

Домашнее задание 8

Дарья Яковлева, М3439

29.11.2016

Задание: Реализовать базу данных Airline, которая содержит информацию о рейсах самолётов.

1. База данных

```
CREATE DATABASE airline;
```

```
CREATE TABLE IF NOT EXISTS planes (  
    plane_id int PRIMARY KEY,  
    capacity int not null  
);
```

```
CREATE TABLE IF NOT EXISTS flights (  
    flight_id SERIAL PRIMARY KEY,  
    flight_num int,  
    flight_time timestamp without time zone,  
    plane_id int not null REFERENCES planes (plane_id)  
    ON DELETE CASCADE ON UPDATE CASCADE  
);
```

```
CREATE TABLE IF NOT EXISTS seats (  
    flight_id int REFERENCES flights (flight_id)  
    ON DELETE CASCADE ON UPDATE CASCADE,  
    seat_no int not null,  
    status int not null, — 1 — reserved, 2 — bought  
    reserved_time timestamp without time zone,  
    PRIMARY KEY(flight_id, seat_no)  
);
```

```
INSERT INTO planes (plane_id, capacity)  
VALUES (1, 100),  
       (2, 120),  
       (3, 50);
```

```
INSERT INTO flights (flight_num, flight_time, plane_id)  
VALUES (1, timestamp '2016-11-28 20:38:40', 1),
```

```

        (2, timestamp '2016-11-29 19:38:40 ', 2),
        (3, timestamp '2016-12-01 01:38:20 ', 1),
        (4, timestamp '2016-12-14 03:38:40 ', 2),
        (5, timestamp '2016-12-16 13:38:10 ', 2);

INSERT INTO seats (flight_id, seat_no, status, reserved_time)
VALUES (1, 1, 1, timestamp '2016-11-28 13:38:10 '),
       (1, 2, 2, timestamp '2016-11-28 10:38:10 '),
       (1, 3, 1, timestamp '2016-11-27 13:38:10 '),
       (2, 2, 2, timestamp '2016-11-26 13:38:10 '),
       (2, 5, 1, timestamp '2016-11-29 13:38:10 ');

```

2. Поддержка бронирования и покупки мест

```

CREATE OR REPLACE FUNCTION seats_action() RETURNS TRIGGER AS $SeatsAct$
DECLARE
    timespent interval;
    cur_bought int;
    flight_time interval;
    cur_capacity int;
BEGIN
    timespent := NEW.reserved_time - (SELECT flight_time
FROM flights WHERE NEW.flights_id = flights.flights_id);
    flight_time := now() - (SELECT flight_time FROM flights
WHERE NEW.flights_id = flights.flights_id);
    cur_capacity := SELECT capacity FROM planes
WHERE plane_id = (
        SELECT plane_id FROM flights
        WHERE flight_id = NEW.flight_id);

    IF (NEW.seat_no > cur_capacity)
    THEN
        NEW := OLD;
        RETURN NULL;
    END IF;

    IF (NEW.status = 2
        AND
        (SELECT EXTRACT (DAY FROM timespent)) = 0
        AND
        (SELECT EXTRACT (HOUR FROM timespent)) < 2)
    OR
        (NEW.status = 1
        AND
        (SELECT EXTRACT (DAY FROM timespent)) < 1
        AND (SELECT EXTRACT (DAY FROM flight_time)) > 0)

```

```

THEN
    NEW := OLD;
    RETURN OLD;
END IF;

cur_bought := (SELECT count(seats_no)
FROM seats WHERE flight_id = NEW.flight_id);

IF cur_bought = (SELECT capacity FROM planes WHERE
plane_id = (
    SELECT plane_id FROM flights
    WHERE flight_id = NEW.flight_id))
THEN
    NEW := OLD;
    RETURN NULL;
END IF;

IF (TG_OP = 'UPDATE') AND (NEW.status = 1) AND (OLD.status = 1)
THEN
    NEW.reserved_time := now();
    RETURN NEW;
ELSIF (TG_OP = 'INSERT')
THEN
    IF (
        SELECT count(*)
FROM seats
WHERE flight_id = NEW.flight_id AND
        seat_no = NEW.seat_no) > 0
    AND (
        SELECT status
FROM seats
WHERE flight_id = NEW.flight_id AND
        seat_no = NEW.seat_no) = 1
    THEN
        IF (now() - (SELECT reserved_time
FROM seats
WHERE flight_id = NEW.flight_id AND
        seat_no = NEW.seat_no)) > 24
        THEN
            DELETE
FROM seats
WHERE flight_id = NEW.flight_id AND
        seat_no = NEW.seat_no;
            RETURN NEW;
        ELSE
            NEW := OLD;

```

```

                                RETURN NULL;
                            END IF;
                        END IF;
                    END IF;
                RETURN NEW;
            END;
$SeatsAct$ LANGUAGE plpgsql;

```

```

CREATE TRIGGER SeatsAct
BEFORE INSERT OR UPDATE ON seats
FOR EACH ROW EXECUTE PROCEDURE seats_action();

```

3. Добавление индексов

Нужно добавить индекс *(flight_id, flight_time)*, так как это это большой и частый запрос.

Не нужно добавлять индексы *(flight_id, seat_no)*, *(flight_id, reserved_time)*, так как это частые запросы, но они ограничены размером самолета, который редко бывает больше 1000.

4. Пусть частым запросом является определение средней заполненности самолёта по рейсу. Какие индексы могут помочь при исполнении данного запроса?

В этом случае индекс *(flight_id, status)* может помочь, так с помощью него можно посчитать среднюю заполненность самолета.

5. Добавление индексов на языке SQL

```

CREATE INDEX USING HASH ON TABLE
    flights (flight_id , flight_time)
CREATE INDEX USING HASH ON TABLE
    seats (flight_id , status)

```