

Домашнее задание 3

Выполнил - Третьяков Александр Юрьевич

Упражнение 18

Задание

Предположим, что нам понадобилось иметь в базе данных сведения о технических характеристиках самолетов, эксплуатируемых в авиакомпании. Пусть это будут такие сведения, как число членов экипажа (пилоты), тип двигателей и их количество. Следовательно, необходимо добавить новый столбец в таблицу «Самолеты» (`aircrafts`). Дадим ему имя `specifications`, а в качестве типа данных выберем `jsonb`. Если впоследствии потребуется добавить и другие характеристики, то мы сможем это сделать, не модифицируя определение таблицы.

```
ALTER TABLE aircrafts ADD COLUMN specifications jsonb;
```

```
ALTER TABLE
```

Добавим сведения для модели самолета `Airbus A320-200`:

```
UPDATE aircrafts
SET specifications =
'{
  "crew": 2,
  "engines": {
    "type": "IAE V2500",
    "num": 2
  }
}'::jsonb
WHERE aircraft_code = '320';

UPDATE 1
```

Посмотрим, что получилось:

```
SELECT model, specifications
FROM aircrafts
WHERE aircraft_code = '320';

model | specifications
-----+-----
Airbus A320-200 | {"crew": 2, "engines": {"num": 2, "type": "IAE V2500"}}

(1 строка)
```

Можно посмотреть только сведения о двигателях:

```
SELECT model, specifications->'engines' AS engines
FROM aircrafts
WHERE aircraft_code = '320';

model | engines
-----+-----
Airbus A320-200 | {"num": 2, "type": "IAE V2500"}

(1 строка)
```

Чтобы получить еще более детальные сведения, например, о типе двигателей, нужно учитывать, что созданный **JSON**-объект имеет сложную структуру: он содержит вложенный **JSON**-объект. Поэтому нужно использовать оператор `#>` для указания пути доступа к ключу второго уровня.

```
SELECT model, specifications #> '{ engines, type }'
FROM aircrafts
WHERE aircraft_code = '320';

model | ?column?
-----+-----
Airbus A320-200 | "IAE V2500"

(1 строка)
```

Задание. Подумайте, какие еще таблицы было бы целесообразно дополнить столбцами типа `json/jsonb`. Вспомните, что, например, в таблице «Билеты» (`tickets`) уже есть столбец такого типа — `contact_data`. Выполните модификации таблиц и измените в них одну-две строки для проверки правильности ваших решений.

Решение

- Добавим в таблица `airports` - информация об услугах аэропорта

```
ALTER TABLE bookings.airports ADD COLUMN services jsonb;
```

Обновим данные для примера

```
UPDATE bookings.airports
SET services =
'{
  "terminals": 3,
  "runways": 2,
  "services": ["business_lounge", "free_wifi", "hotels", "parking"],
```

```
"capacity": 15000000,  
"international": true  
}':::jsonb  
WHERE airport_code = 'SVO';
```

```
UPDATE bookings.airports  
SET services =  
'{  
    "terminals": 1,  
    "runways": 2,  
    "services": ["business_lounge", "free_wifi", "parking", "hotels", "metro",  
"taxi"],  
    "capacity": 12000000,  
    "international": true,  
    "features": {  
        "parking_capacity": 5000,  
        "metro_connected": true,  
        "shops": 45,  
        "restaurants": 25  
    }  
}':::jsonb  
WHERE airport_code = 'DME';
```

Протестируем

Количество терминалов

```
SELECT  
    airport_code,  
    airport_name,  
    services #> '{terminals}' as terminals  
FROM bookings.airports  
WHERE airport_code IN ('SVO', 'DME');
```

```
demo=# SELECT
    airport_code,
    airport_name,
    services #> '{terminals}' as terminals
FROM bookings.airports
WHERE airport_code IN ('SVO', 'DME');
    airport_code | airport_name | terminals
-----+-----+-----+
    SVO          | Шереметьево | 3
    DME          | Домодедово | 1
(2 rows)
```

Первые три услуги из массива

```
SELECT
    airport_code,
    airport_name,
    services #> '{services, 0}' as service_1,
    services #> '{services, 1}' as service_2,
    services #> '{services, 2}' as service_3
FROM bookings.airports
WHERE airport_code IN ('SVO', 'DME');
```

```
demo=# SELECT
    airport_code,
    airport_name,
    services #> '{services, 0}' as service_1,
    services #> '{services, 1}' as service_2,
    services #> '{services, 2}' as service_3
FROM bookings.airports
WHERE airport_code IN ('SVO', 'DME');
    airport_code | airport_name | service_1 | service_2 | service_3
-----+-----+-----+-----+
    SVO          | Шереметьево | "business_lounge" | "free_wifi" | "hotels"
    DME          | Домодедово | "business_lounge" | "free_wifi" | "parking"
(2 rows)
```

Информация о парковке (вложенный объект features)

```

SELECT
    airport_code,
    airport_name,
    services #> '{features, parking_capacity}' as parking_spaces,
    services #> '{features, metro_connected}' as has_metro
FROM bookings.airports
WHERE airport_code IN ('SVO', 'DME');

```

```

demo=# SELECT
    airport_code,
    airport_name,
    services #> '{features, parking_capacity}' as parking_spaces,
    services #> '{features, metro_connected}' as has_metro
FROM bookings.airports
WHERE airport_code IN ('SVO', 'DME');
    airport_code | airport_name | parking_spaces | has_metro
-----+-----+-----+-----+
  SVO      | Шереметьево |          | 
  DME      | Домодедово |    5000    | true
(2 rows)

```

- Добавим в таблицу `bookings` детали бронирования

```
ALTER TABLE bookings.bookings ADD COLUMN booking_details jsonb;
```

```

UPDATE bookings.bookings
SET booking_details =
'{
    "payment_method": "credit_card",
    "loyalty_program": "AeroBonus",
    "special_requests": ["window_seat", "vegetarian_meal"],
    "insurance": true,
    "priority_boarding": false
}'::jsonb
WHERE book_ref = '00000F';

```

```

UPDATE bookings.bookings
SET booking_details =
'{
    "payment_method": "debit_card",
    "loyalty_program": "AeroBonus",
    "special_requests": ["window_seat", "child_meal"],
    "insurance": true,
}
```

```

    "priority_boarding": true
}':::jsonb
WHERE book_ref = '000012';

```

Сравним методы оплаты

```

SELECT
  book_ref,
  total_amount,
  booking_details #> '{payment_method}' as payment_method,
  booking_details #> '{insurance}' as has_insurance
FROM bookings.bookings
WHERE book_ref IN ('00000F', '000012');

```

```

demo=# SELECT
  book_ref,
  total_amount,
  booking_details #> '{payment_method}' as payment_method,
  booking_details #> '{insurance}' as has_insurance
FROM bookings.bookings
WHERE book_ref IN ('00000F', '000012');
  book_ref | total_amount | payment_method | has_insurance
-----+-----+-----+-----+
  00000F   |    265700.00 | "credit_card" | true
  000012   |    37900.00 | "debit_card" | true
(2 rows)

```

Полностью сравним все бронирования

```

SELECT
  book_ref as "Номер брони",
  total_amount as "Сумма",
  booking_details #> '{payment_method}' as "Метод оплаты",
  booking_details #> '{loyalty_program}' as "Программа лояльности",
  (booking_details #> '{insurance}')::bool as "Страховка",
  (booking_details #> '{priority_boarding}')::bool as "Приоритетная посадка",
  booking_details #> '{special_requests}' as "Специальные запросы"
FROM bookings.bookings
WHERE booking_details IS NOT NULL
ORDER BY book_ref;

```

```
demo=# SELECT
    book_ref as "Номер брони",
    total_amount as "Сумма",
    booking_details #> '{payment_method}' as "Метод оплаты",
    booking_details #> '{loyalty_program}' as "Программа лояльности",
    (booking_details #> '{insurance}')::bool as "Страховка",
    (booking_details #> '{priority_boarding}')::bool as "Приоритетная посадка",
    booking_details #> '{special_requests}' as "Специальные запросы"
FROM bookings.bookings
WHERE booking_details IS NOT NULL
ORDER BY book_ref;
Номер брони | Сумма | Метод оплаты | Программа лояльности | Страховка | Приоритетная посадка | Специальные запросы
-----+-----+-----+-----+-----+-----+-----+
00000F   | 265700.00 | "credit_card" | "AeroBonus"          | t           | f           | ["window_seat", "vegetarian_meal"]
000012   | 37900.00  | "debit_card"  | "AeroBonus"          | t           | t           | ["window_seat", "child_meal"]
(2 rows)
```