Screaming Beckers: Harry Tran, Daniel Stanev

Dr. Eric Becker

CS 4347.501

1 Nov 2023

Implementation Log

```
-- MySQL dump 10.13  Distrib 8.0.34, for Win64 (x86_64)
--
-- Host: localhost    Database: newschema
-- ------------------------------------------------------
-- Server version  8.0.34
DROP
  TABLE IF EXISTS article;
CREATE TABLE article (
  Media_ID int NOT NULL,
  Journal varchar(100) DEFAULT NULL,
  PRIMARY KEY (Media_ID),
  CONSTRAINT article_ibfk_1 FOREIGN KEY (Media_ID) REFERENCES media (Media_ID)
);
DROP
  TABLE IF EXISTS associations;
CREATE TABLE associations (
  Association varchar(100) NOT NULL,
  FK_Person_ID int NOT NULL,
  PRIMARY KEY (FK_Person_ID, Association),
  CONSTRAINT associations_ibfk_1 FOREIGN KEY (FK_Person_ID) REFERENCES
librarian (Person_ID)
);
DROP
  TABLE IF EXISTS audiobook;
CREATE TABLE audiobook (
  Media_ID int NOT NULL,
  Book_Length time DEFAULT NULL,
  PRIMARY KEY (Media_ID),
  CONSTRAINT audiobook_ibfk_1 FOREIGN KEY (Media_ID) REFERENCES book
(Media_ID)
);
DROP
  TABLE IF EXISTS authors;
CREATE TABLE authors (
  Author varchar(100) DEFAULT NULL,
  FK_Media_ID int NOT NULL,
  PRIMARY KEY (FK_Media_ID),
  CONSTRAINT authors_ibfk_1 FOREIGN KEY (FK_Media_ID) REFERENCES article
(Media_ID)
);
DROP
  TABLE IF EXISTS bank_accounts;
CREATE TABLE bank_accounts (
  Routing_Num int NOT NULL,
  FK_Person_ID int DEFAULT NULL,
  PRIMARY KEY (Routing_Num),
  KEY FK_Person_ID (FK_Person_ID),
  CONSTRAINT bank_accounts_ibfk_1 FOREIGN KEY (FK_Person_ID) REFERENCES
person (Person_ID)
);
DROP
  TABLE IF EXISTS book;
CREATE TABLE book (
  Media_ID int NOT NULL,
  Author varchar(100) DEFAULT NULL,
  ISBN int DEFAULT NULL,
  PRIMARY KEY (Media_ID),
  CONSTRAINT book_ibfk_1 FOREIGN KEY (Media_ID) REFERENCES media (Media_ID)
);
DROP
  TABLE IF EXISTS borrow;
CREATE TABLE borrow (
  FK_Person_ID int NOT NULL,
  FK_Media_ID int NOT NULL,
  Return_Date date DEFAULT NULL,
  PRIMARY KEY (FK_Person_ID, FK_Media_ID),
  KEY FK_Media_ID (FK_Media_ID),
  CONSTRAINT borrow_ibfk_1 FOREIGN KEY (FK_Person_ID) REFERENCES customer
(Person_ID),
  CONSTRAINT borrow_ibfk_2 FOREIGN KEY (FK_Media_ID) REFERENCES media
(Media_ID),
  CONSTRAINT borrow_ibfk_3 FOREIGN KEY (FK_Person_ID) REFERENCES customer
(Person_ID)
);
DROP
  TABLE IF EXISTS clerk;
CREATE TABLE clerk (
  Person_ID int NOT NULL,
  Register_No int DEFAULT NULL,
  training_complete tinyint(1) DEFAULT NULL,
  PRIMARY KEY (Person_ID),
  CONSTRAINT clerk_ibfk_1 FOREIGN KEY (Person_ID) REFERENCES employee
(Person_ID),
  CONSTRAINT clerk_ibfk_2 FOREIGN KEY (Person_ID) REFERENCES employee
(Person_ID)
);
DROP
  TABLE IF EXISTS customer;
CREATE TABLE customer (
  Person_ID int NOT NULL,
  Num_Borrowed int DEFAULT NULL,
  Borrow_Limit int DEFAULT NULL,
  PRIMARY KEY (Person_ID),
  CONSTRAINT customer_ibfk_1 FOREIGN KEY (Person_ID) REFERENCES person
(Person_ID),
  CONSTRAINT customer_ibfk_2 FOREIGN KEY (Person_ID) REFERENCES person
(Person_ID)
);
--
-- Dumping data for table `customer`
--
--
-- Table structure for table `employee`
--
DROP
  TABLE IF EXISTS employee;
CREATE TABLE employee (
  Person_ID int NOT NULL,
  Supervisor varchar(50) DEFAULT NULL,
  PRIMARY KEY (Person_ID),
  CONSTRAINT employee_ibfk_1 FOREIGN KEY (Person_ID) REFERENCES person
(Person_ID),
  CONSTRAINT employee_ibfk_2 FOREIGN KEY (Person_ID) REFERENCES person
(Person_ID)
);
DROP
  TABLE IF EXISTS keywords;
CREATE TABLE keywords (
  Keyword varchar(100) NOT NULL,
  FK_Media_ID int NOT NULL,
  PRIMARY KEY (Keyword, FK_Media_ID),
  KEY FK_Media_ID (FK_Media_ID),
  CONSTRAINT keywords_ibfk_1 FOREIGN KEY (FK_Media_ID) REFERENCES article
(Media_ID)
);
DROP
```

```
  TABLE IF EXISTS librarian;
CREATE TABLE librarian (
  Person_ID int NOT NULL,
  Degree varchar(50) DEFAULT NULL,
  Policy_Develop tinyint(1) DEFAULT NULL,
  PRIMARY KEY (Person_ID),
  CONSTRAINT librarian_ibfk_1 FOREIGN KEY (Person_ID) REFERENCES employee
(Person_ID),
  CONSTRAINT librarian_ibfk_2 FOREIGN KEY (Person_ID) REFERENCES employee
(Person_ID)
);
DROP
  TABLE IF EXISTS media;
CREATE TABLE media (
  Media_ID int NOT NULL AUTO_INCREMENT,
  Title varchar(100) DEFAULT NULL,
  Genre varchar(50) DEFAULT NULL,
  Publisher varchar(100) DEFAULT NULL,
  Total_Stock int DEFAULT NULL,
  Available_Stock int DEFAULT NULL,
  PRIMARY KEY (Media_ID)
);
DROP
  TABLE IF EXISTS movie;
CREATE TABLE movie (
  Media_ID int NOT NULL,
  Movie_Length time DEFAULT NULL,
  Production_Co varchar(50) DEFAULT NULL,
  PRIMARY KEY (Media_ID),
  CONSTRAINT movie_ibfk_1 FOREIGN KEY (Media_ID) REFERENCES media (Media_ID)
);
DROP
  TABLE IF EXISTS music;
CREATE TABLE music (
  Media_ID int NOT NULL,
  Artist varchar(50) DEFAULT NULL,
  Album varchar(50) DEFAULT NULL,
  Song_Length time DEFAULT NULL,
  PRIMARY KEY (Media_ID),
  CONSTRAINT music_ibfk_1 FOREIGN KEY (Media_ID) REFERENCES media (Media_ID)
);
DROP
  TABLE IF EXISTS person;
CREATE TABLE person (
  Person_ID int NOT NULL AUTO_INCREMENT,
  first_name varchar(32) DEFAULT NULL,
  last_name varchar(32) DEFAULT NULL,
  Street_Address varchar(100) DEFAULT NULL,
  City varchar(50) DEFAULT NULL,
  State varchar(2) DEFAULT NULL,
  PRIMARY KEY (Person_ID)
);
DROP
  TABLE IF EXISTS private_room;
CREATE TABLE private_room (
  Room_Num int NOT NULL,
  Max_Occupancy int DEFAULT NULL,
  Has_Screen tinyint(1) DEFAULT NULL,
  Has_Whiteboard tinyint(1) DEFAULT NULL,
  PRIMARY KEY (Room_Num)
);
DROP
  TABLE IF EXISTS rent;
CREATE TABLE rent (
  FK_Person_ID int NOT NULL,
  FK_Room_Num int NOT NULL,
  `Date` date DEFAULT NULL,
  PRIMARY KEY (FK_Person_ID, FK_Room_Num),
  KEY FK_Room_Num (FK_Room_Num),
  CONSTRAINT rent_ibfk_1 FOREIGN KEY (FK_Room_Num) REFERENCES private_room
(Room_Num),
  CONSTRAINT rent_ibfk_2 FOREIGN KEY (FK_Person_ID) REFERENCES customer
(Person_ID),
  CONSTRAINT rent_ibfk_3 FOREIGN KEY (FK_Person_ID) REFERENCES customer
(Person_ID)
);
DROP
  TABLE IF EXISTS schedule;
CREATE TABLE `schedule` (
  Person_ID int NOT NULL,
  Shifts varbinary(14) DEFAULT NULL,
  Wage int DEFAULT NULL,
  PRIMARY KEY (Person_ID),
  CONSTRAINT schedule_ibfk_1 FOREIGN KEY (Person_ID) REFERENCES employee
(Person_ID),
  CONSTRAINT schedule_ibfk_2 FOREIGN KEY (Person_ID) REFERENCES employee
(Person_ID)
)
DROP
  TABLE IF EXISTS supervision;
CREATE TABLE supervision (
  FK_Sub_ID int NOT NULL,
  FK_Super_ID int NOT NULL,
  PRIMARY KEY (FK_Sub_ID, FK_Super_ID),
  KEY FK_Super_ID (FK_Super_ID),
  CONSTRAINT supervision_ibfk_1 FOREIGN KEY (FK_Sub_ID) REFERENCES employee (Person_ID),
  CONSTRAINT supervision_ibfk_2 FOREIGN KEY (FK_Super_ID) REFERENCES supervisor (Person_ID)
);
DROP
  TABLE IF EXISTS supervisor;
CREATE TABLE supervisor (
  Person_ID int NOT NULL,
  Department int DEFAULT NULL,
  Subordinates int DEFAULT NULL,
  PRIMARY KEY (Person_ID),
  CONSTRAINT supervisor_ibfk_1 FOREIGN KEY (Person_ID) REFERENCES employee (Person_ID),
  CONSTRAINT supervisor_ibfk_2 FOREIGN KEY (Person_ID) REFERENCES employee (Person_ID),
  CONSTRAINT supervisor_ibfk_3 FOREIGN KEY (Person_ID) REFERENCES employee (Person_ID)
);
DROP
  TABLE IF EXISTS volunteer;
CREATE TABLE volunteer (
  Person_ID int NOT NULL,
  Department int DEFAULT NULL,
  Court_mandated tinyint(1) DEFAULT NULL,
  PRIMARY KEY (Person_ID),
  CONSTRAINT volunteer_ibfk_1 FOREIGN KEY (Person_ID) REFERENCES employee (Person_ID),
  CONSTRAINT volunteer_ibfk_2 FOREIGN KEY (Person_ID) REFERENCES employee (Person_ID)
)
```

Date	**Activity**	**Description**
10.20.2023 2am	Created Schema Created all tables	I started using DB Browser to create the schema, and I also created the tables from our diagram.
10.22.2023 3pm	Added attributes	I used the dictionary from the last phase to create attributes for all of the tables.

10.27.2023 2am	Added primary keys Added foreign keys	I again used the dictionaries and the diagram to create the primary keys first and then created the foreign keys.
10.29.2023 6pm	Moved to MySQL	I was getting angry with how limited DB Browser was, so I had started to move the schema over to MySQL
10.30.2023 10am	Started Fixing Attributes	There were basically only two choices for what datatype I could use which were string and int, so I updated the attributes for Clerk, Librarian, Private_Room, and Volunteer.
10.31.2023 6pm	Finished Fixing Attributes	Continuation of Last Activity
11.01.2023 4pm	Figuring out MySQL cmd prompt. Everything works.	I figured out how to connect to the local database through the terminal to be able to do the TableListing part of the assignment. I haven’t encountered any errors.
11.02.2023 6:13pm	Forgot Something	Added in the Keywords table and created the keys. Added the DESC keywords to the tableListing file.