Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут”

Кафедра АСОІУ

ЗВІТ

про виконання контрольної роботи

з дисципліни

“Програмне застосування баз данних”

Тема: Аналітичні запити

|  |  |  |
| --- | --- | --- |
| Прийняв: |  | Виконав: |
| Клименко О.М. |  | студент 2-го курсу  гр. ІП-52 ФІОТ  Набоков Е.М |

Київ 2017

1 ПОСТАНОВКА ЗАДАЧІ

|  |  |  |  |
| --- | --- | --- | --- |
| 9. | База трейлерів  **1.** У якому гаражі найменша  кількість ремонтів.  **2.** Надати список назв ремонтів  по маркам машин. В кінці  надати кількість ремонтів за  кожною маркою загалом.  **3.** Чи є гаражі не  укомплектовані механіками.  **4.** Надати номер гаража, марку  машин та табельний номер  для машин, що не булі у  ремонті жодного разу.  **5.** У яких гаражах  ремонтувалися машини з  «Дата» по «Дата» за  ремонтом „Назва” | Парк трейлерів | Номенклатурний н-р машини  Марка машини  Дата вироблення  Номер гаража |
| Види | Код виду ремонтів  Назва виду |
| Персонал | Табельній номер механіка  Прізвище  ІПБ  Номер гаража  Стаж роботи |
| Ремонтні | Дата  Номенклатурній номер  машини  Табельній номер механіка  Код вигляду ремонту. |

2 СХЕМА БАЗИ ДАННИХ

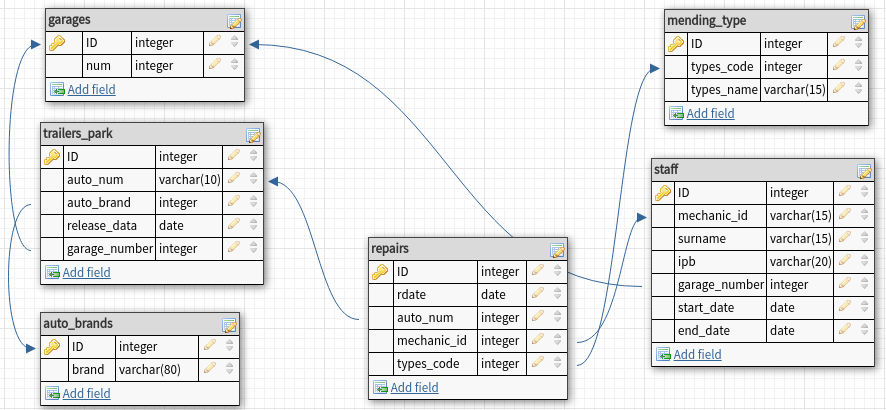


Рисунок 2.1 — схема.

**3** **ВИХІДНИЙ КОД**

**3.1** Створення бази данних

-- Create database named trailers

CREATE DATABASE trailers;

-- Create table with well-known brands of cars

CREATE TABLE auto\_brands (

id SERIAL PRIMARY KEY,

brand VARCHAR(80) UNIQUE

);

-- Create table with all garages' numbers (just their numbers)

CREATE TABLE garages (

id SERIAL PRIMARY KEY,

num INT NOT NULL UNIQUE

);

/\* Create table with

\* - code of repairs

\* - name of that repairs

\*/

CREATE TABLE mending\_type (

id SERIAL PRIMARY KEY,

types\_code INT UNIQUE,

types\_name VARCHAR(15) UNIQUE

);

/\* Create table with:

\* - unique number of car

\* - brand of car

\* - data release of car

\* - garage's number

\*/

CREATE TABLE trailers\_park (

id SERIAL PRIMARY KEY,

auto\_num VARCHAR(10),

auto\_brand INT references auto\_brands(id),

release\_date DATE,

garage\_number INT references garages(id)

);

/\* Create table with:

\* - unique id for each mechanic

\* - mechanics' surname

\* - ipb == name + middle name

\* - garage number, which belongs to that mechanic

\* - experiance (start date and end date)

\*/

CREATE TABLE staff (

id SERIAL PRIMARY KEY,

mechanic\_id VARCHAR(15) UNIQUE,

surname VARCHAR(15),

IPB VARCHAR(20),

garage\_number INT references garages(id),

start\_date DATE,

end\_date DATE

);

/\* Create table with:

\* - date of repairs, when it was done (or is going to be)

\* - number of the car, that was/is going to be repaired

\* - mechanic id, who did/is going to do it

\* - type of repairs

\*/

CREATE TABLE repairs (

id SERIAL PRIMARY KEY,

rdate DATE,

auto\_num INT references trailers\_park(id),

mechanic\_id INT references staff(id),

types\_code INT references mending\_type(id)

);

**3.2** Загальні запити

-- COMMON TASKS

-- З використанням агрегуючих функцій та конструкції HAVING.

SELECT garage\_number, COUNT(garage\_number) FROM repairs

JOIN trailers\_park ON trailers\_park.id = repairs.auto\_num

GROUP BY garage\_number

HAVING COUNT(garage\_number) > 10

ORDER BY COUNT(garage\_number);

-- З перетворенням типу даних результату запиту або формату дати.

SELECT id, to\_char(start\_date, 'HH12:MI:SS') FROM staff;

-- З пошуком по фрагменту текстового поля. (Ех: знайти всі прізвища, що закінчуються на «ко»)

SELECT \* FROM staff WHERE surname LIKE 'Wa%';

-- зі вставкою будь-якого значення в поле результату запиту, що набуло значення NULL в результаті запиту (Ех: якщо прізвище «AUDI», що купив, - NULL, поставити символ «\_» оператор NVL або CASE ). �

SELECT num as garage, brand, trailers\_park.auto\_num as tab\_num\_auto FROM repairs

RIGHT JOIN trailers\_park ON repairs.auto\_num = trailers\_park.id

JOIN garages ON trailers\_park.garage\_number = garages.id

JOIN auto\_brands ON trailers\_park.auto\_brand = auto\_brands.id

CASE WHEN repairs.auto\_num IS NULL THEN 'No car' ELSE repairs.auto\_num END;

**4** **РЕЗУЛЬТАТ ВИКОНАННЯ РОБОТИ ЗАПИТІВ**

**4.1** Перший запит

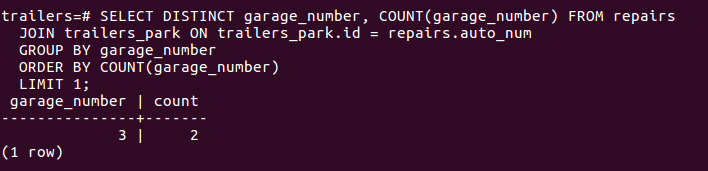
SELECT DISTINCT garage\_number, COUNT(garage\_number) FROM repairs

JOIN trailers\_park ON trailers\_park.id = repairs.auto\_num

GROUP BY garage\_number

ORDER BY COUNT(garage\_number)

LIMIT 1;



**4.2** Другий запит (a)

SELECT brand, types\_name FROM auto\_brands

JOIN trailers\_park ON trailers\_park.auto\_brand =auto\_brands.id

JOIN repairs ON trailers\_park.id = repairs.auto\_num

JOIN mending\_type ON mending\_type.id = repairs.types\_code;



**4.2** Другий запит (b)

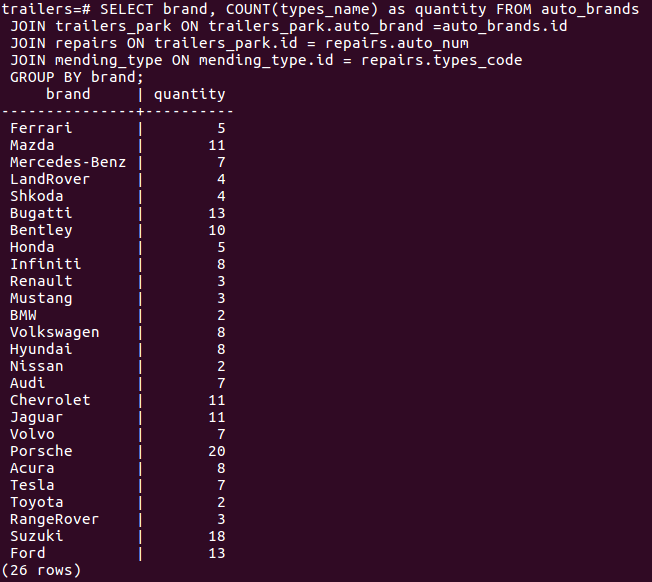
SELECT brand, COUNT(types\_name) as quantity FROM auto\_brands

JOIN trailers\_park ON trailers\_park.auto\_brand =auto\_brands.id

JOIN repairs ON trailers\_park.id = repairs.auto\_num

JOIN mending\_type ON mending\_type.id = repairs.types\_code

GROUP BY brand;

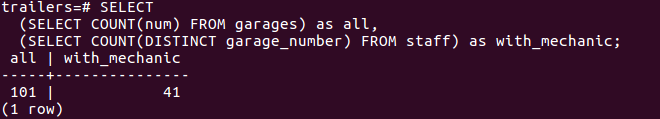


**4.3** Третій запит

SELECT

(SELECT COUNT(num) FROM garages) as all,

(SELECT COUNT(DISTINCT garage\_number) FROM staff) as with\_mechanic;



**4.4** Четвертий запит

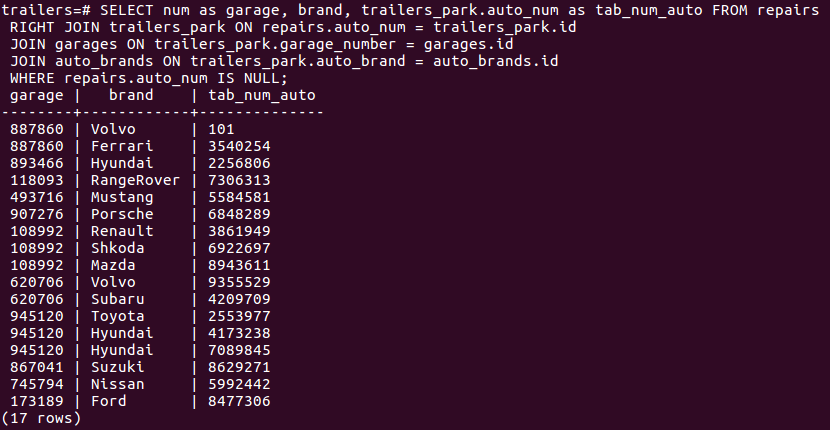
SELECT num as garage, brand, trailers\_park.auto\_num as tab\_num\_auto FROM repairs

RIGHT JOIN trailers\_park ON repairs.auto\_num = trailers\_park.id

JOIN garages ON trailers\_park.garage\_number = garages.id

JOIN auto\_brands ON trailers\_park.auto\_brand = auto\_brands.id

WHERE repairs.auto\_num IS NULL;



**4.5**  П ятий запит

CREATE FUNCTION discover(DATE, DATE, VARCHAR(15))

RETURNS TABLE(garage INT, cdate DATE, name\_repairs VARCHAR(15))

AS $$ SELECT num, rdate, $3 FROM repairs

JOIN trailers\_park ON repairs.auto\_num = trailers\_park.id

JOIN garages ON trailers\_park.garage\_number = garages.id

JOIN mending\_type ON repairs.types\_code = mending\_type.id

WHERE mending\_type.types\_name = $3 AND rdate BETWEEN $1 AND $2

ORDER BY rdate $$

LANGUAGE SQL;

SELECT \* FROM discover('1998-1-1', '2015-1-1', 'AA');

