DDL commands：

```
CREATE SCHMA `JJYY_1`;
CREATE TABLE `JJYY_1`.Actor (
	actorID INTEGER NOT NULL UNIQUE AUTO_INCREMENT,
	name	VARCHAR(40) NOT NULL,
	birthYear	INT,
	PRIMARY KEY(actorID)
);
CREATE TABLE `JJYY_1`.Director (
	directorID	INTEGER NOT NULL UNIQUE AUTO_INCREMENT,
	name	VARCHAR(40),
	birthYear	INT,
	PRIMARY KEY(directorID)
);
CREATE TABLE `JJYY_1`.Movie (
	movieID	INTEGER NOT NULL UNIQUE AUTO_INCREMENT,
	name	VARCHAR(255) NOT NULL,
	releaseDate	VARCHAR(15),
	duration	INTEGER NOT NULL,
	genre	VARCHAR(50),
	ratingFromTomato	FLOAT,
	contentRating VARCHAR(20),
	ratingFromIMDB	FLOAT,
	directorID	INTEGER NOT NULL,
	PRIMARY KEY(movieID),
	FOREIGN KEY(directorID) REFERENCES Director(directorID) ON UPDATE CASCADE
);
CREATE TABLE `JJYY_1`.Act (
	actorID INTEGER NOT NULL,
	movieID	INTEGER NOT NULL,
	roleName	VARCHAR(255) NOT NULL,
```

```
	PRIMARY KEY(actorID, movieID),
	FOREIGN KEY(movieID) REFERENCES Movie(movieID) ON UPDATE CASCADE,
	FOREIGN KEY(actorID) REFERENCES Actor(actorID) ON UPDATE CASCADE
);
CREATE TABLE `JJYY_1`.User (
	userID  INTEGER NOT NULL UNIQUE AUTO_INCREMENT,
	userName	VARCHAR(20) NOT NULL,
	password	VARCHAR(20) NOT NULL,
	preferredGenresVARCHAR(255),
	email	VARCHAR(50),
	phoneNumber	INTEGER,
	adultStatus	BOOL,
	PRIMARY KEY(userID)
);
CREATE TABLE `JJYY_1`.Review (
	reviewID	INTEGER NOT NULL UNIQUE AUTO_INCREMENT,
	source	VARCHAR(20) NOT NULL,
   rating INTEGER NOT NULL,
	movieID  INTEGER NOT NULL,
	content TEXT,
	userID  INTEGER NOT NULL,
	PRIMARY KEY(reviewID),
	FOREIGN KEY(userID) REFERENCES User(userID) ON UPDATE CASCADE ON DELETE
CASCADE,
	FOREIGN KEY(movieID) REFERENCES Movie(movieID) ON UPDATE CASCADE ON
DELETE CASCADE
);
CREATE TABLE `JJYY_1`.Watch (
	movieID	INTEGER NOT NULL,
	userID  INTEGER NOT NULL,
	PRIMARY KEY(movieID, userID),
```

FOREIGN KEY(movieID) REFERENCES Movie(movieID) ON UPDATE CASCADE ON DELETE CASCADE,

FOREIGN KEY(userID) REFERENCES User(userID) ON UPDATE CASCADE ON DELETE CASCADE

);

Table:

```
mysql> show tables;
+------------------+
| Tables_in_JJYY_1 |
+------------------+
| Act              |
| Actor            |
| Director         |
| Movie            |
| Review           |
| User             |
| Watch            |
+------------------+
7 rows in set (0.00 sec)
```

Count row number：

Review：

```
mysql> select count(reviewID) as rowNum from Review;
+--------+
| rowNum |
+--------+
|  83121 |
+--------+
```

Act：

```
mysql> select count(actorID) as rowNum from Act;
+--------+
| rowNum |
+--------+
|   4888 |
+--------+
```

Actor：

```
mysql> select count(actorID) as rowNum from Actor;
+--------+
| rowNum |
+--------+
|   4091 |
+--------+
```

Director：

```
mysql> select count(directorID) as rowNum from Director;
+--------+
| rowNum |
+--------+
|   1305 |
+--------+
```

Movie：

```
mysql> select count(movieID) as rowNum from Movie;
+--------+
| rowNum |
+--------+
|   1387 |
+--------+
```

Watch：

```
mysql> select count(userID) as rowNum from Watch;
+--------+
| rowNum |
+--------+
|  83121 |
+--------+
```

User：

```
mysql> select count(userID) as rowNum from User;
+--------+
| rowNum |
+--------+
|   2000 |
+--------+
```