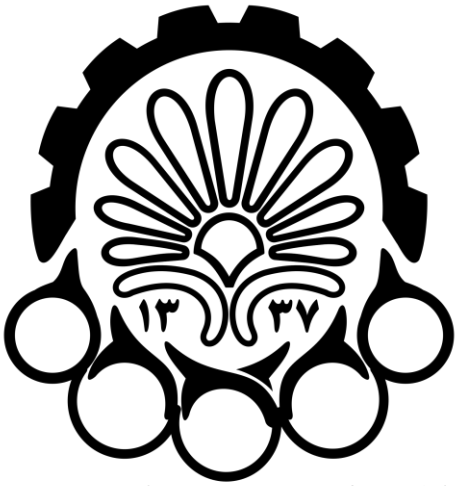


دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

گزارش پروژه

سیاوش کاوسی ۹۲۳۱۰۴۸

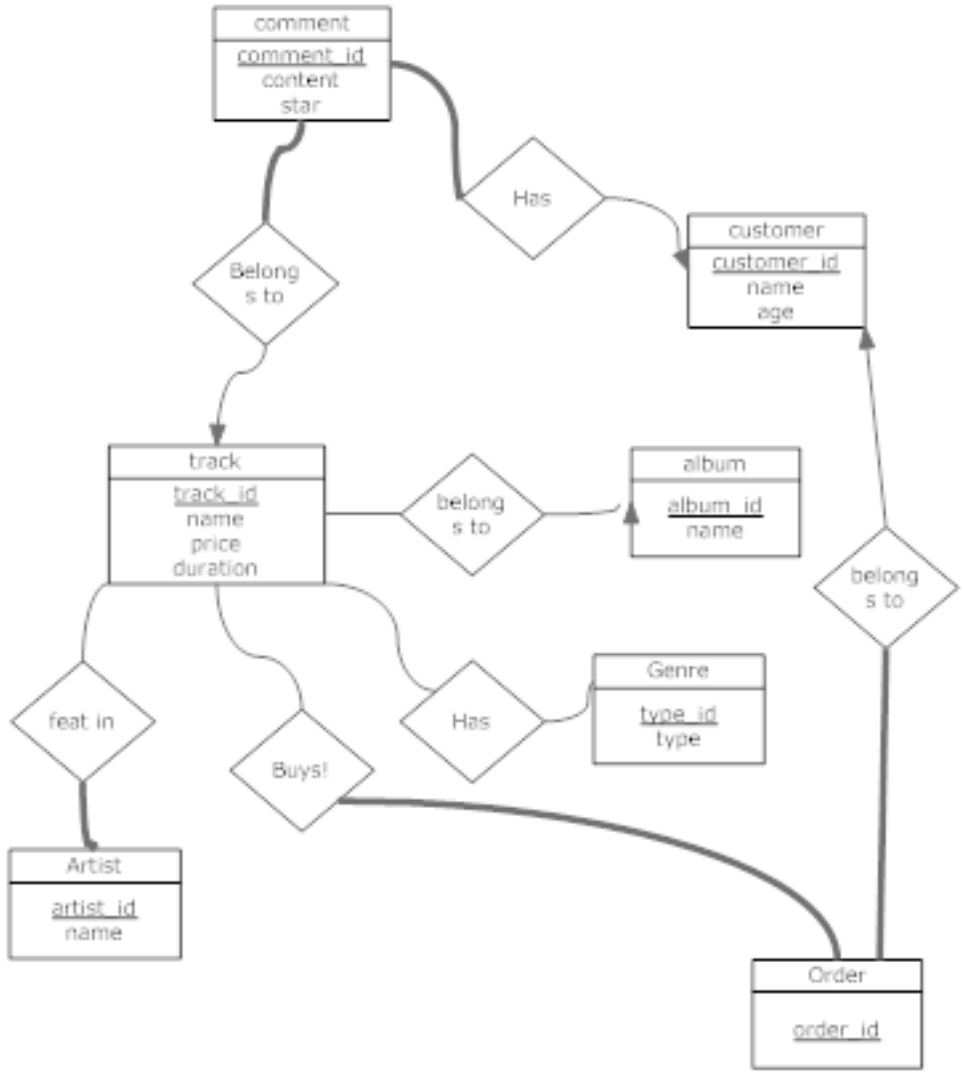
استاد: دکتر شهریاری

اردیبهشت ۱۳۹۵

گزارش کار

جداول ERD طراحی شده برای این پایگاه داده ☺

نکته: در شکل زیر خط های پر رنگ به جای دوخط استفاده شده است (total participation)



و پس از تبدیل آن به مدل رابطه ای، با استفاده از زبان sql آنرا پیاده سازی کرده ام

```sql
CREATE TABLE album (
  album_id INT PRIMARY KEY,
  name     VARCHAR(32) NOT NULL
);

CREATE TABLE track (
  track_id INT PRIMARY KEY,
  album_id INT,
  name     VARCHAR(32) NOT NULL,
  price    INT         NOT NULL,
  duration TIME,
```

```
  FOREIGN KEY (album_id) REFERENCES album (album_id)
);

CREATE TABLE artist (
  artist_id INT PRIMARY KEY,
  name      VARCHAR(32) NOT NULL
);

CREATE TABLE track_artist (
  track_id  INT,
  artist_id INT NOT NULL,
  FOREIGN KEY (track_id) REFERENCES track (track_id),
  FOREIGN KEY (artist_id) REFERENCES artist (artist_id),
  PRIMARY KEY (track_id, artist_id)
);

CREATE TABLE customer (
  customer_id INT PRIMARY KEY,
  name        VARCHAR(32) NOT NULL,
  age         INT
);

CREATE TABLE comment (
  comment_id  INT PRIMARY KEY,
  track_id    INT          NOT NULL,
  customer_id INT          NOT NULL,
  content     VARCHAR(255) NOT NULL,
  star        INT          NOT NULL,
  FOREIGN KEY (track_id) REFERENCES track (track_id),
  FOREIGN KEY (customer_id) REFERENCES customer (customer_id)
);

CREATE TABLE genre (
  type_id INT PRIMARY KEY,
  type    VARCHAR(20) NOT NULL
);

CREATE TABLE track_genre (
  track_id INT NOT NULL,
  type_id  INT NOT NULL,
  FOREIGN KEY (track_id) REFERENCES track (track_id),
  FOREIGN KEY (type_id) REFERENCES genre (type_id),
  PRIMARY KEY (track_id, type_id)
);

CREATE TABLE music_store.order (
  order_id    INT PRIMARY KEY,
  customer_id INT NOT NULL
);

CREATE TABLE track_order (
  track_id INT NOT NULL,
  order_id INT,
  FOREIGN KEY (track_id) REFERENCES track (track_id),
  FOREIGN KEY (order_id) REFERENCES music_store.order (order_id),
  PRIMARY KEY (track_id, order_id)
);
```

پس از طراحی پایگاه داده به سراغ پرس و جوها می رویم

.1

```
SELECT
  album_id,
  sum(price)
FROM track
  NATURAL JOIN track_genre
  NATURAL JOIN genre
WHERE type = 'Pop'
GROUP BY album_id
HAVING sum(price)
ORDER BY sum(price);
```

album_id	`sum(price)`
3	1200
1	7500

.2

```
SELECT
  track_id,
  track.name,
  avg(star)
FROM (track
  NATURAL JOIN track_genre
  NATURAL JOIN genre) JOIN comment USING (track_id)
  JOIN customer USING (customer_id)
WHERE type = 'Rock'
GROUP BY track_id, track.name
HAVING avg(star)
ORDER BY avg(star) DESC;
```

	track_id	name	`avg(star)`
1	9	A Line In The Sand	9.5000
2	7	Guilty All the Same	3.0000
3	8	Drawbar	2.0000

.3

```
SELECT DISTINCT name
FROM customer
  NATURAL JOIN comment
WHERE track_id = (SELECT track_id
                  FROM comment
                  GROUP BY track_id
                  HAVING 9 <= min(star));
```

	name
1	Siavash
2	Ali
3	Goolakh

.4

```
SELECT
  artist.name,
  count(DISTINCT type)
FROM (artist
  JOIN track_artist USING (artist_id)
  JOIN track USING (track_id)) JOIN track_genre USING (track_id)
  JOIN genre USING (type_id)
GROUP BY artist_id, artist.name
HAVING count(DISTINCT type)
ORDER BY artist_id;
```

	name	`count(DISTINCT type)`
1	Moein	1
2	The Piano Guys	2
3	Secret Garden	1
4	Linkin Park	1
5	Backfield	2
6	Aziz Veisi	1
7	Kazem	1
8	Johnson	1

.5

```
SELECT artist.name
FROM artist
WHERE artist_id NOT IN
      (SELECT artist_id
       FROM artist
         JOIN track_artist USING (artist_id)
         JOIN track USING (track_id)
       WHERE album_id IS NULL);
```

	name
1	Moein
2	The Piano Guys
3	Secret Garden
4	Linkin Park
5	Aziz Veisi
6	Kazem
7	Johnson

.6

```
SELECT artist.name
FROM artist
  JOIN track_artist USING (artist_id)
  JOIN
  ((SELECT track_id
    FROM (artist
      JOIN track_artist USING (artist_id)
```

```
8.        JOIN track USING (track_id))
9.      GROUP BY track_id
10.     HAVING count(artist_id) > 2) AS t) USING (track_id);
```

	name
1	Backfield
2	Aziz Veisi
3	Kazem
4	Johnson

.7

```
1.  SELECT
2.    type,
3.    count(order_id) AS num_of_orders
4.  FROM (`order`
5.    NATURAL JOIN track_order
6.    NATURAL JOIN track) NATURAL JOIN track_genre
7.    NATURAL JOIN genre
8.  GROUP BY type
9.  HAVING count(order_id);
```

	type	num_of_orders
1	Metal	1
2	Pop	4
3	Rock	4

داده های اضافه شده به این پایگاه داده

```
1.  USE music_store;
2.
3.  INSERT INTO album (album_id, name) VALUES
4.    (1, 'Tolou'), (2, 'Once In A Red Moon'), (3, 'The Piano Guys'), (4, 'The Hunting Part
    y'),
5.    (5, 'Shalgham');
6.
7.  INSERT INTO track (track_id, album_id, name, price, duration) VALUES
8.    (1, 1, 'Naajee', 3000, '430'), (2, 1, 'Molaghat', 4500, '530'),
9.    (3, 2, 'Gates of Dawn', 8000, '429'), (4, 3, 'A Thousand Years', 5000, '436'),
10.   (5, 3, 'Let It Go', 1200, '402'), (6, 3, 'Summer Jam', 6700, '354'),
11.   (7, 4, 'Guilty All the Same', 12000, '554'), (8, 4, 'Drawbar', 9200, '312'),
12.   (9, 4, 'A Line In The Sand', 9900, '638'), (10, NULL, 'Backfield', 4300, '407'),
13.   (11, 5, 'To Azizi', 500, '433');
14.
15. INSERT INTO artist (artist_id, name) VALUES (1, 'Moein'), (2, 'The Piano Guys'),
16.   (3, 'Secret Garden'), (4, 'Linkin Park'), (5, 'Backfield'), (6, 'Aziz Veisi'),
17.   (7, 'Kazem'), (8, 'Johnson');
18.
19. INSERT INTO track_artist (track_id, artist_id) VALUES (1, 1), (2, 1), (3, 3), (4, 2),
20.   (5, 2), (6, 2), (7, 4), (8, 4), (9, 4), (10, 5), (11, 6), (11, 7), (11, 8), (11, 5);
21.
22. INSERT INTO customer (customer_id, name, age) VALUES (1, 'Siavash', 22),
23.   (2, 'Ali', 20), (3, 'Ghasemi', 21), (4, 'Goolakh', 33);
24.
```

```sql
INSERT INTO genre (type_id, type) VALUES (1, 'Pop'), (2, 'Jazz'), (3, 'Instrumental'),
  (4, 'Rock'), (5, 'Hip Hop'), (6, 'Metal'), (7, 'Kordi');

INSERT INTO track_genre (track_id, type_id) VALUES (1, 1), (2, 1), (3, 3), (4, 3),
  (5, 1), (6, 3), (7, 4), (8, 4), (9, 4), (10, 6), (11, 7);

INSERT INTO comment (comment_id, track_id, customer_id, content, star) VALUES
  (1, 2, 1, 'Alie in ahang', 5), (2, 6, 2, 'Bass khoobi dare', 2),
  (3, 7, 1, 'Excellent', 5), (4, 8, 3, 'WTF', 1), (5, 8, 2, 'Good :)', 3),
  (6, 7, 3, 'Terrible', 1), (7, 9, 1, 'Love it', 9.5),
  (8, 9, 1, 'Fascinating', 10), (9, 9, 2, 'Wow', 9), (10, 9, 4, 'Lol', 9);

INSERT INTO `order` (order_id, customer_id) VALUES
  (1, 1), (2, 1), (3, 1), (4, 5), (5, 1), (6, 2);

INSERT INTO track_order (track_id, order_id) VALUES
  (1, 1), (1, 2), (8, 2), (2, 3), (5, 4), (8, 5), (9, 5), (10, 5), (8, 6);
```