

تسک دیجی شهر

شایان صادقی

۱۳ بهمن ۱۴۰۲

۱ بخش اول

در این بخش نیاز به اجرای چند کوئری به زبان sql وجود داشت. از طرفی فایل داده‌های موجود با پسوند .db ارائه شده بودند. بنابراین برای دسترسی به محتویات دیتابیس sqlite از یک ابزار کلاینت دیتابیس مناسب مانند dbaver استفاده شد.

۱.۱ سوال ۱

کوئری مربوطه در شکل ۱ و نتایج در شکل ۲ نشان داده شده است. در بررسی اولیه داده‌ها، وجود دو واحد پول مشخص شد که برای همین منظور لازم است مقدار sales در داده‌ها یکسان شود. مقدار تبدیل نشده برای مقایسه در ستونی مجزا مشخص شده است. با توجه به رکورد ردیف ۴ می‌توان تأثیر بی‌توجهی به نرخ تبدیل ارز را مشاهده کرد.

```
SELECT usr.id, usr.registeredDate, usr.country, SUM(ord.sales) as total_sales_no_exchange, SUM(
CASE
when currency='sek' then ord.sales*0.09
else ord.sales
END) as total_sales
from users usr
JOIN orders ord
ON usr.id = ord.userId
WHERE STRFTIME('%Y',usr.registeredDate)="2023"
GROUP BY usr.id, usr.registeredDate, usr.country
ORDER BY total_sales DESC
LIMIT 10;
```

شکل ۱: کوئری ۱

	123 id	123 registeredDate	123 country	total sales no exchange	total sales
1	6,715,274,565,762,698,009	2023-03-11 16:57:11	FI	23,560	23,560
2	1,065,583,221,749,642,853	2023-01-06 11:58:47	FI	23,023	23,023
3	7,763,147,301,647,778,346	2023-01-11 21:43:33	FI	22,270	22,270
4	1,909,500,535,585,009,223	2023-07-16 19:01:59	SE	235,600	21,204
5	6,376,554,357,665,245,665	2023-01-08 12:58:44	FI	21,005	21,005
6	444,730,773,991,424,385	2023-02-12 10:21:40	FI	19,060	19,060
7	5,099,322,350,756,277,912	2023-01-12 21:02:50	FI	18,940	18,940
8	5,469,390,330,980,797,994	2023-03-17 06:04:55	FI	18,220	18,220
9	3,443,456,032,781,699,683	2023-01-08 22:20:17	FI	17,394	17,394
10	1,826,925,887,983,501,506	2023-03-21 14:52:59	EE	16,738	16,738

شکل ۲: ۱۰ کاربر با بیشترین مقدار sales در سال ۲۰۲۳

۲.۱ سوال ۲

برای مشخص کردن بیشترین سفارش ثبت شده از اتصال نتایج دو جدول orders و providers استفاده شده است. مطابق کوئری شکل ۳ بیشترین سفارش ثبت شده مربوط به خرید وعده غذایی با تعداد ۳۰۵,۲۵۴ می‌باشد.

```
SELECT SUM(quantity) as total_quantity, defaultOfferType
FROM providers
JOIN orders
ON providers.id = orders.providerId
GROUP BY defaultOfferType
ORDER BY total_quantity DESC
LIMIT 1;
```

شکل ۳: کوئری ۱-۲

۲ بخش دوم

در ادامه این بخش لازم است فایل csv تعطیلات را در دیتابیس فعلی باز کرده تا دسترسی لازم به همه دیتاها ایجاد شود.

۱.۲ سوال ۱

به کمک کوئری شکل ۴ و نتایج به دست آمده که در شکل ۵ نشان داده شده است. یکی از دلایل احتمالی، سفر مردم به خارج از شهر و در نتیجه عدم استفاده از این اپلیکیشن است. همچنین ممکن است به دلیل حضور مردم در خانه - و نه در محل کار - فرصت کافی برای پخت غذا توسط خودشان در روزهای تعطیل وجود داشته باشد.

۲.۲ سوال ۲

با توجه به نتایج به دست آمده، تعداد تأمین کنندگان نیز در روزهای تعطیل کاهش یافته است. در این کوئری تعداد روزهای تعطیل و غیرتعطیل نشان داده شده است. تفاوت تعداد نمونه‌های این دو دسته می‌تواند منجر به تفاوت واریانس دو نمونه و نقض شرط برابری واریانس نمونه‌ها شود. بنابراین بهتر است در صورت نیاز به انجام آزمون فرض آماری برای بررسی معناداری، از آزمون‌های غیر پارامتری استفاده شود. علت احتمالی کاهش تأمین کنندگان در روزهای تعطیل را می‌توان مرتبط با کاهش تقاضای کاربران در این روزها دانست.

۳.۲ سوال ۳

با توجه به نتایج قسمت‌های قبل به نظر می‌رسد این کمپین موفقیت‌آمیز نبوده است.

```

SELECT AVG(daily_total_quantity) as avg_daily_quantity,
       AVG(daily_total_sale) as avg_daily_sale,
       CASE
         WHEN hld.holyDate IS NULL THEN 'Regular Day'
         ELSE 'Holiday'
       END AS day_type
FROM orders ord
JOIN (
  SELECT
    date(createdAt) as daily_date,
    SUM(quantity) AS daily_total_quantity,
    SUM(CASE
      when currency='sek' then sales*0.8
      else sales
    END) AS daily_total_sale
  FROM
    orders
  GROUP BY
    daily_date
) as subq
ON date(ord.createdAt) = subq.daily_date
LEFT JOIN holidays hld
ON SUBSTR(ord.createdAt,0,11) = hld.holyDate
GROUP BY day_type

```

شکل ۴: کوئری ۱-۲

	123 avg daily quantity ▼	123 avg daily sale ▼	ABC day type ▼
1	511.3267684887	274,671.2704983923	Holiday
2	1,050.4928558111	581,201.6315429076	Regular Day

شکل ۵: میانگین مجموع تعداد و درآمد فروش روزانه در تعطیلات و روزهای عادی

```

SELECT
  COUNT(DISTINCT SUBSTR(ord.createdAt,0,11)) as all_days,
  COUNT(DISTINCT ord.providerId) AS providers_count,
  CASE
    WHEN hld.holyDate IS NULL THEN 'Regular Day'
    ELSE 'Holiday'
  END AS day_type
FROM orders ord
LEFT JOIN holidays hld
ON SUBSTR(ord.createdAt,0,11) = hld.holyDate
GROUP BY day_type

```

شکل ۶: کوئری ۲-۲

	123 all days ▼	123 providers count ▼	ABC day type ▼	
1	16	1,206	Holiday	
2	403	3,561	Regular Day	

شکل ۷: تعداد تأمین کنندگان به تفکیک روزهای عادی و تعطیلات

۳ بخش سوم

داشبورد مورد نظر در شکل ۸ نشان داده شده است. برای محاسبه m1 retention از کد پایتون نیز کمک گرفته شده است که در شاخه python_files قرار دارند.



شکل ۸: داشبورد