Jamal Azad
05/11/2022

SQL Assignment 3

5.3.1: What are all the records in the store table, ordered by the primary key of the table?

QUERY: dvdrental=# **SELECT * FROM store order by store_id;**

***In this table (store), store_id is the Primary Key.**

To check primary key of the table (store), write Command: dvdrental=# \d store

Successfully ran. Total query runtime: 409 msec. 2 rows affected.

```
 store_id | manager_staff_id | address_id |     last_update
----------+------------------+------------+---------------------
        1 |                1 |          1 | 2006-02-15 09:57:12
        2 |                2 |          2 | 2006-02-15 09:57:12
(2 rows)
```

5.3.2: What are the first 10 records in the inventory table in ascending order of update (show just the primary key and the update timestamp)?

QUERY: dvdrental=# **SELECT inventory_id, last_update FROM inventory order by last_update ASC limit 10;**

***In this table (inventory), inventory_id is the Primary Key.**

To check primary key of the table (inventory), write Command: dvdrental=# \d inventory

Successfully ran. Total query runtime: 391 msec. 10 rows affected.

```
 inventory_id |     last_update
--------------+---------------------
            2 | 2006-02-15 10:09:17
            3 | 2006-02-15 10:09:17
            4 | 2006-02-15 10:09:17
            5 | 2006-02-15 10:09:17
            6 | 2006-02-15 10:09:17
            7 | 2006-02-15 10:09:17
            8 | 2006-02-15 10:09:17
            9 | 2006-02-15 10:09:17
           10 | 2006-02-15 10:09:17
            1 | 2006-02-15 10:09:17
(10 rows)
```

5.3.3: What is the street address, district, city name, and phone number for all addresses in the 18743 postal code?

QUERY: dvdrental=# select a.address as "Street Address", a.district, x.city, a.phone number
dvdrental-# from address as a
dvdrental-# inner join city as x on x.city_id = a.city_id
dvdrental-# where a.postal_code = '18743';

*** Here, I'm using the < INNER JOIN > sql constructor between two tables ('address' and 'city') based < ON > the 'city_id' column which happens to be a common column to build the relation.**

Successfully ran. Total query runtime: 263 msec. 1 row affected.

```
       Street Address        | district |  city  |    number
-----------------------------------+----------+--------+--------------
 1795 Santiago de Compostela Way | Texas    | Laredo | 860452626434
(1 row)
```

5.3.4: How many records are in the inventory table?
QUERY: dvdrental=# select count(*) from inventory;
Successfully ran. Total query runtime: 47 msec. 1 row affected.
Display count 4581.

```
count
-------
  4581
(1 row)
```

5.3.5: What is the timestamp of the most recent update to the staff table?
QUERY: dvdrental=# select last_update from staff order by last_update desc limit 1;
Successfully ran. Total query runtime: 127 msec. 1 row affected.

```
       last_update
---------------------------
 2006-05-16 16:13:11.79328
(1 row)
```

5.3.6: How many distinct values of country_id exist in the city table?
QUERY (1): dvdrental=# select distinct count(country_id) from city where country_id is not NULL;
Worked: Displayed 1 row showing, there are 600 distinct values in the city table.

```
count
-------
  600
(1 row)
```

QUERY (2): dvdrental=# select distinct count(country_id) from city;
Worked: Displayed 1 row showing, there are 600 distinct values in the city table.

```
 count
-------
   600
(1 row)
```

5.3.7: How many actors' last names begin with either a "B" or a "C"?

QUERY: dvdrental=# select count(last_name) from actor where last_name like 'B%' or last_name like 'C%';
Worked: Displayed 1 row showing, there are 37 last_name beginning with either “B” or “C” in the actor table.

```
 count
-------
    37
(1 row)
```

5.3.8: What is the average payment amount (rounded to the nearest cent) and number of payments made in the month of April, 2007, by customers that are currently active, and whose average payment is at least $5.00?

QUERY:
```
dvdrental=# select a.customer_id, b.active,
dvdrental=# ROUND(avg(a.amount),2), count(*)
dvdrental=# from payment AS a
dvdrental=# inner join customer AS b ON a.customer_id=b.customer_id
dvdrental=# where cast(a.payment_date AS DATE)
dvdrental=# between '2007-04-01' and '2007-04-30'
dvdrental=# and b.active=1
dvdrental=# group by a.customer_id, b.active
dvdrental=# having avg(a.amount) >= 5.00;
```
Successfully ran. Total query runtime: 926 msec. 79 rows affected.

5.3.9: Show customer's first and last name and email, and label the average payment field as "Avg. Payment Amount" and the number of payments field as "Number of Payments".

QUERY:
```
dvdrental=# select a.first_name, a.last_name, a.email, avg(b.amount) as "Avg. Payment Amount",
dvdrental=# count(b.amount) as "Number of Payments"
dvdrental=# FROM customer as a
dvdrental=# inner join payment as b ON a.customer_id=b.customer_id
dvdrental=# group by a.customer_id, b.amount
```
Successfully ran. Total query runtime: 259 msec. 4677 rows affected.

5.3.10: Order the results by highest payment first, and then by last name.

QUERY:
```
dvdrental=# select a.amount, b.last_name
dvdrental=# from payment as a
dvdrental=# inner join customer as b ON a.customer_id = b.customer_id
dvdrental=# group by  a.amount, b.last_name
dvdrental=# order by a.amount desc;
```

5.3.11: What are all the first names of both the actors and the staff members that begin with the letter 'S' (in alphabetical order)?

QUERY:
```
dvdrental=# select first_name from actor
dvdrental=# where first_name like 'S%'
dvdrental=# UNION ALL
dvdrental=# select first_name from staff
dvdrental=# where first_name like 'S%'
dvdrental=# order by first_name asc
```
Successfully ran. Total query runtime: 62 msec. 14 rows affected.

5.3.12: Show the title of the films that Jon rented on May 29, 2005 in common with those that Mike rented on June 20, 2005. Order by film title.

QUERY:
```
dvdrental=# select f.title from film as f
dvdrental=# inner join inventory as i on i.film_id=f.film_id
dvdrental=# inner join rental as r on r.inventory_id=i.inventory_id
dvdrental=# inner join customer as c on c.customer_id=r.customer_id
dvdrental=# where c.first_name='Jon' and cast (r.rental_date as
DATE)='2005-05-29'
dvdrental=# union all
dvdrental=# select f.title from film as f
dvdrental=# inner join inventory as i on i.film_id=f.film_id
dvdrental=# inner join rental as r on r.inventory_id=i.inventory_id
dvdrental=# inner join customer as c on c.customer_id=r.customer_id
dvdrental=# where c.first_name='Mike' and cast (r.rental_date as
DATE)='2005-06-20'
```
Successfully ran. Total query runtime: 54 msec. 2 rows affected.

```
 title
------------------
 Boogie Amelie
 Camelot Vacation
(2 rows)
```