ANLT-242 RELATIONAL DATABASES

COURSE PROJECT:

BIG TEETH REALITY TV SHOW DATABASE

by

TEAM STU

Matthew Swogger
989295197
m_swogger@u.pacific.edu

Ali Taheri
989294669
s_taheritari@u.pacific.edu

Arda Ugur
989289609
a_ugur1@pacific.edu

School of Engineering and Computer Science, Ms. Sci. Data Science
University of the Pacific

December 2017

This page intentionally left blank.

Table of Contents

Business Case

A new reality show is entering production. A cross between Animal Planet and Fear Factor, it focuses on conflicts between humans and animals with big teeth (crocodiles, tigers, lions, and so on). Contestants never know if they will be chased by the animals or if they will be eating them. The insurance costs will be huge, but the producers think there is a big audience. Producers always think that, but at least they are willing to pay money to find out. The next step is to find people crazy enough to sign the waivers and participate in the show.

Contestant Details

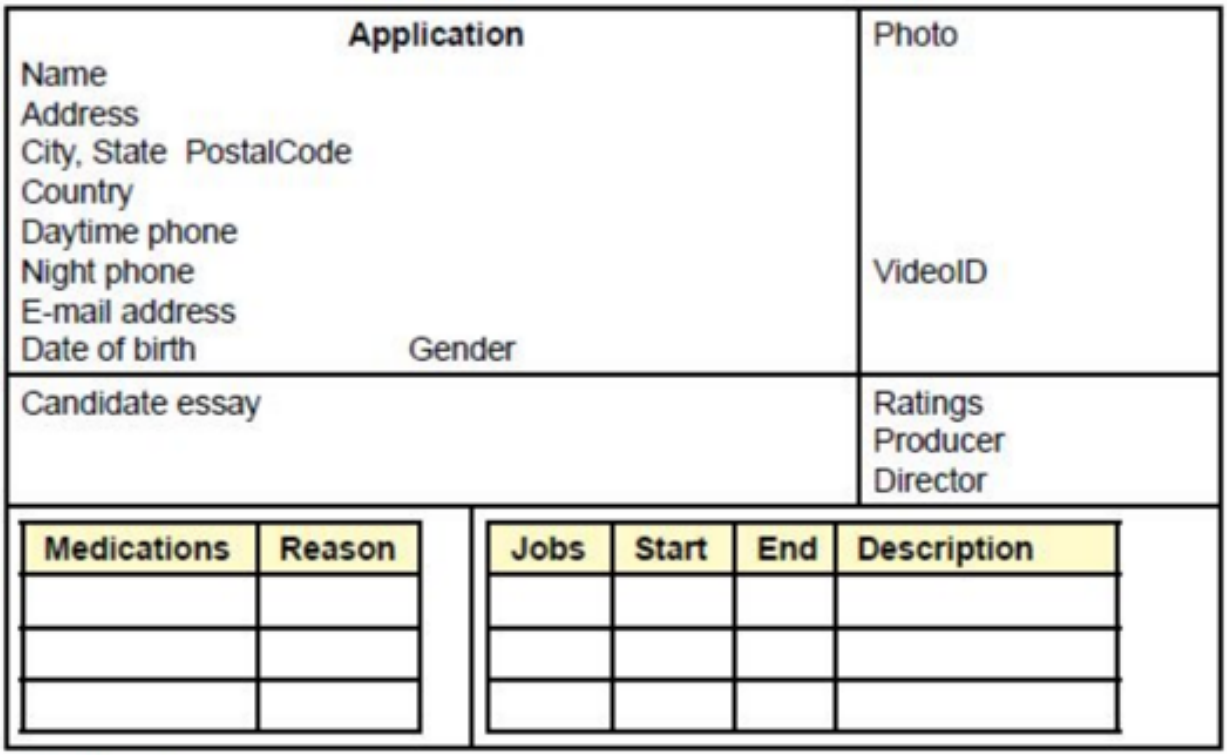


Figure 1

Figure-1 shows the basic contestant application form. The producers are adding a twist and recruiting worldwide. The goal is to build more suspense by adding communication problems among contestants and perhaps to foster some nationalistic audience participation. Contestants are asked to submit a photo, which is scanned and stored in the database. They are also encouraged to submit a short video interview. These interviews are only used for evaluating the final tier of candidates and portions might be used during a broadcast. The lead producer and the director evaluate each applicant and provide a quick rating (1-5). The highest rated candidates get a second look and become a finalist in the selection process.

All applicants undergo a background check. Several reality-based shows have experienced problems when contestants were revealed to have unsavory pasts— including being arrested for violent crimes. Although the producers want risk-taking contestants, they also want to avoid embarrassing public disclosures. **Figure-2** shows the basic background data that investigators obtain on the finalist contestants. It is not foolproof, but by making a few phone calls, the investigators get a reasonable idea of the candidate's background. National ID and Religion would be text fields. Appearance Rating and Strength Rating are numeric from 1-10. The producers and directors then select the contestants for an episode.

Note: The Employer list below does not have to link directly to the Job list in **Figure-1.**

Background Check

Applicant
National ID Appearance Rating
Religion Strength Rating

Employer	Phone	Comments

Education	Contact	Degree	Comments

Police and Judicial Records

Date	Category	Description	Outcome

Figure 2

Episodes and Events

Actual episodes consist of one or more events. Generally, there is only time for one or two events per episode, but the producers are thinking about the possibility of an hour- long special. **Figure-3** shows the two main aspects of the event: the setup from the perspective of the director, and the results from the perspective of the contestants. The director focuses on the sequence of actions, where the cameras will be located, and the estimated time of each section. For each event, a contestant is given one primary task. Sometimes they work in teams, so two or three people might be given the same task. The task result might just be a completion, or it might include a time for races. Either way, contestants are awarded points (sometimes negative points) for their role in the task (points range from -10 to +10). Occasionally, small prizes are awarded for completing a task, but contestants are really competing for the overall prizes in the event. The danger level will be numeric from 1-10.

Event	
Title Description Estimated time Estimated danger	Producer Director Episode

Contestants

Name	Task	Result	Points	Prize

Actions

Seq.	Description	Cameras	Est. Time

Figure 3

Voting

When the episode is aired, the audience is asked to participate by voting for favored contestants. The vote totals are used to determine which contestants to bring back for future episodes, and to give prizes for the overall season leader. Each vote has the episode title, date, contestant, region, and method associated with it. The executive producers are trying to entice advertisers by supporting several means of voting. Consequently, they want to track the actual method used to cast a vote. Common methods include telephone, cell phone/text messaging, e-mail, and a website. These should be available in a dropdown on a form which means it is not a simple text field. They also need to track audience participation by geographic region. On the report shown in **Figure-4**, the regional level is global and also supports a dropdown list of predefined areas (you can use continents, but it could be broken down to finer grained level).

Voting

Episode Title
Episode Air Date

Contestant	Region	Method	Votes
			Total

Contestant	Region	Method	Votes
			Total

...	

Figure 4

Required Business Solution

1. Create a normalized list of tables for each of the above forms.

2. Create an ERD which shows these tables and fields.

3. Define the tables in a DBMS along with the necessary relationships and integrity constraints.

4. Enter sample data to populate the tables to test your design.

5. Create queries to answer the following questions:

    a. Calculate the average producer and director ratings for a specific event's team members (you pick the event title).

    b. Determine which actions will take the longest. Rank all actions from longest to shortest.

    c. List each contestant's total votes by region and method for a specific episode (you pick the episode title). Rank this from highest to lowest.

    d. Identify which contestants have not participated in any events.

    e. What is the highest estimated danger level for any event?

    f. Show a relational expression for any one query

Note: Insert constraints on the fields specified in above that have limits on them.

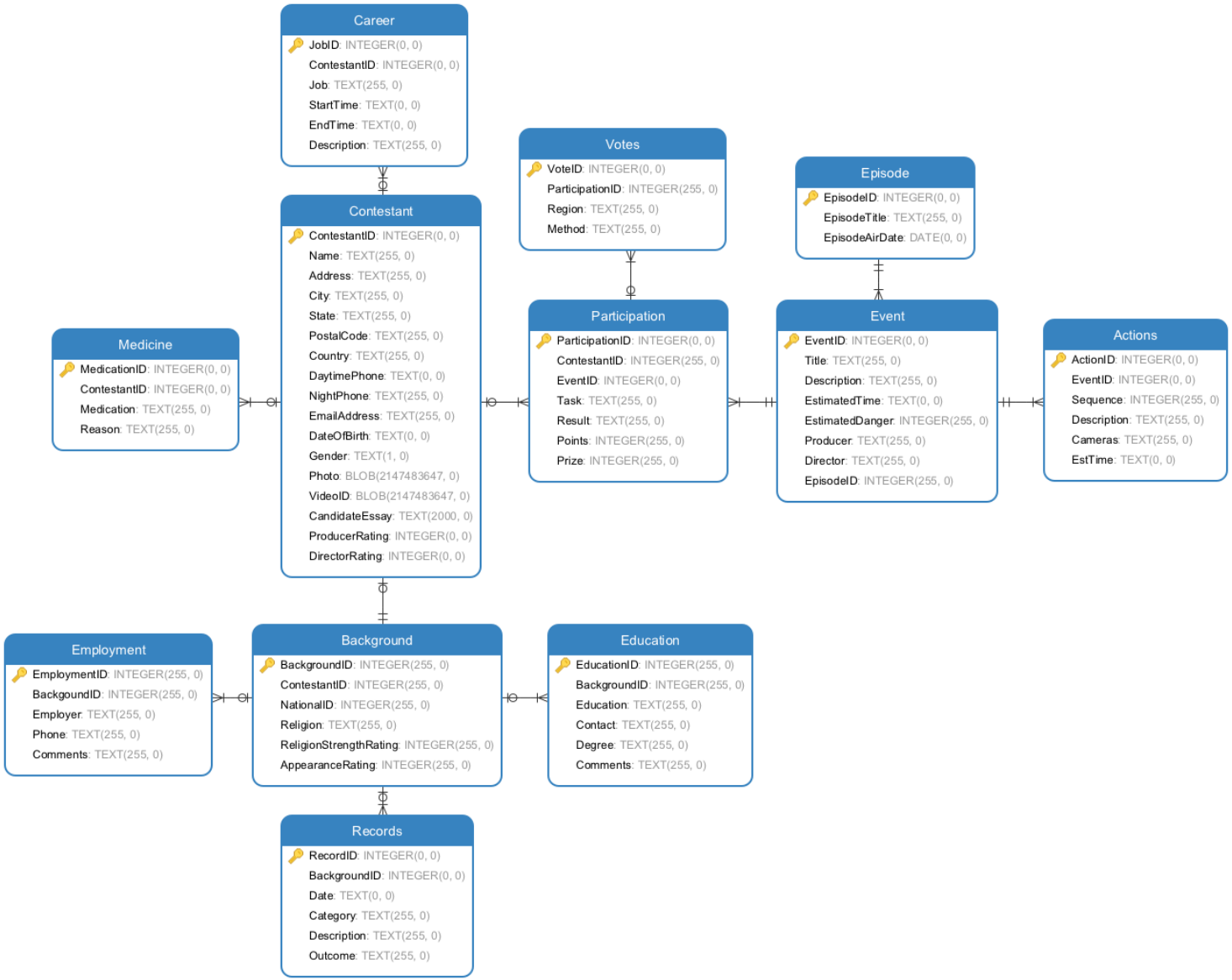
Proposed Business Solutions

Normalized List of Tables

The following list shows the database tables attributed to each information form collected from the applicants of **Big Theeth Reality TV** show.

Information Form		Corresponding Database Table(s)
Basic Contestant Application Form		Contestant
		Medicine
		Career
Background Check Form		Background
		Employment
		Education
		Records
Episodes and Events Form		Event
		Participation
		Actions
		Episode
Voting		Votes

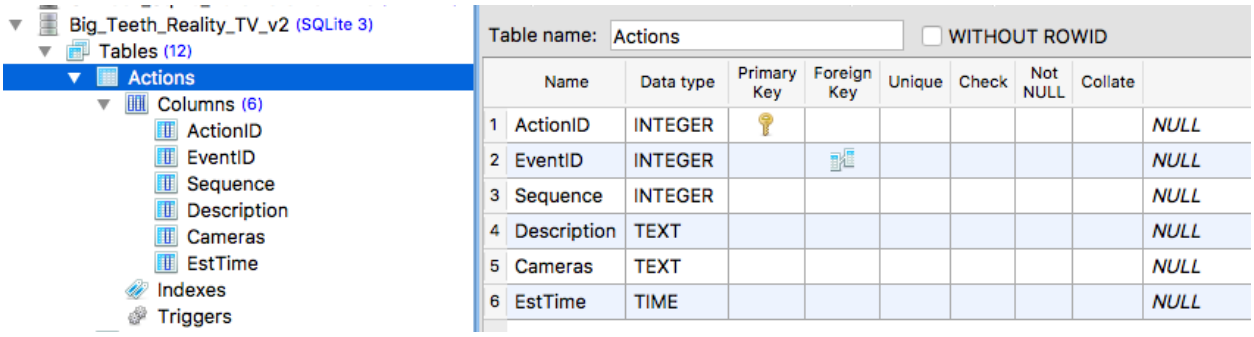
Database Entity Relationship Diagram



Database Tables

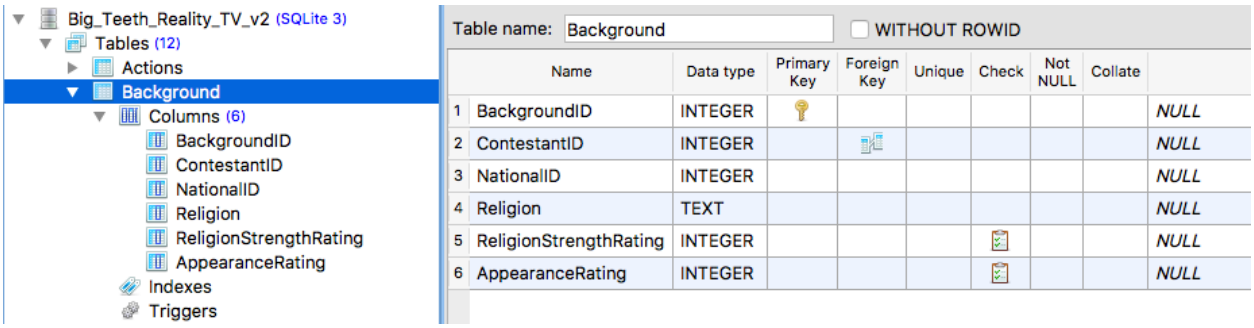
Below are the screenshots of database tables defined in SQLiteStudio database management software for Big Teeth Reality TV show database.

Actions Table



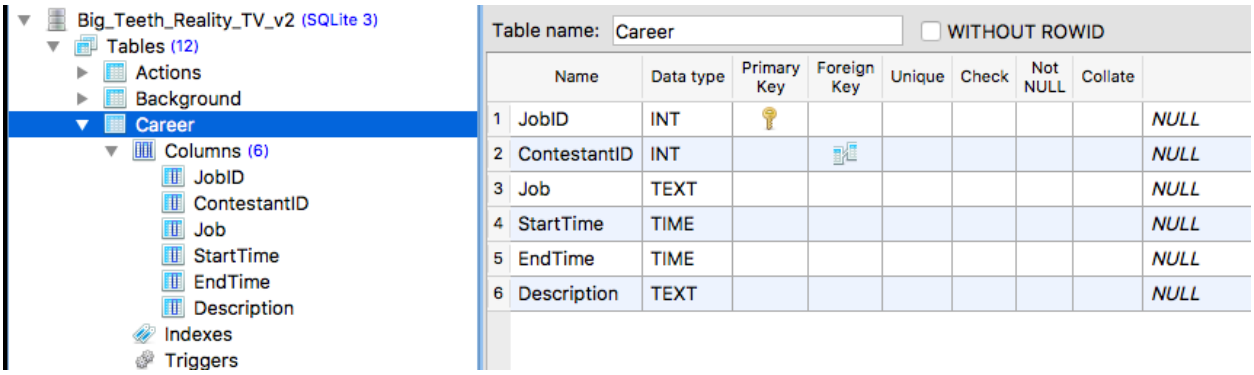
	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	
1	ActionID	INTEGER							NULL
2	EventID	INTEGER							NULL
3	Sequence	INTEGER							NULL
4	Description	TEXT							NULL
5	Cameras	TEXT							NULL
6	EstTime	TIME							NULL

Background Table



	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	
1	BackgroundID	INTEGER							NULL
2	ContestantID	INTEGER							NULL
3	NationalID	INTEGER							NULL
4	Religion	TEXT							NULL
5	ReligionStrengthRating	INTEGER							NULL
6	AppearanceRating	INTEGER							NULL

Career Table



	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	
1	JobID	INT							NULL
2	ContestantID	INT							NULL
3	Job	TEXT							NULL
4	StartTime	TIME							NULL
5	EndTime	TIME							NULL
6	Description	TEXT							NULL

Contestant Table

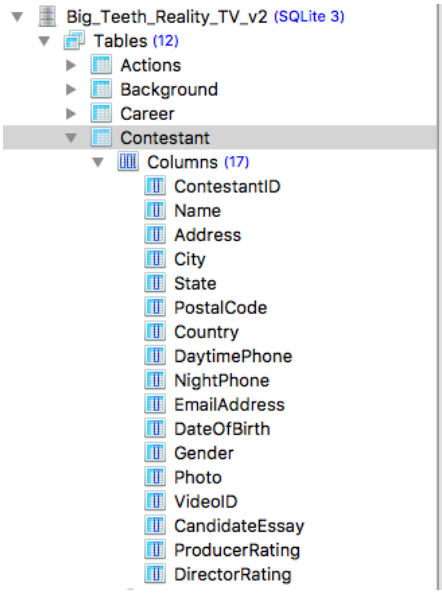


Table name: Contestant WITHOUT ROWID

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	
1	ContestantID	INTEGER							NULL
2	Name	TEXT							NULL
3	Address	TEXT							NULL
4	City	TEXT							NULL
5	State	TEXT							NULL
6	PostalCode	TEXT							NULL
7	Country	TEXT							NULL
8	DaytimePhone	TEXT							NULL
9	NightPhone	TIME							NULL
10	EmailAddress	TEXT							NULL
11	DateOfBirth	DATE							NULL
12	Gender	CHAR							NULL

Type Name

Education Table

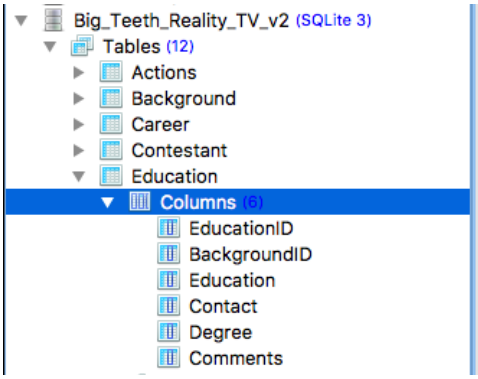


Table name: Education WITHOUT ROWID

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	
1	EducationID	INTEGER							NULL
2	BackgroundID	INTEGER							NULL
3	Education	TEXT							NULL
4	Contact	TEXT							NULL
5	Degree	TEXT							NULL
6	Comments	TEXT							NULL

Employement Table

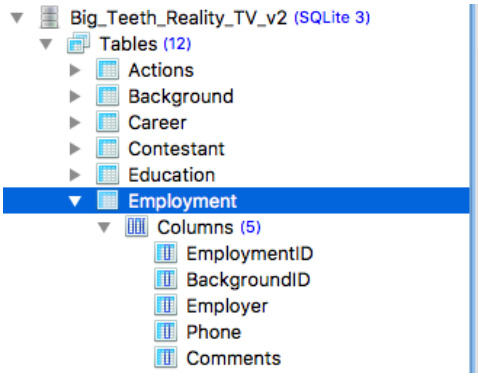
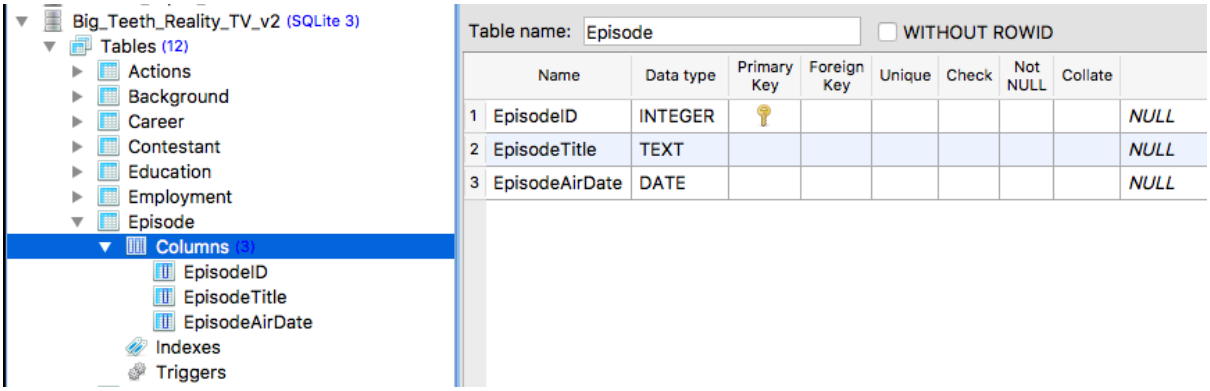


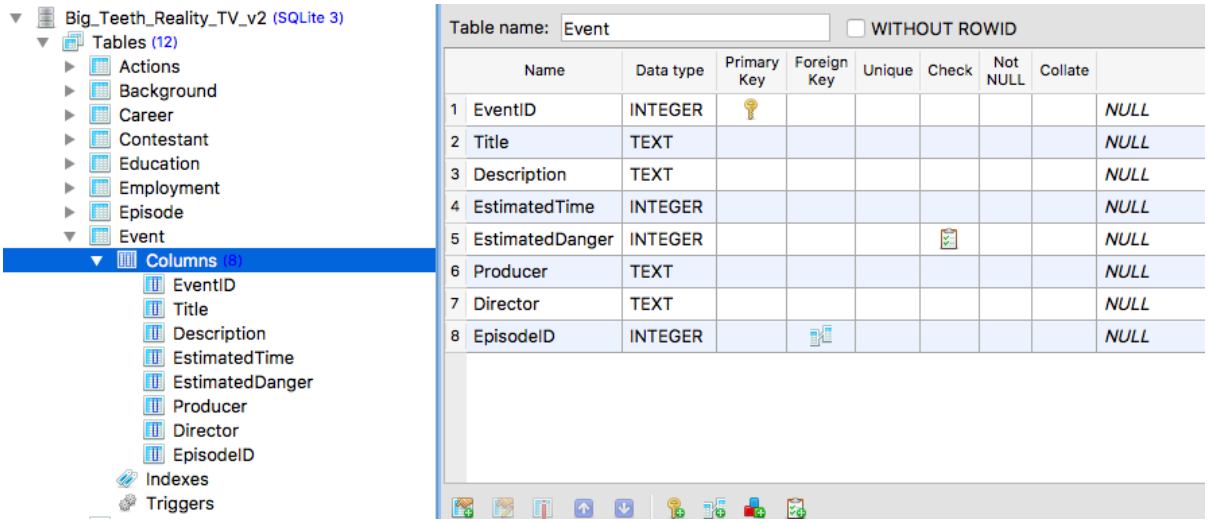
Table name: Employment WITHOUT ROWID

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	
1	EmploymentID	INTEGER							NULL
2	BackgroundID	INTEGER							NULL
3	Employer	TEXT							NULL
4	Phone	TEXT							NULL
5	Comments	TEXT							NULL

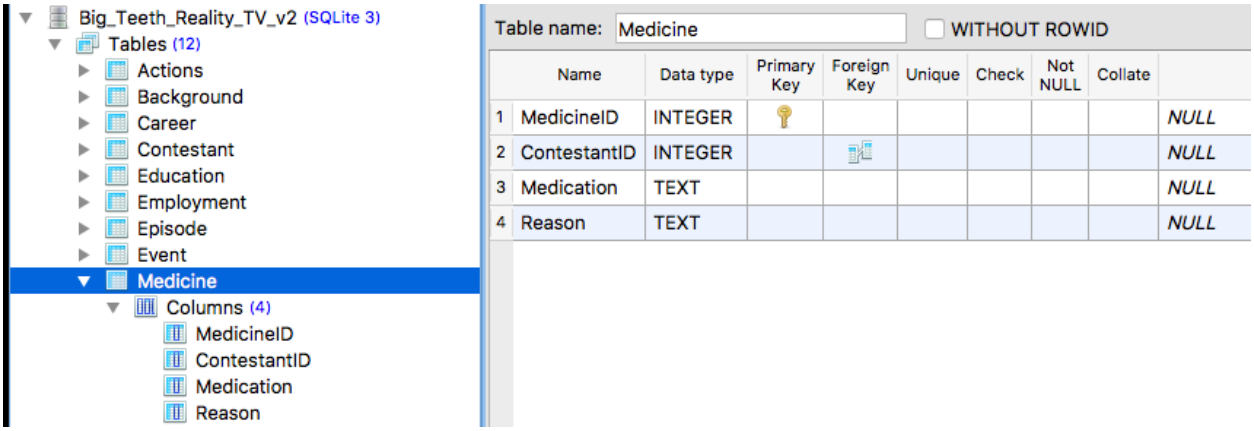
Episode Table



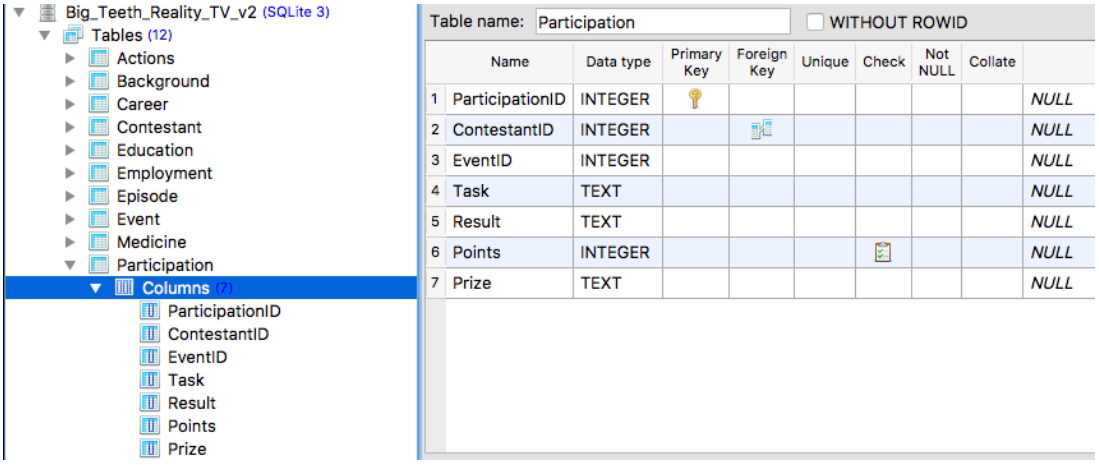
Event Table



Medicine Table

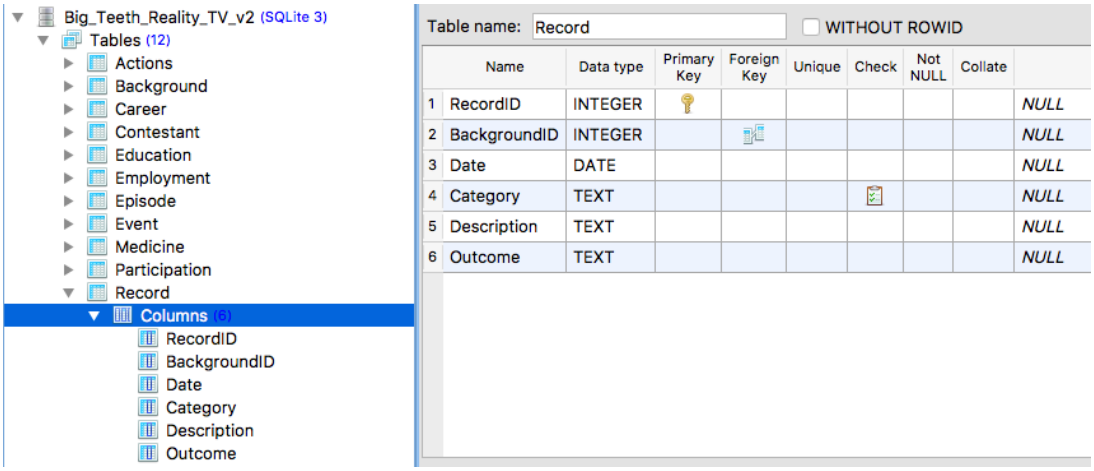


Participation Table



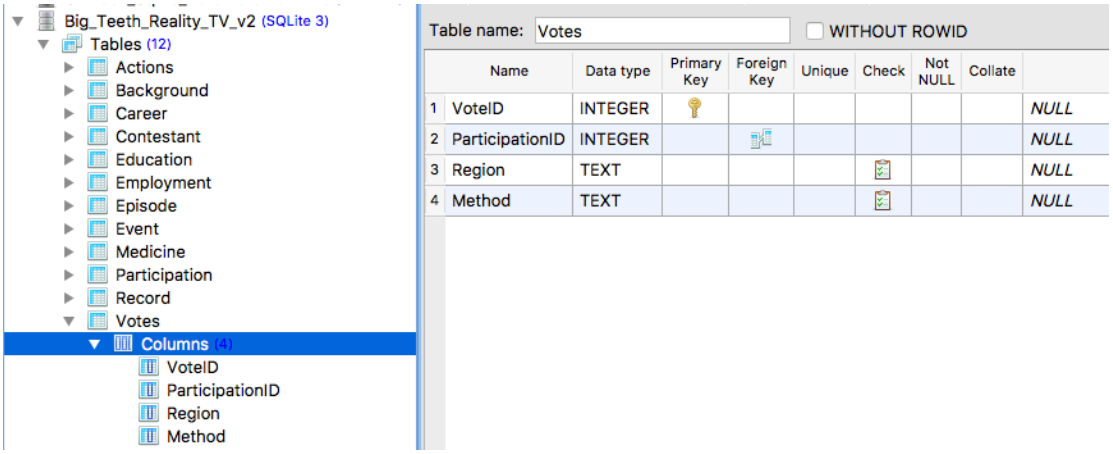
	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	
1	ParticipationID	INTEGER							NULL
2	ContestantID	INTEGER							NULL
3	EventID	INTEGER							NULL
4	Task	TEXT							NULL
5	Result	TEXT							NULL
6	Points	INTEGER							NULL
7	Prize	TEXT							NULL

Record Table



	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	
1	RecordID	INTEGER							NULL
2	BackgroundID	INTEGER							NULL
3	Date	DATE							NULL
4	Category	TEXT							NULL
5	Description	TEXT							NULL
6	Outcome	TEXT							NULL

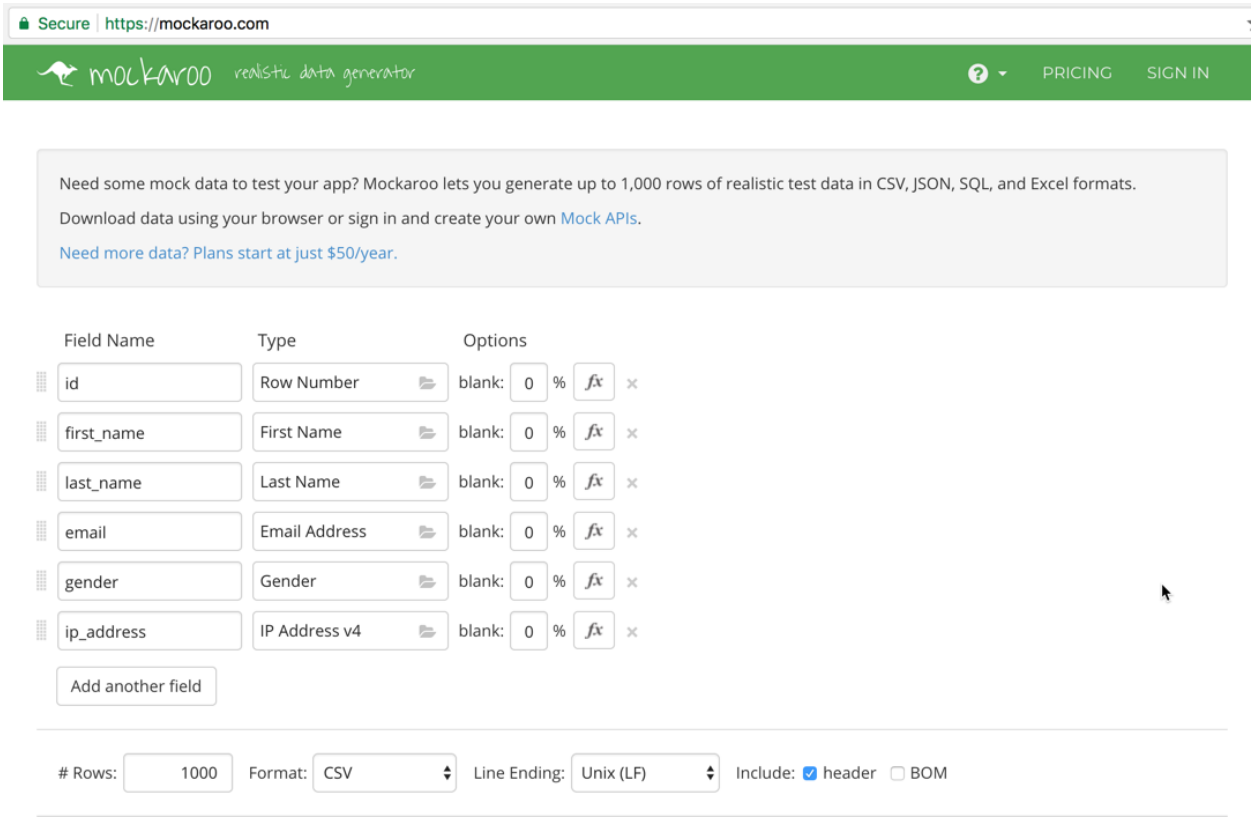
Votes Table



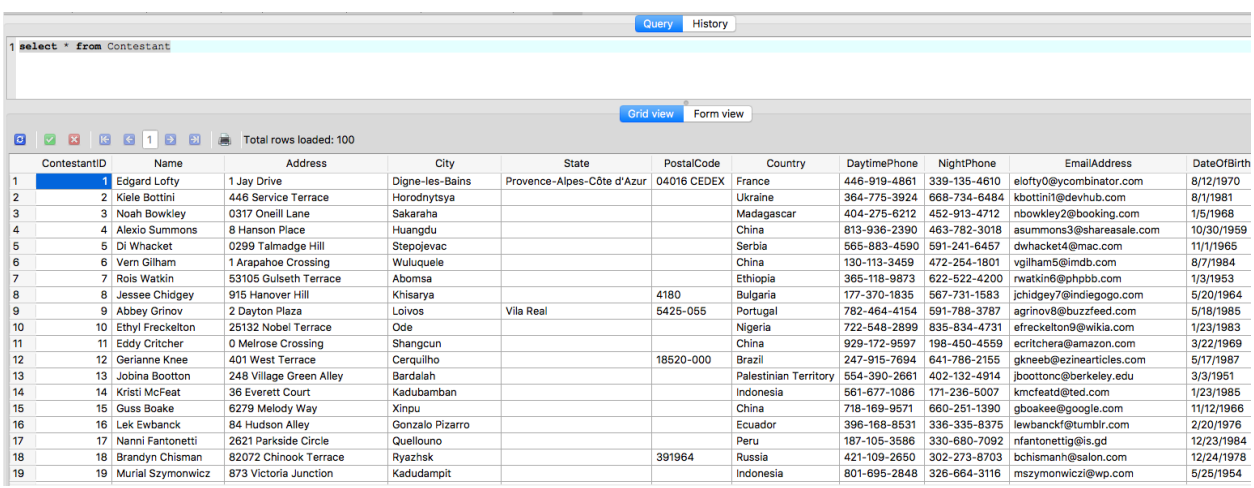
	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	
1	VoteID	INTEGER							NULL
2	ParticipationID	INTEGER							NULL
3	Region	TEXT							NULL
4	Method	TEXT							NULL

Sample Data

Sample data for the proposed database solution is generated using mockaroo.com, an online, realistic test data generation application. The test data generated is provided as .csv files along with the submitted proposed database design specifications document.



A sample generic search query is shown in below screenshot as working proof of concept.



	ContestantID	Name	Address	City	State	PostalCode	Country	DaytimePhone	NightPhone	EmailAddress	DateOfBirth
1	1	Edgard Lofty	1 Jay Drive	Digne-les-Bains	Provence-Alpes-Côte d'Azur	04016 CEDEX	France	446-919-4861	339-135-4610	elofty0@ycombinator.com	8/12/1970
2	2	Kiele Bottini	446 Service Terrace	Horodnytsya			Ukraine	364-775-3924	668-734-6484	kbottini1@devhub.com	8/1/1981
3	3	Noah Bowkley	0317 Oneill Lane	Sakaraha			Madagascar	404-275-6212	452-913-4712	nbowkley2@booking.com	1/5/1968
4	4	Alexio Summons	8 Hanson Place	Huangdu			China	813-936-2390	463-782-3018	asummons3@shareasale.com	10/30/1959
5	5	Di Whacket	0299 Talmadge Hill	Stepojevac			Serbia	565-883-4590	591-241-6457	dwhacket4@mac.com	11/1/1965
6	6	Vern Gilham	1 Arapahoe Crossing	Wuluquele			China	130-113-3459	472-254-1801	vgilham5@imdb.com	8/7/1984
7	7	Rois Watkin	53105 Gulseth Terrace	Abomsa			Ethiopia	365-118-9873	622-522-4200	rwatkin6@phpbb.com	1/3/1953
8	8	Jessee Chidgey	915 Hanover Hill	Khisarya		4180	Bulgaria	177-370-1835	567-731-1583	jchidgey7@indiegogo.com	5/20/1964
9	9	Abbey Grinov	2 Dayton Plaza	Loivos	Vila Real	5425-055	Portugal	782-464-4154	591-788-3787	agrinov8@buzzfeed.com	5/18/1985
10	10	Ethyl Freckelton	25132 Nobel Terrace	Ode			Nigeria	722-548-2899	835-834-4731	efreckelton9@wikia.com	1/23/1983
11	11	Eddy Critcher	0 Melrose Crossing	Shangcun			China	929-172-9597	198-450-4559	ecritchera@amazon.com	3/22/1969
12	12	Gerianne Knee	401 West Terrace	Cerquilho		18520-000	Brazil	247-915-7694	641-786-2155	gkneeb@ezinearticles.com	5/17/1987
13	13	Jobina Bootton	248 Village Green Alley	Bardalah			Palestinian Territory	554-390-2661	402-132-4914	jbootton c@berkeley.edu	3/3/1951
14	14	Kristi McFeat	36 Everett Court	Kadubamban			Indonesia	561-677-1086	171-236-5007	kmcfeatd@ted.com	1/23/1985
15	15	Guss Boake	6279 Melody Way	Xinpu			China	718-169-9571	660-251-1390	gboakee@google.com	11/12/1966
16	16	Lek Ewbanck	84 Hudson Alley	Gonzalo Pizarro			Ecuador	396-168-8531	336-335-8375	lewbanckf@tumblr.com	2/20/1976
17	17	Nanni Fantonetti	2621 Parkside Circle	Quellouno			Peru	187-105-3586	330-680-7092	nfantonettig@is.gd	12/23/1984
18	18	Brandyn Chisman	82072 Chinook Terrace	Ryazhsk		391964	Russia	421-109-2650	302-273-8703	bchismanh@salon.com	12/24/1978
19	19	Murial Szymonwicz	873 Victoria Junction	Kadudampit			Indonesia	801-695-2848	326-664-3116	mszymonwiczi@wp.com	5/25/1954

Requested Database Queries

The average producer and director ratings for a specific event's team members

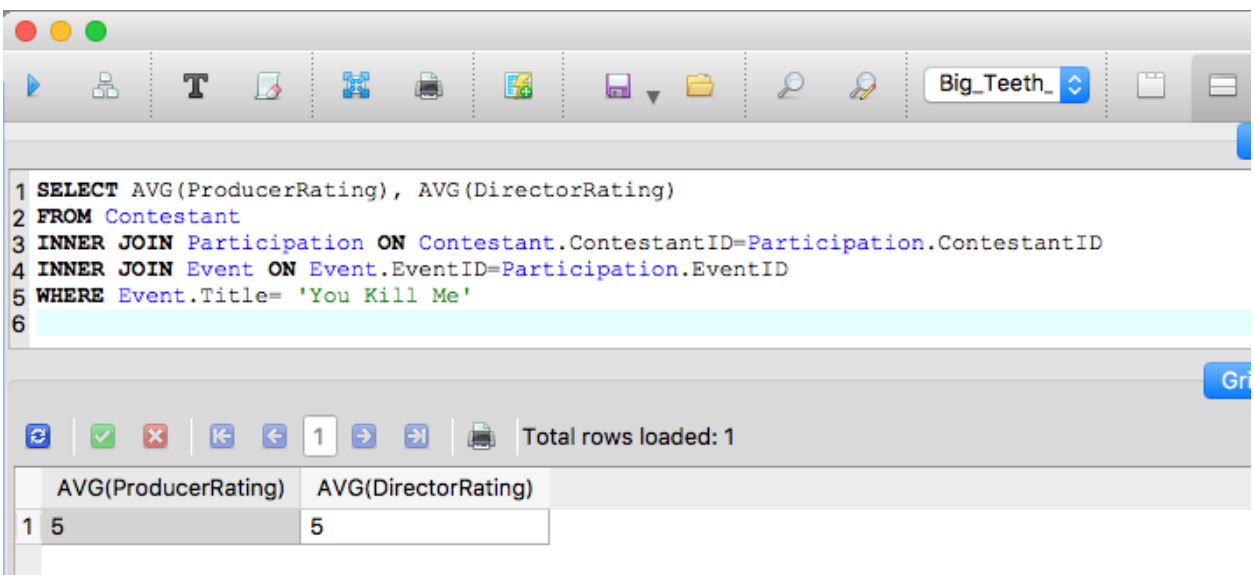
SQL Query:

```
SELECT AVG(ProducerRating), AVG(DirectorRating)
FROM Contestant
INNER JOIN Participation ON Contestant.ContestantID=Participation.ContestantID
INNER JOIN Event ON Event.EventID=Participation.EventID
WHERE Event.Title= 'You Kill Me'
```

Relational Algebra:

$$\Im_{\text{AVG(ProducerRating), AVG(DirectorRating)}} (\sigma_{\text{Event.Title= 'You Kill Me'}} (\text{Contestant} \bowtie_{\text{< ContestantID= ContestantID>}} (\text{Participation}_{\text{<EventID=EventID>}} \text{Event})))$$

Query Output on SQLiteStudio:



Determine which actions will take the longest. Rank all actions from longest to shortest.

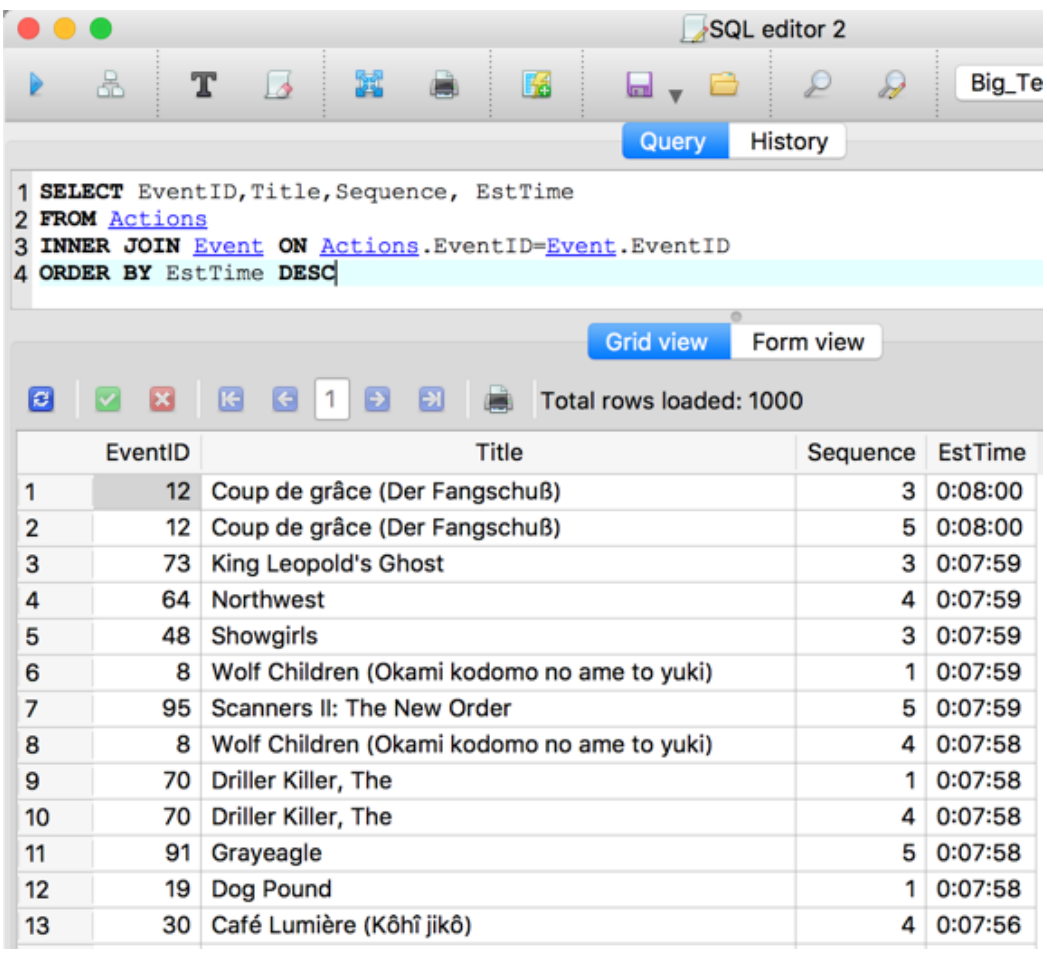
SQL Query:

```
SELECT EventID,Title,Sequence, EstTime
FROM Actions
INNER JOIN Event ON Actions.EventID=Event.EventID
ORDER BY EstTime DESC
```

Relational Algebra:

$$\pi_{\text{EventID,Title,Sequence, EstTime}} (\text{Actions} \bowtie_{\text{<EventID=EventID>}} \text{Event}_{\text{<EventID=EventID>}})$$

Query Output on SQLiteStudio:



	EventID	Title	Sequence	EstTime
1	12	Coup de grâce (Der Fangschuß)	3	0:08:00
2	12	Coup de grâce (Der Fangschuß)	5	0:08:00
3	73	King Leopold's Ghost	3	0:07:59
4	64	Northwest	4	0:07:59
5	48	Showgirls	3	0:07:59
6	8	Wolf Children (Okami kodomo no ame to yuki)	1	0:07:59
7	95	Scanners II: The New Order	5	0:07:59
8	8	Wolf Children (Okami kodomo no ame to yuki)	4	0:07:58
9	70	Driller Killer, The	1	0:07:58
10	70	Driller Killer, The	4	0:07:58
11	91	Grayeagle	5	0:07:58
12	19	Dog Pound	1	0:07:58
13	30	Café Lumière (Kôhî jikô)	4	0:07:56

SQL editor 2

Big_Teeth_

Query | History

```
1 SELECT EventID,Title,Sequence, EstTime
2 FROM Actions
3 INNER JOIN Event ON Actions.EventID=Event.EventID
4 ORDER BY EstTime DESC
```

Grid view | Form view

Total rows loaded: 1000

	EventID	Title	Sequence	EstTime
988	72	Bells Are Ringing	2	0:03:03
989	32	Stroszek	1	0:03:02
990	93	Rescuers Down Under, The	2	0:03:02
991	67	Fear, The	4	0:03:02
992	28	Post Tenebras Lux	1	0:03:02
993	48	Showgirls	4	0:03:02
994	53	Åsa-Nisse - Wälkom to Knohult	4	0:03:01
995	64	Northwest	3	0:03:01
996	24	Bob Saget: That Ain't Right	3	0:03:01
997	88	Restless Souls (Bag det stille ydre)	1	0:03:01
998	25	Special Delivery	2	0:03:00
999	33	Sorrow and the Pity, The (Le chagrin et la pitié)	5	0:03:00
1000	89	Tarzan and the Lost City	4	0:03:00

List each contestant's total votes by region and method for a specific episode and rank this from highest to lowest.

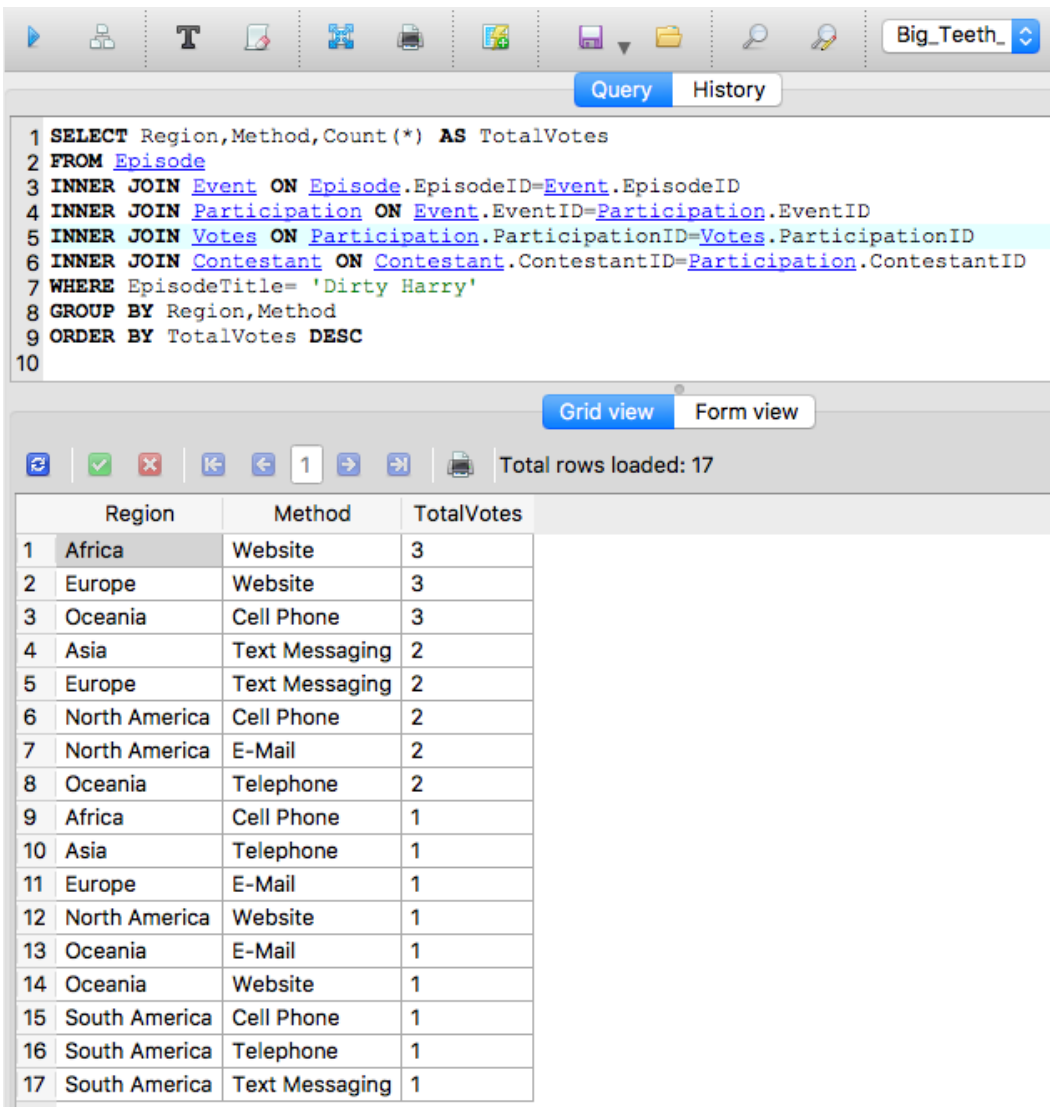
SQL Query:

```
SELECT Region,Method,Count(*) AS TotalVotes
FROM Episode
INNER JOIN Event ON Episode.EpisodeID=Event.EpisodeID
INNER JOIN Participation ON Event.EventID=Participation.EventID
INNER JOIN Votes ON Participation.ParticipationID=Votes.ParticipationID
INNER JOIN Contestant ON Contestant.ContestantID=Participation.ContestantID
WHERE EpisodeTitle= 'Dirty Harry'
GROUP BY Region,Method
ORDER BY TotalVotes DESC
```

Relational Algebra:

$$\text{Region , Method } \Im \text{ Count(*) AS TotalVotes } (\sigma_{\text{ EpisodeTitle= 'Dirty Harry'}} (\text{Episode} \bowtie_{\text{<EpisodeID= EpisodeID>}} (\text{Event} \bowtie_{\text{<EventID=EventID>}} (\text{Participation} \bowtie_{\text{<PartcipationID= PartcipationID >}} (\text{Votes} \bowtie_{\text{< ContestantID= ContestantID >}} \text{Contestant}))))$$

Query Output on SQLiteStudio:



```
SELECT Region,Method,Count(*) AS TotalVotes
FROM Episode
INNER JOIN Event ON Episode.EpisodeID=Event.EpisodeID
INNER JOIN Participation ON Event.EventID=Participation.EventID
INNER JOIN Votes ON Participation.ParticipationID=Votes.ParticipationID
INNER JOIN Contestant ON Contestant.ContestantID=Participation.ContestantID
WHERE EpisodeTitle= 'Dirty Harry'
GROUP BY Region,Method
ORDER BY TotalVotes DESC
```

Total rows loaded: 17

	Region	Method	TotalVotes
1	Africa	Website	3
2	Europe	Website	3
3	Oceania	Cell Phone	3
4	Asia	Text Messaging	2
5	Europe	Text Messaging	2
6	North America	Cell Phone	2
7	North America	E-Mail	2
8	Oceania	Telephone	2
9	Africa	Cell Phone	1
10	Asia	Telephone	1
11	Europe	E-Mail	1
12	North America	Website	1
13	Oceania	E-Mail	1
14	Oceania	Website	1
15	South America	Cell Phone	1
16	South America	Telephone	1
17	South America	Text Messaging	1

Identify which contestants have not participated in any events.

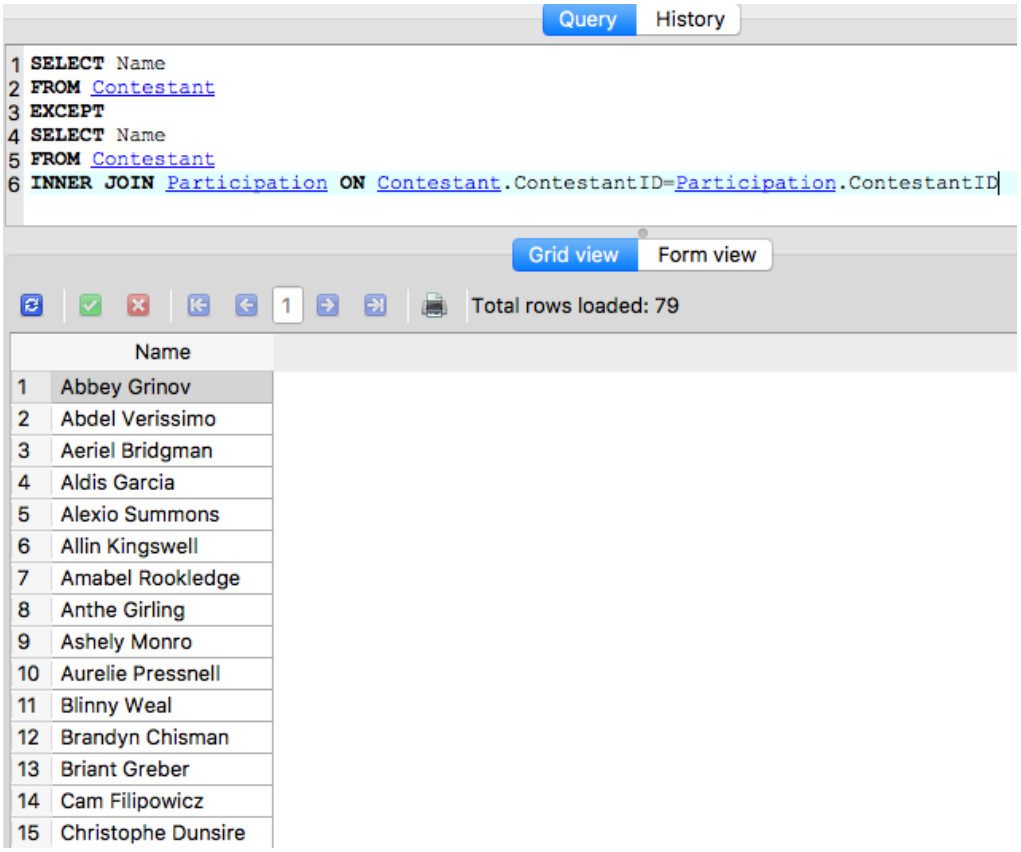
SQL Query:

```
SELECT Name
FROM Contestant
EXCEPT
SELECT Name
FROM Contestant
INNER JOIN Participation ON Contestant.ContestantID=Participation.ContestantID
```

Relational Algebra:

$$\pi_{\text{Name}}(\text{Contestant}) - \pi_{\text{Name}}(\text{Contestant} \bowtie_{\text{ContestantID} = \text{ContestantID}} \text{Participation})$$

Query Output on SQLiteStudio:



Query History

```
1 SELECT Name
2 FROM Contestant
3 EXCEPT
4 SELECT Name
5 FROM Contestant
6 INNER JOIN Participation ON Contestant.ContestantID=Participation.ContestantID
```

Grid view Form view

1 Total rows loaded: 79

	Name
54	Michael Colter
55	Mischa Ary
56	Murial Szymonwicz
57	Nanni Fantonetti
58	Noah Bowkley
59	Pat Trorey
60	Pepita Varns
61	Phylys Fehners
62	Prissie Boyett
63	Rachel Corris
64	Reggy Mcsarry
65	Rois Watkin
66	Rudiger Dennerley
67	Ruthi Spadoni
68	Sascha Ranby
69	Shane Fretwell
70	Steffie Robatham
71	Sue Leverett
72	Troy Ottewill
73	Valentia Stanfield
74	Vern Gilham
75	Vito Fuggles
76	Wally Rassmann
77	Wang Kilbourne
78	Wilow Krzysztof
79	Yvette Santore

What is the highest estimated danger level for any event?

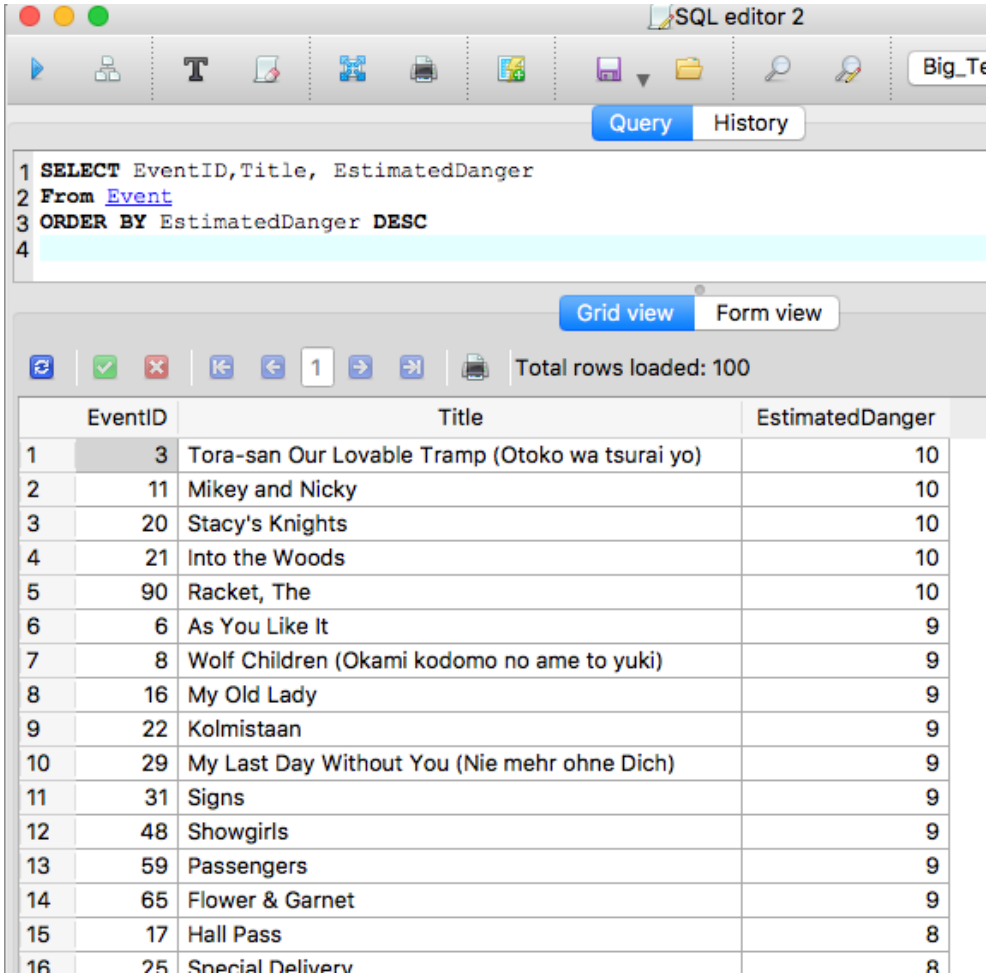
SQL Query:

```
SELECT EventID,Title, EstimatedDanger
From Event
ORDER BY EstimatedDanger DESC
```

Relational Algebra:

$$\pi_{\text{EventID, Title, EsttimatedDanger}} (\text{Event})$$

Query Output on SQLiteStudio:



	EventID	Title	EstimatedDanger
1	3	Tora-san Our Lovable Tramp (Otoko wa tsurai yo)	10
2	11	Mikey and Nicky	10
3	20	Stacy's Knights	10
4	21	Into the Woods	10
5	90	Racket, The	10
6	6	As You Like It	9
7	8	Wolf Children (Okami kodomo no ame to yuki)	9
8	16	My Old Lady	9
9	22	Kolmistaan	9
10	29	My Last Day Without You (Nie mehr ohne Dich)	9
11	31	Signs	9
12	48	Showgirls	9
13	59	Passengers	9
14	65	Flower & Garnet	9
15	17	Hall Pass	8
16	25	Special Delivery	8

Query History

```
1 SELECT EventID,Title, EstimatedDanger
2 From Event
3 ORDER BY EstimatedDanger DESC
4
```

Grid view Form view

1 Total rows loaded: 100

	EventID	Title	EstimatedDanger
73	38	Red Dust	3
74	42	Romulus, My Father	3
75	44	The Invisible Boy	3
76	52	Clean, Shaven	3
77	54	Caught Inside	3
78	67	Fear, The	3
79	73	King Leopold's Ghost	3
80	85	The End of the Tour	3
81	94	Courtship of Eddie's Father, The	3
82	100	Hopscotch	3
83	12	Coup de grâce (Der Fangschuß)	2
84	35	Among Wolves (Entrelobos)	2
85	36	Criminal, The (a.k.a. Concrete Jungle)	2
86	45	貞子3D	2
87	51	Blank Check	2
88	60	Tetsuo II: Body Hammer	2
89	61	Soupe aux Choux, La	2
90	64	Northwest	2
91	9	Big One, The	1
92	13	Scoop	1
93	30	Café Lumière (Kôhî jikô)	1
94	37	Upstream Color	1
95	41	Love	1
96	57	Doodlebug	1
97	71	Covenant, The	1
98	86	Joy of Sex	1
99	89	Tarzan and the Lost City	1
100	96	Lathe of Heaven, The	1

Show a relational expression for any one query

Relational expressions are provided for each query above.

Technical Notes on Database Design

In this section, we are going to provide a summary information regarding the underlying reasoning of Big Teeth Reality TV Show database.

- Entity Relationship Diagram designed with business end users' easy understanding of the database structure.

- While it could be possible to have less number of tables in the database, we decided to segregate the information in separate tables for easy readability and retrieval of information with simpler SQL statements. For example, our proposed database structure could have Background table embedded in Contestant table and Episode table could be embedded in Votes table. However, we decided to implement our present entity relationship solution, mainly to let business end users with little or no database/SQL experience to get up to speed and productive in shorter time.

- For large databases in order of gigabytes, our present database structure could have encounter some performance issues, however, for the given and projected size of Big Teeth Reality TV Show database, we decided to trade off optimized storage concern with more readable and understandable database with simpler queries.

- We know that every episode has some events and every event has some contestants with different participationID values. Required business solution asks for vote contestants in episode but in our ERD, people vote contestants in event. It means that our voting system is more accurate than problem definition.

Database Team: Team STU

This proposed database solution came to life with perfectly balanced and coordinated efforts of Mr. Swogger, Mr. Taheri, and Mr. Ugur. While everyone equally worked on developing the presented database solution, special highlights for the team members could be summarized as follows:

- Ali Taheri: Finding alternative and easier tools and coming up with relational algebraic expressions for required database queries.

- Matt Swogger: Matt's current software development experience was instrumental in deciding on the final design decisions of the presented database structure and ERD.

- Arda Ugur: Arda's extensive technical writing experience made it possible to document the team's technical work and presenting a readable business proposal document. Arda

also has experience with SQL and Oracle DBS which made him an important member of the team for verifying the proposed solutions and illustrated database queries.

Database Tools

Database Vendor:

SQLite: https://www.sqlite.org/

Relational Database Management Software:

SQLiteStudio: https://sqlitestudio.pl/index.rvt

Database Data Modeling:

Navicat Data Modeler: https://www.navicat.com/

Test Data Generation:

Mockaroo: https://mockaroo.com/