CSCI 5408

DATA MANAGEMENT AND WAREHOUSING

LAB ASSIGNMENT - 2

Banner ID: B00952865

GitLab Assignment Link:
https://git.cs.dal.ca/apurohit/CSCI5408_F23_B00952865_AdityaMaheshbhai_Purohit

Table of Contents

Conceptual Model: ER Diagram using Chen's notations

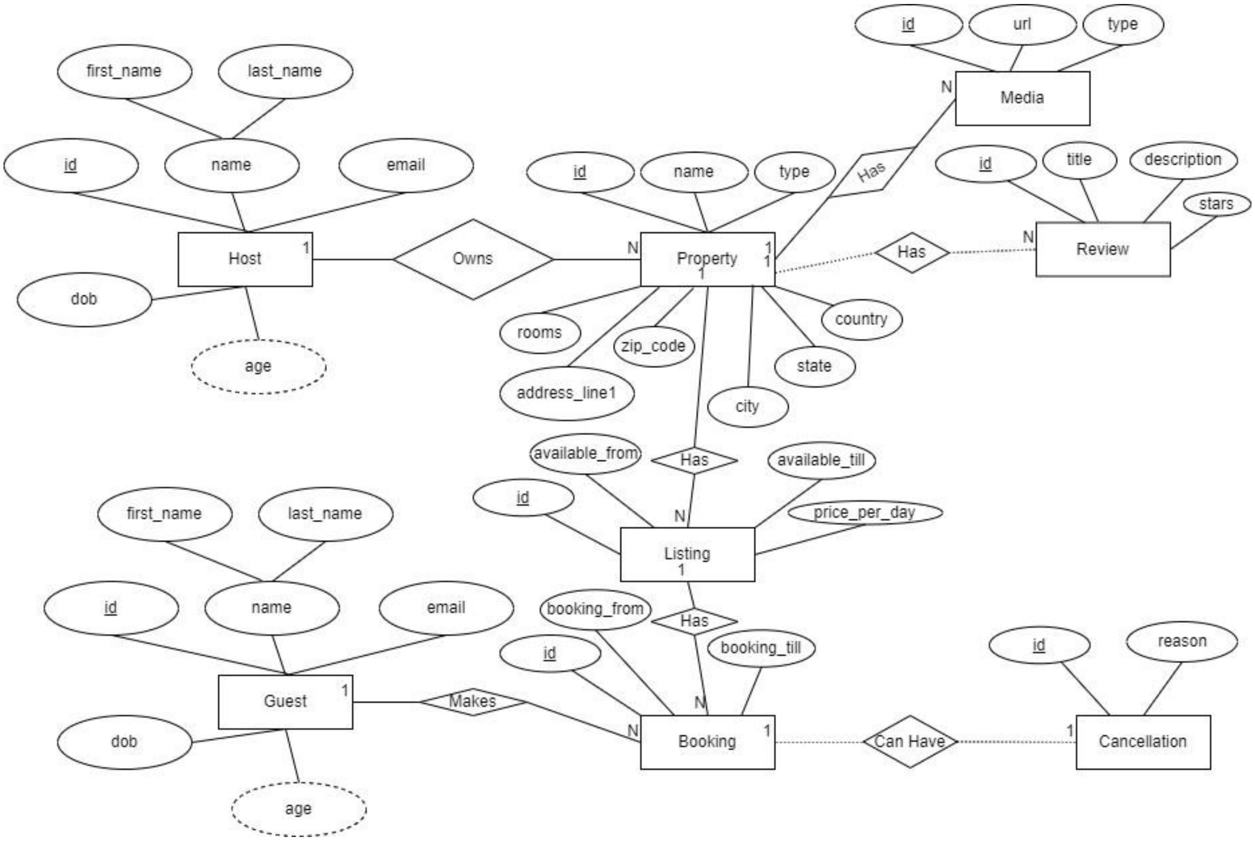


Figure 1: ER Diagram for AirBnB [1][2]

Explanation:

I have identified 8 entities: Host, Guest, Property, Listing, Booking, Media, Review and Cancellation.

Composite attribute name exists in Host and Guest attribute, as it can split into first and last name. Moreover, age is derived from date of birth (dob). Id is primary/key attribute in all entities.

For the relationship between, booking and cancellation entity, I have used optional participation, as only few or none of the bookings may get cancelled and not all. Similarly, between property and review, it's optional, as properties may not have any reviews yet.

Design Issues: No issue of history as booking entry remains even if cancelled, No Fan Trap exists as no entity has 2, 1-N relationships; having any actual relation between other 2 entities. Also, no chasm trap, as usually occurs when we resolve a Fan Trap.

Logical Model:

Partial Dependencies: Does not exist.

Explanation: There is no composite primary key, so no chance of partial dependency.

Transitive Dependencies: Does not exist.

Explanation: There is no such non-prime attribute that can uniquely identify another non-prime attribute.

Normalization:

1NF: Already in 1NF, as primary keys are already identified, there are no attributes which can have multiple values for it.

2NF: Already in 2NF, as it’s already in 1NF and there is no partial dependency.

3NF: Already in 3NF as it’s already in 2NF and there is no transitive dependency.

Updated Crow’s Foot ER Model:

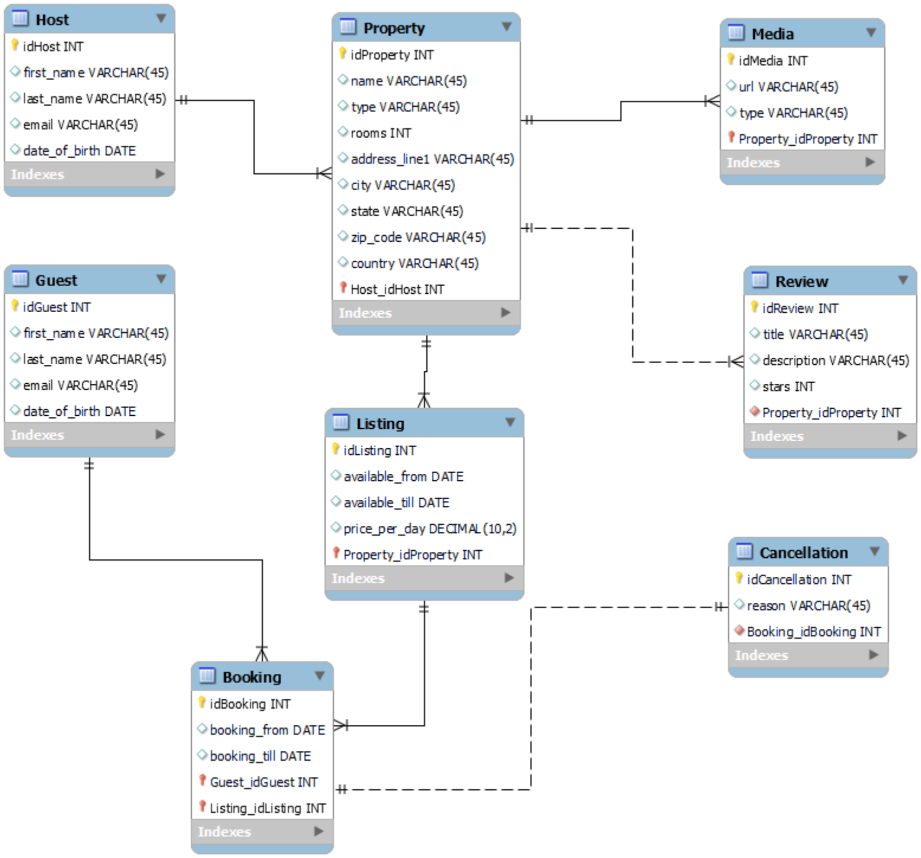


Figure 2: Crow-Foot ER Model, after logical phase. [3]

Physical Model:

After drawing the output of logical model in MySQL Workbench, I ran forward engineering to get the physical model. The basic step and output DDL queries are given below.

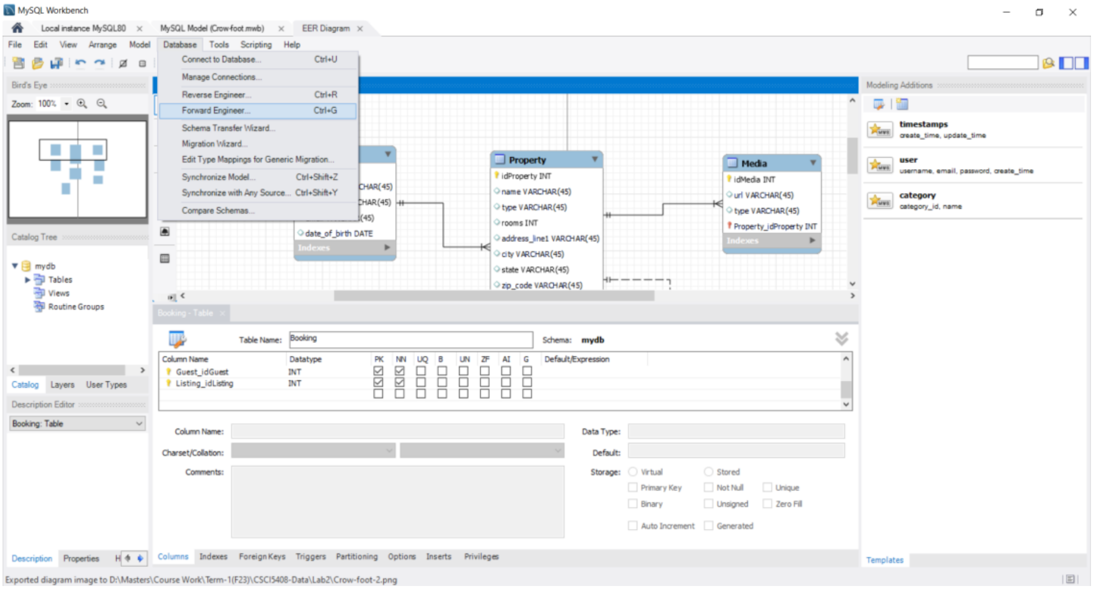


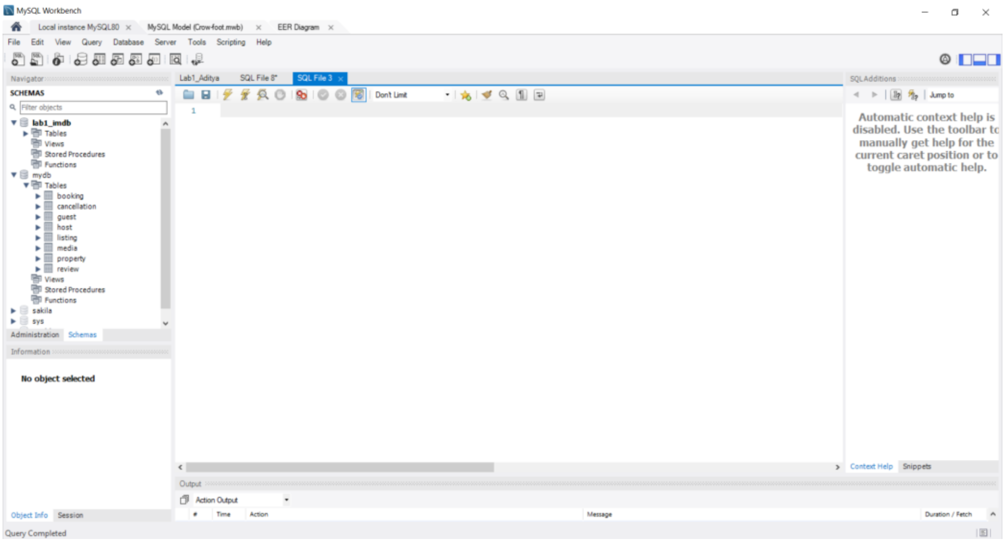
Table Name: Booking Schema: mydb

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
Guest_idGuest	INT	☑	☑	□	□	□	□	□	□	
Listing_idListing	INT	☑	☑	□	□	□	□	□	□	
		□	□	□	□	□	□	□	□	

Column Name:
Charset/Collation:
Comments:
Data Type:
Default:
Storage: ○ Virtual ○ Stored
□ Primary Key □ Not Null □ Unique
□ Binary □ Unsigned □ Zero Fill
□ Auto Increment □ Generated

Exported diagram image to D:\Masters\Course Work\Term-1(F23)\CSCI5408-Data\Lab2\Crow-foot-2.png

Figure 3: Run Forward Engineering. [3]



No object selected

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Query Completed

Figure 4: Physical Model. [3]

DDL Queries:

```
-- 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_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';


-- -----------------------------------------------------
-- Schema mydb
-- -----------------------------------------------------


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


-- -----------------------------------------------------
-- Table `mydb`.`Host`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Host` (
  `idHost` INT NOT NULL,
  `first_name` VARCHAR(45) NULL,
  `last_name` VARCHAR(45) NULL,
  `email` VARCHAR(45) NULL,
  `date_of_birth` DATE NULL,
  PRIMARY KEY (`idHost`),
  UNIQUE INDEX `email_UNIQUE` (`email` ASC) VISIBLE)
ENGINE = InnoDB;
```

```
-- -----------------------------------------------------
-- Table `mydb`.`Property`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Property` (
  `idProperty` INT NOT NULL,
  `name` VARCHAR(45) NULL,
  `type` VARCHAR(45) NULL,
  `rooms` INT NULL,
  `address_line1` VARCHAR(45) NULL,
  `city` VARCHAR(45) NULL,
  `state` VARCHAR(45) NULL,
  `zip_code` VARCHAR(45) NULL,
  `country` VARCHAR(45) NULL,
  `Host_idHost` INT NOT NULL,
  PRIMARY KEY (`idProperty`, `Host_idHost`),
  INDEX `fk_Property_Host_idx` (`Host_idHost` ASC) VISIBLE,
  CONSTRAINT `fk_Property_Host`
    FOREIGN KEY (`Host_idHost`)
    REFERENCES `mydb`.`Host` (`idHost`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Media`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Media` (
  `idMedia` INT NOT NULL,
  `url` VARCHAR(45) NULL,
  `type` VARCHAR(45) NULL,
  `Property_idProperty` INT NOT NULL,
  PRIMARY KEY (`idMedia`, `Property_idProperty`),
  INDEX `fk_Media_Property1_idx` (`Property_idProperty` ASC) VISIBLE,
  CONSTRAINT `fk_Media_Property1`
    FOREIGN KEY (`Property_idProperty`)
    REFERENCES `mydb`.`Property` (`idProperty`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Guest`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Guest` (
  `idGuest` INT NOT NULL,
  `first_name` VARCHAR(45) NULL,
  `last_name` VARCHAR(45) NULL,
  `email` VARCHAR(45) NULL,
  `date_of_birth` DATE NULL,
  PRIMARY KEY (`idGuest`),
  UNIQUE INDEX `email_UNIQUE` (`email` ASC) VISIBLE)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Listing`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Listing` (
  `idListing` INT NOT NULL,
  `available_from` DATE NULL,
  `available_till` DATE NULL,
  `price_per_day` DECIMAL(10,2) NULL,
  `Property_idProperty` INT NOT NULL,
  PRIMARY KEY (`idListing`, `Property_idProperty`),
  INDEX `fk_Listing_Property1_idx` (`Property_idProperty` ASC) VISIBLE,
  CONSTRAINT `fk_Listing_Property1`
    FOREIGN KEY (`Property_idProperty`)
    REFERENCES `mydb`.`Property` (`idProperty`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Review`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Review` (
  `idReview` INT NOT NULL,
  `title` VARCHAR(45) NULL,
  `description` VARCHAR(45) NULL,
  `stars` INT NULL,
  `Property_idProperty` INT NOT NULL,
  PRIMARY KEY (`idReview`),
  INDEX `fk_Review_Property1_idx` (`Property_idProperty` ASC) VISIBLE,
  CONSTRAINT `fk_Review_Property1`
    FOREIGN KEY (`Property_idProperty`)
    REFERENCES `mydb`.`Property` (`idProperty`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Booking`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Booking` (
  `idBooking` INT NOT NULL,
  `booking_from` DATE NULL,
  `booking_till` DATE NULL,
  `Guest_idGuest` INT NOT NULL,
  `Listing_idListing` INT NOT NULL,
  PRIMARY KEY (`idBooking`, `Guest_idGuest`, `Listing_idListing`),
  INDEX `fk_Booking_Guest1_idx` (`Guest_idGuest` ASC) VISIBLE,
  INDEX `fk_Booking_Listing1_idx` (`Listing_idListing` ASC) VISIBLE,
  CONSTRAINT `fk_Booking_Guest1`
    FOREIGN KEY (`Guest_idGuest`)
    REFERENCES `mydb`.`Guest` (`idGuest`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Booking_Listing1`
    FOREIGN KEY (`Listing_idListing`)
    REFERENCES `mydb`.`Listing` (`idListing`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Cancellation`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Cancellation` (
  `idCancellation` INT NOT NULL,
  `reason` VARCHAR(45) NULL,
  `Booking_idBooking` INT NOT NULL,
  PRIMARY KEY (`idCancellation`),
  INDEX `fk_Cancellation_Booking1_idx` (`Booking_idBooking` ASC) VISIBLE,
  CONSTRAINT `fk_Cancellation_Booking1`
    FOREIGN KEY (`Booking_idBooking`)
    REFERENCES `mydb`.`Booking` (`idBooking`)
    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;
```

References:

[1] “Holiday Homes & Apartment Rentals,” Airbnb, https://www.airbnb.co.in/ (accessed Sep. 30, 2023).

[2] “Free Flowchart Maker and diagrams online,” Flowchart Maker & Online Diagram Software, https://draw.io/ (accessed Sep. 30, 2023).

[3] “MySQL Workbench",” MySQL, https://www.mysql.com/products/workbench/ (accessed Sep. 30, 2023).