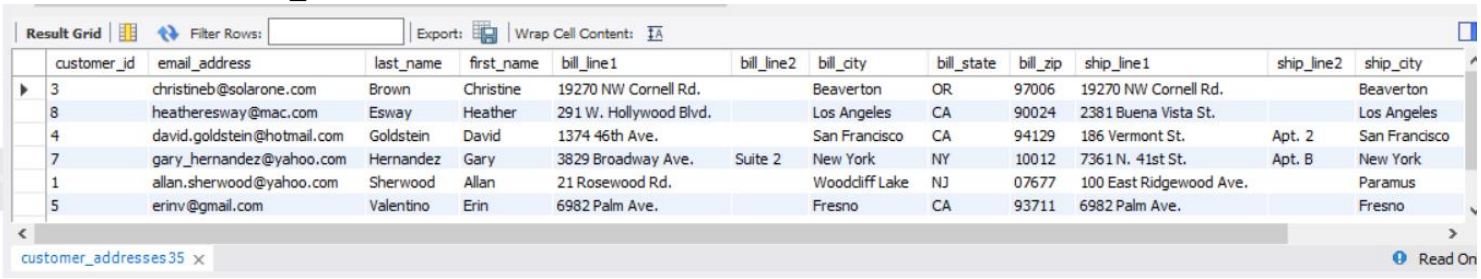
email_address	first_name	last_name	download_date	filename	product_name
johnsmith@gmail.com	John	Smith	2020-01-24 12:51:57	pedals_are_falling.mp3	Local Music Vol 1
janedoe@yahoo.com	Jane	Doe	2020-01-24 12:51:57	turn_signal.mp3	Local Music Vol 1
janedoe@yahoo.com	Jane	Doe	2020-01-24 12:51:57	one_horse_down.mp3	Local Music Vol 2

4. **Create a view named customer_addresses that shows the shipping and billing addresses for each customer in the *my_guitar_shop* database. 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.

```
create view customer_addresses as
select c.customer_id, c.email_address, c.last_name,c.first_name,a1.line1 as bill_line1,a1.line2 as bill_line2,a1.city as bill_city,a1.state as bill_state,a1.zip_code as bill_zip,a2.line1 as ship_line1,a2.line2 as ship_line2,a2.city as ship_city,a2.state as ship_state,a2.zip_code as ship_zip
from customers c,addresses a1 ,addresses a2
where c.billing_address_id = a1.address_id
and c.shipping_address_id = a2.address_id
order by last_name, first_name;
```

```
select * from customer_addresses
```



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

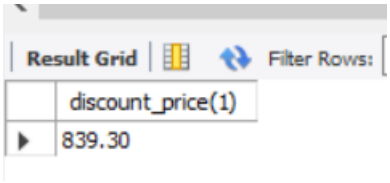
customer_id	email_address	last_name	first_name	bill_line1	bill_line2	bill_city	bill_state	bill_zip	ship_line1	ship_line2	ship_city
3	christineb@solarone.com	Brown	Christine	19270 NW Cornell Rd.		Beaverton	OR	97006	19270 NW Cornell Rd.		Beaverton
8	heatheresway@mac.com	Esway	Heather	291 W. Hollywood Blvd.		Los Angeles	CA	90024	2381 Buena Vista St.		Los Angeles
4	david.goldstein@hotmail.com	Goldstein	David	1374 46th Ave.		San Francisco	CA	94129	186 Vermont St.	Apt. 2	San Francisco
7	gary_hernandez@yahoo.com	Hernandez	Gary	3829 Broadway Ave.	Suite 2	New York	NY	10012	7361 N. 41st St.	Apt. B	New York
1	allan.sherwood@yahoo.com	Sherwood	Allan	21 Rosewood Rd.		Woodcliff Lake	NJ	07677	100 East Ridgewood Ave.		Paramus
5	erinv@gmail.com	Valentino	Erin	6982 Palm Ave.		Fresno	CA	93711	6982 Palm Ave.		Fresno

customer_addresses35 | Read On

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.

```
create function discount_price(
items_id int)
returns decimal(10,2)
deterministic
return
(select item_price-discount_amount as discount_price
from order_items
where item_id = items_id);
```

```
        select discount_price(1);
```



Result Grid | Filter Rows:

discount_price(1)
839.30