

Course Number	COE 848
Course Title	Fundamentals of Data Engineering
Semester/Year	2022 Winter
Instructor	Dr. Faezeh Ensan
Teaching Assistant	Shirin Seyedsalehi

Lab Report No.	4
-----------------------	----------

Report Title	Lab 4 Report
--------------	---------------------

Section No.	2
Submission Date	March 14th, 2022
Due Date	March 14th, 2022

Name	Student ID	Signature*
Yasar Zaman	xxxx13595	Y.Z

(Note: remove the first 4 digits from your student ID)

**By signing above you attest that you have contributed to this submission and confirm that all work you have contributed to this submission is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a "0" on the work, an "F" in the course, or possibly more severe penalties, as well as a Disciplinary Notice on your academic record under the Student Code of Academic Conduct, which can be found online at:*

<http://www.ryerson.ca/senate/policies/pol60.pdf>.

Tables

```
PRAGMA foreign_keys=OFF;
BEGIN TRANSACTION;
CREATE TABLE contract (
contractID INTEGER PRIMARY KEY AUTOINCREMENT,
contractSalary INTEGER NOT NULL,
contractExpire DATE,
studioID INT NOT NULL,
vaID INT NOT NULL,
CONSTRAINT fk_studioID FOREIGN KEY (studioID) REFERENCES studio (studioID),
CONSTRAINT fk_vaID FOREIGN KEY (vaID) REFERENCES voiceActor (vaID)
);
INSERT INTO contract VALUES(1,1000000,1991,1,1);
INSERT INTO contract VALUES(2,200000,1985,2,2);
INSERT INTO contract VALUES(3,100000,1975,3,3);
INSERT INTO contract VALUES(4,150000,1978,4,4);
INSERT INTO contract VALUES(5,250000,1981,5,5);
INSERT INTO contract VALUES(6,100000,1970,6,6);
INSERT INTO contract VALUES(7,250000,1980,3,7);
INSERT INTO contract VALUES(8,150000,1981,7,8);
INSERT INTO contract VALUES(9,200000,1980,8,9);
INSERT INTO contract VALUES(10,150000,1977,9,10);
CREATE TABLE merchandise (
merchandiseID INTEGER PRIMARY KEY AUTOINCREMENT,
merchandiseType VARCHAR(255) NULL,
merchandiseMat VARCHAR(255) NULL,
animeID INT NOT NULL,
CONSTRAINT fk_animeID FOREIGN KEY (animeID) REFERENCES anime (animeID),
CONSTRAINT ck_merchandiseType UNIQUE (merchandiseType)
);
INSERT INTO merchandise VALUES(1,'Figure','Plastic',1);
INSERT INTO merchandise VALUES(2,'Pillow','Polyester',7);
INSERT INTO merchandise VALUES(3,'Sword','Metal',5);
```

```

CREATE TABLE author (
authorID INTEGER PRIMARY KEY AUTOINCREMENT,
authorName VARCHAR(255) NOT NULL,
authorDOB DATE,
authorAge INT NOT NULL,
contractID INT NOT NULL,
CONSTRAINT fk_contractID FOREIGN KEY (contractID) REFERENCES contract (contractID),
CONSTRAINT ck_author UNIQUE (authorName)
);
INSERT INTO author VALUES(1,'Masashi Kishimoto',1975.1179999999999381,47,1);
INSERT INTO author VALUES(2,'Matcha Hazuki',1968,31,2);
INSERT INTO author VALUES(3,'Hiro Fujiwara',1946,40,3);
INSERT INTO author VALUES(4,'Haruho Shiina',1942,46,4);
INSERT INTO author VALUES(5,'Nobuhiro Watsuki',1939,51,5);
INSERT INTO author VALUES(6,'Kanae Hazuki',1962,41,6);
INSERT INTO author VALUES(7,'Yuyuko Takemiya',1952,44,7);
INSERT INTO author VALUES(8,'Miki Yoshikawa',1970,39,8);
INSERT INTO author VALUES(9,'Satoshi Wagahara',1958,40,9);
INSERT INTO author VALUES(10,'Karino Takatsu',1955,39,10);
CREATE TABLE voiceActor (
vaID INTEGER PRIMARY KEY AUTOINCREMENT,
vaName VARCHAR(255) NOT NULL,
vaDOB DATE,
vaSex VARCHAR(255) NOT NULL,
vaAge INT NOT NULL,
contractID INT NOT NULL,
CONSTRAINT fk_contractID FOREIGN KEY (contractID) REFERENCES contract (contractID),
CONSTRAINT ck_voiceActor UNIQUE (vaName)
);
INSERT INTO voiceActor VALUES(1,'Junko Takeuchi',1963,'Female',49,1);
INSERT INTO voiceActor VALUES(2,'Yoshitaka Yamaya',1975,'Male',31,2);
INSERT INTO voiceActor VALUES(3,'Nobuhiko Okamoto',1952,'Male',36,3);
INSERT INTO voiceActor VALUES(4,'Mamiko Noto',1972,'Female',43,4);
INSERT INTO voiceActor VALUES(5,'Mayo Suzukaze',1940,'Female',62,5);
INSERT INTO voiceActor VALUES(6,'Ai Kayano',1965,'Female',35,6);
INSERT INTO voiceActor VALUES(7,'Junji Majima',1960,'Male',44,7);
INSERT INTO voiceActor VALUES(8,'Sori Hayami',1957,'Female',32,8);
INSERT INTO voiceActor VALUES(9,'Ryouta Oosaka',1976,'Male',36,9);
INSERT INTO voiceActor VALUES(10,'Jun Fukuyama',1941,'Male',44,10);

```

```

CREATE TABLE studio (
studioID INTEGER PRIMARY KEY AUTOINCREMENT,
studioName VARCHAR(255) NOT NULL,
studioDOB DATE,
studioAge INT NOT NULL,
contractID INT NOT NULL,
CONSTRAINT fk_contractID FOREIGN KEY (contractID) REFERENCES contract (contractID),
CONSTRAINT ck_studio UNIQUE (studioName)
);
INSERT INTO studio VALUES(1,'Studio Pierrot',1969,43,1);
INSERT INTO studio VALUES(3,'J.C.Staff',1967,36,3);
INSERT INTO studio VALUES(4,'Production I.G',1985,15,4);
INSERT INTO studio VALUES(5,'Gallop',1953,44,5);
INSERT INTO studio VALUES(6,'Zecxs',1974,25,6);
INSERT INTO studio VALUES(7,'LIDENFILMS',1988,10,7);
INSERT INTO studio VALUES(8,'White Fox',1999,15,8);
INSERT INTO studio VALUES(9,'A-1 Pictures',1991,17,9);
INSERT INTO studio VALUES(10,'Studio Deen',1958,47,10);
CREATE TABLE anime (

```

```

CREATE TABLE anime (
animeID INTEGER PRIMARY KEY AUTOINCREMENT,
animeName VARCHAR(255) NOT NULL,
animeGenre VARCHAR(255) NOT NULL,
animeRD DATE,
animeNOE INT NOT NULL,
animeRating REAL NOT NULL,
animeAge INT NOT NULL,
contractID INT NOT NULL,
CONSTRAINT fk_contractID FOREIGN KEY (contractID) REFERENCES contract (contractID),
CONSTRAINT ck_animeName UNIQUE (animeName)
);
INSERT INTO anime VALUES(1,'Naruto Shippuden','Action',1990,500,8.2300000000000004263,15,1);
INSERT INTO anime VALUES(2,'One Week Friends','Slice of Life',2003,12,7.540000000000000355,7,2);
INSERT INTO anime VALUES(3,'Kaichou wa Maid-Sama','Romance',2004,26,8.019999999999995736,11,3);
INSERT INTO anime VALUES(4,'Kimi no Todoke','Romance',1998,37,8.0,10,4);
INSERT INTO anime VALUES(5,'Rurouni Kenshin','Action',1985,94,8.3000000000000007105,16,5);
INSERT INTO anime VALUES(6,'Say I love you','Romance',1974,10,6.8099999999999996092,9,6);
INSERT INTO anime VALUES(7,'Toradora!','Romance',1996,11,8.1199999999999992184,13,7);
INSERT INTO anime VALUES(8,'Yamada-kun and the Seven Witches','Slice of Life',1999,12,7.58000000000000071,6,8);
INSERT INTO anime VALUES(9,'The Devil is a Part-Timer!','Comedy',2005,13,7.790000000000000355,8,9);
INSERT INTO anime VALUES(10,'Wagnaria!!','Slice of Life',2002,39,7.83000000000000071,11,10);
DELETE FROM sqlite_sequence;
INSERT INTO sqlite_sequence VALUES('contract',10);
INSERT INTO sqlite_sequence VALUES('author',10);
INSERT INTO sqlite_sequence VALUES('merchandise',3);
INSERT INTO sqlite_sequence VALUES('studio',10);
INSERT INTO sqlite_sequence VALUES('anime',10);
INSERT INTO sqlite_sequence VALUES('voiceActor',10);
COMMIT;

```

*One important note is that the entity merchandise does not have 10 rows. This is because it is a weak entity and thus I feel there is no need to make it have 10 rows.

Queries

1) What is the most watched genre of anime?

```

SELECT animeGenre, COUNT(animeGenre) AS cnt FROM anime
GROUP BY animeGenre,
ORDER BY cnt DESC,

```

This will count the number of occurrences of the genre of the anime from the table anime and order them in descending order. The anime genre on the top will be the most watched genre.

2) Which voice actor appears the most out of all the anime?

```

SELECT vaName, COUNT(vaName) AS cnt FROM voiceActor,
GROUP BY vaName,
ORDER BY cnt DESC;

```

This will count the number of occurrences the name of the voiceActor will appear in the table voiceActor and then order them in descending order. The voice actor's name at the top will be the one that appears the most out of all the anime in the database.

3) Which anime has at least a 7.00 rating?

```
SELECT animeName FROM anime WHERE animeRating >= 7.00 ORDER BY  
animeRating
```

This will select the anime that have at least a 7.00 rating.

4) Which year of anime did you watch the most?

```
SELECT animeRD, COUNT(strftime('%Y', animeRD)) AS cnt FROM anime,  
GROUP BY animeRD,  
ORDER BY cnt DESC;
```

This will arrange the table by years and organize it by the ones that occur the most at the top.

5) What is your lowest rated anime?

```
SELECT min(animeRating) FROM anime;
```

This will select the lowest rating from all the anime. It won't give the name, but that is because the question isn't asking for the name.

6) What was the highest rated season?

```
SELECT max(animeRating) FROM anime;
```

This will select the highest rating from all the anime. It won't give the name, but that is because the question isn't asking for the name.

7) How many anime did a certain voice actor play in?

```
SELECT COUNT(vaName) WHERE vaName = 'Ai Kayano';
```

This will select the number of times the voice actor Ai Kayano has occurred in the table.

8) What is the average length in episodes of a particular genre?

```
SELECT avg(animeNOE) FROM anime WHERE animeGenre = 'Slice of Life';
```

This will generate the average number of episodes from the table anime where the genre is slice of life.

9) Which studio is the oldest established studio?

```
SELECT studioDOB, COUNT(strftime('%Y', studioDOB)) AS cnt FROM
studio,
GROUP BY studioDOB,
ORDER BY cnt ASC;
```

This will create the table in ascending order by the year the studio was established. The top of the table will be the oldest established studio.

10) What is the average rating for a particular genre?

```
SELECT avg(animeRating) FROM anime WHERE animeGenre = 'Slice of
Life';
```

This will generate the average rating for the anime genre of slice of life.