Пороскун Олена Олегівна ПМ-81

Лабораторна робота 11

1. Задания

Напишіть запит, який би повертав з таблиці emp інформацію:

* про ім'я співробітника;
* про рівні підпорядкованості (найвищий рівень - головний начальник, який нікому не підпорядковується - рівень 0);
* про шляхи підзвітності в форматі:

/ Керівник 1 / керівник 2 / рядовий співробітник

* Звіт повинен бути відсортований за рівнем підпорядкованості.

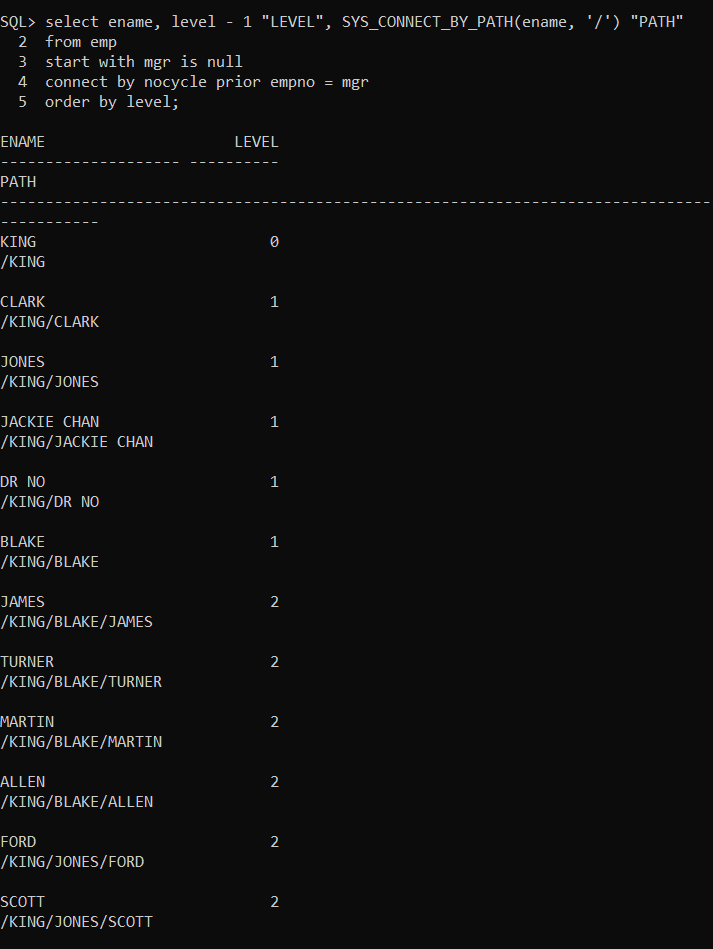
**select ename, level - 1 "LEVEL", SYS\_CONNECT\_BY\_PATH(ename, '/') "PATH"**

**from emp**

**start with mgr is null**

**connect by nocycle prior empno = mgr**

**order by level;**

****

****

2. Начальники

Виберіть всі імена начальників співробітника на ім'я «JET LI»

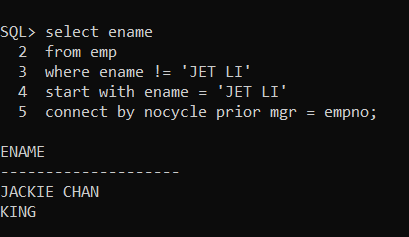
**select ename**

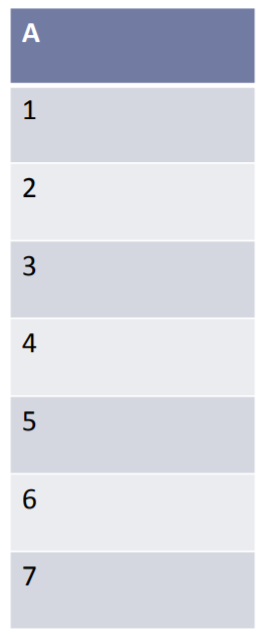
**from emp**

**where ename != 'JET LI'**

**start with ename = 'JET LI'**

**connect by nocycle prior mgr = empno;**



3\*. Пропуски

1. Створіть таблицю А

2. Видаліть з таблиці рядки 4 і 6.

3. Напишіть запит, який виведе віддалені числа.

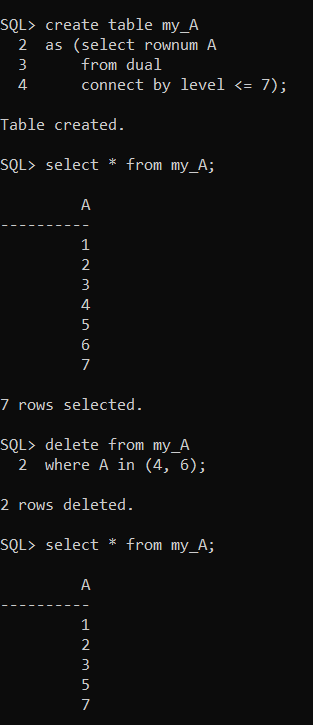
4. \* Як буде виглядати запит, якщо максимальний елемент наперед не відомий

5. \* Як буде виглядати запит якщо максимальний і мінімальний елемент наперед невідомі?

6. Видаліть таблицю А.

*Підказка*: числову послідовність можна згенерувати запитом

Select rownum From dual Connect by level <= 7

1. Створіть таблицю А

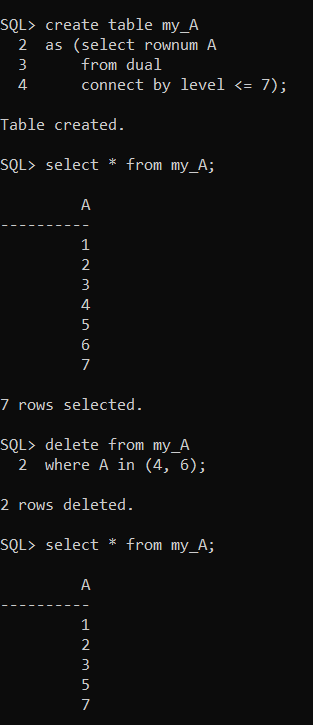
**create table my\_A**

**as (select rownum A**

**from dual**

**connect by level <= 7);**

**select \* from my\_A;**

****

2. Видаліть з таблиці рядки 4 і 6.

**delete from my\_A**

**where A in (4, 6);**

**select \* from my\_A;**

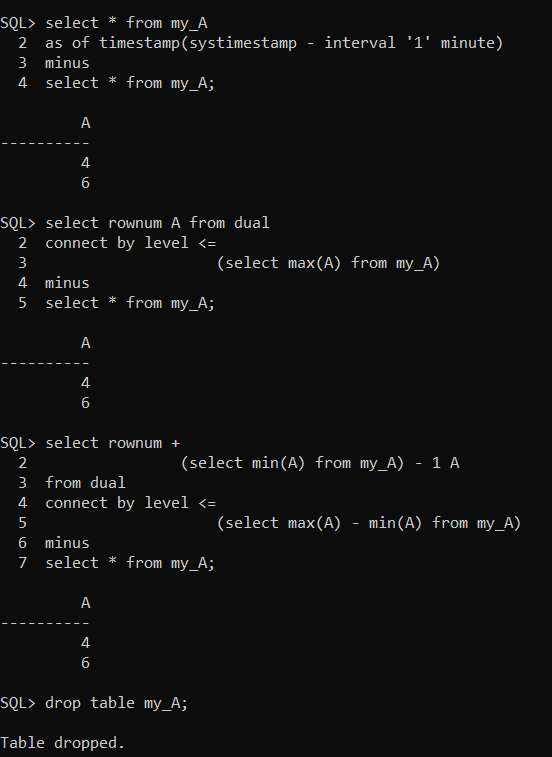
**3. Напишіть запит, який виведе віддалені числа.**

**select \* from my\_A**

**as of timestamp(systimestamp - interval '1' minute)**

**minus**

**select \* from my\_A;**

****

4. \* Як буде виглядати запит, якщо максимальний елемент наперед не відомий

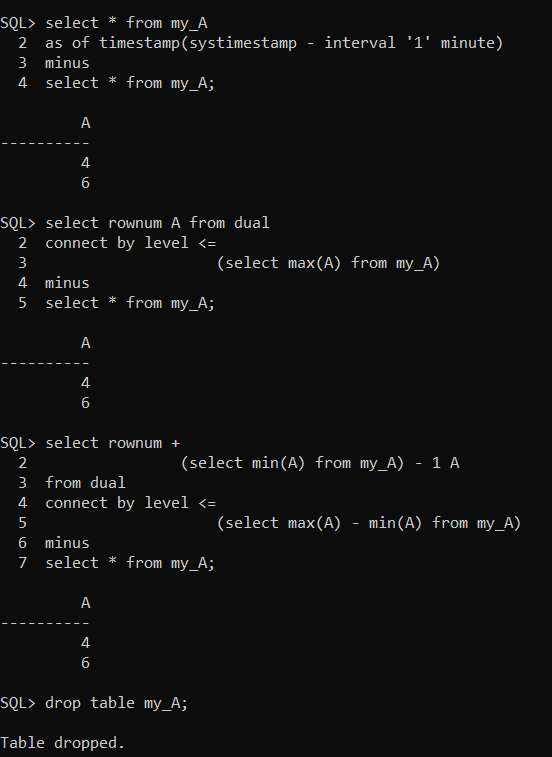
**select rownum A from dual**

**connect by level <=**

**(select max(A) from my\_A)**

**minus**

**select \* from my\_A;**

****

5. \* Як буде виглядати запит якщо максимальний і мінімальний елемент наперед невідомі?

**select rownum +**

**(select min(A) from my\_A) - 1 A**

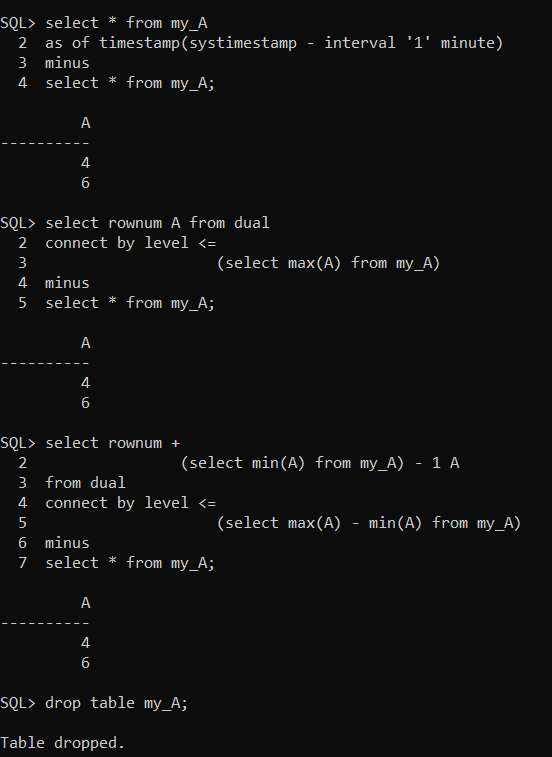
**from dual**

**connect by level <=**

**(select max(A) - min(A) from my\_A)**

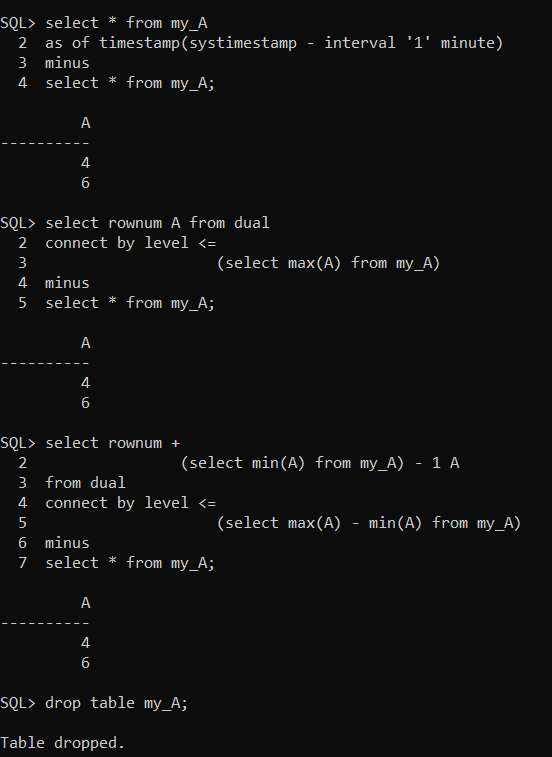
**minus**

**select \* from my\_A;**

****

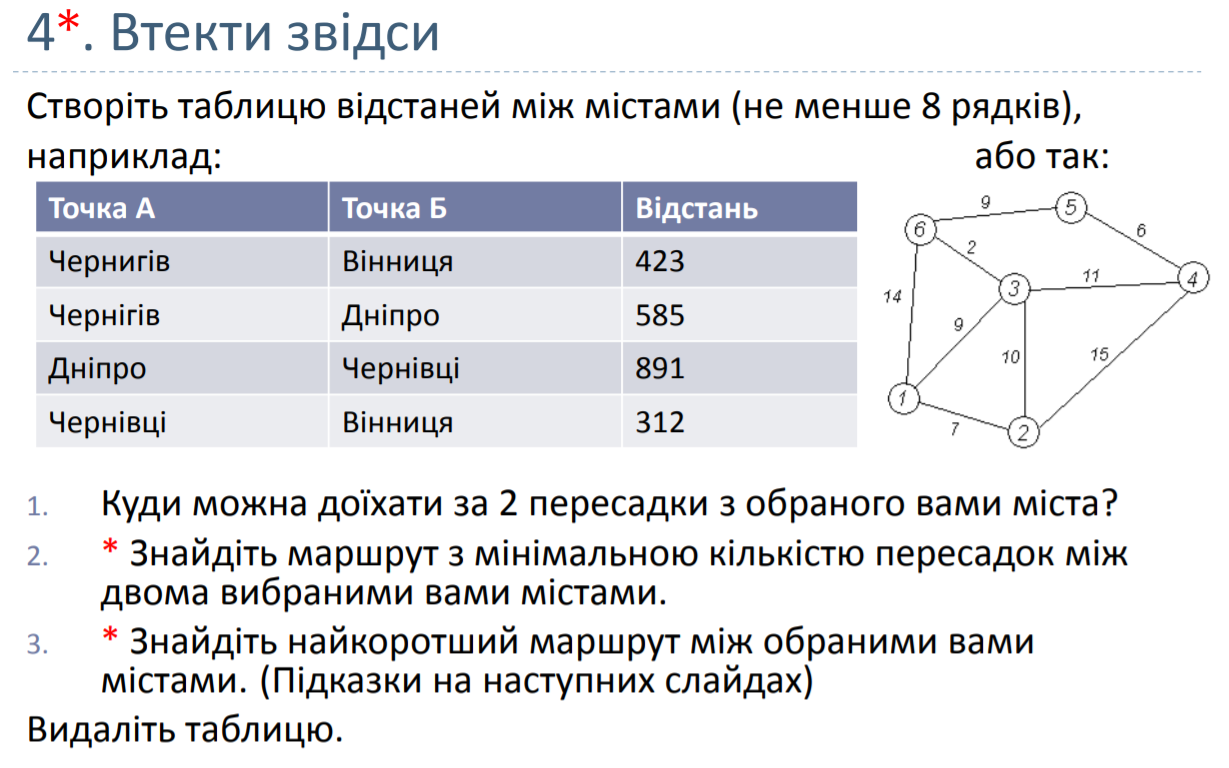
6. Видаліть таблицю А.

**drop table my\_A;**

****

4\*. Втекти звідси

Створіть таблицю відстаней між містами (не менше 8 рядків), наприклад: або так:



**create table my\_table(point\_A varchar(13),**

**point\_B varchar(13),**

**distance integer);**

**insert into my\_table values ('Chernihiv', 'Vinnytsia', 423);**

**insert into my\_table values ('Chernihiv', 'Dnipro', 585);**

**insert into my\_table values ('Dnipro', 'Chernivtsi', 891);**

**insert into my\_table values ('Chernivtsi', 'Vinnytsia', 312);**

**insert into my\_table values ('Kyiv', 'Dnipro', 533);**

**insert into my\_table values ('Simferopol', 'Odesa', 392);**

**insert into my\_table values ('Sumy', 'Lviv', 896);**

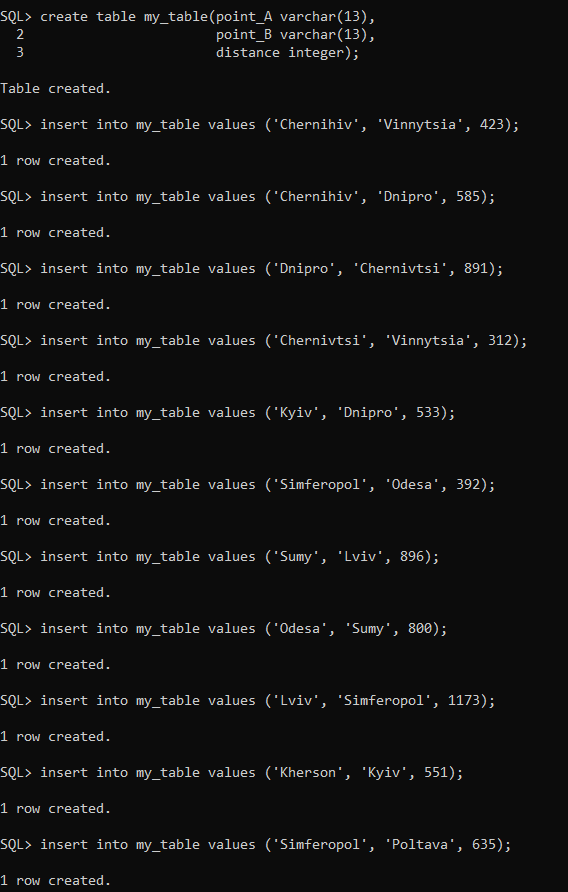
**insert into my\_table values ('Odesa', 'Sumy', 800);**

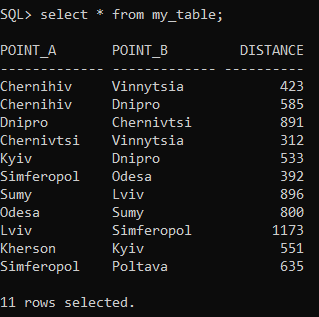
**insert into my\_table values ('Lviv', 'Simferopol', 1173);**

**insert into my\_table values ('Kherson', 'Kyiv', 551);**

**insert into my\_table values ('Simferopol', 'Poltava', 635);**

**select \* from my\_table;**

****

****

1. Куди можна доїхати за 2 пересадки з обраного вами міста?

**select distinct point\_B, level transfer**

**from my\_table**

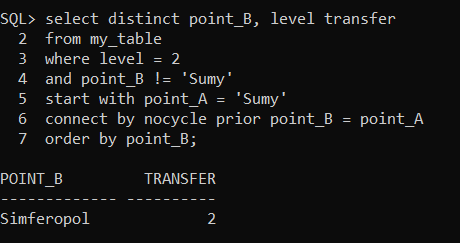
**where level = 2**

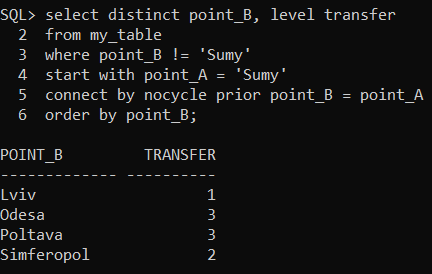
**and point\_B != 'Sumy'**

**start with point\_A = 'Sumy'**

**connect by nocycle prior point\_B = point\_A**

**order by point\_B;**

****

Перевірка:

**select distinct point\_B, level transfer**

**from my\_table**

**where point\_B != 'Sumy'**

**start with point\_A = 'Sumy'**

**connect by nocycle prior point\_B = point\_A**

**order by point\_B;**

2. \* Знайдіть маршрут з мінімальною кількістю пересадок між двома вибраними вами містами.

**select \***

**from**

**(select level transfer,**

**sys\_connect\_by\_path(point\_A,' -> ') || ' ==> ' || point\_B full\_path**

**from my\_table**

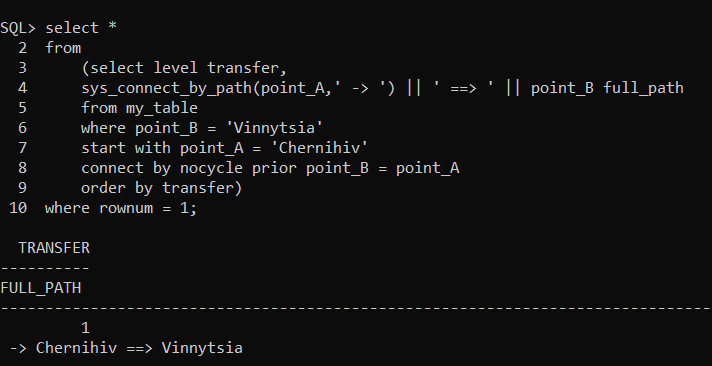
**where point\_B = 'Vinnytsia'**

**start with point\_A = 'Chernihiv'**

**connect by nocycle prior point\_B = point\_A**

**order by transfer)**

**where rownum = 1;**



Перевірка:

**select \***

**from**

**(select level transfer,**

**sys\_connect\_by\_path(point\_A,' -> ') || ' ==> ' || point\_B full\_path**

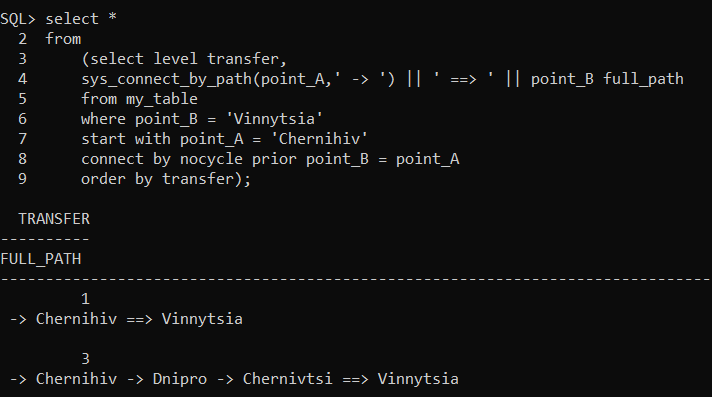
**from my\_table**

**where point\_B = 'Vinnytsia'**

**start with point\_A = 'Chernihiv'**

**connect by nocycle prior point\_B = point\_A**

**order by transfer);**



3. \* Знайдіть найкоротший маршрут між обраними вами містами. (Підказки на наступних слайдах)

<https://stackoverflow.com/questions/1267025/how-to-calculate-value-of-string-in-oracle> :

**create or replace function new\_user.calc(pi\_val varchar2) return number**

**is**

**v\_return number;**

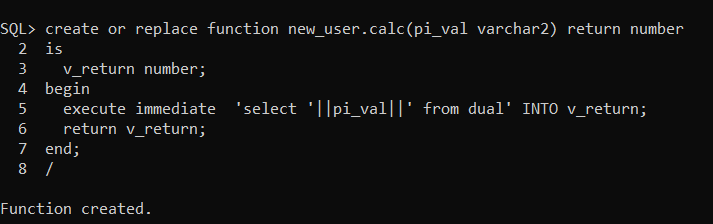
**begin**

**execute immediate 'select '||pi\_val||' from dual' INTO v\_return;**

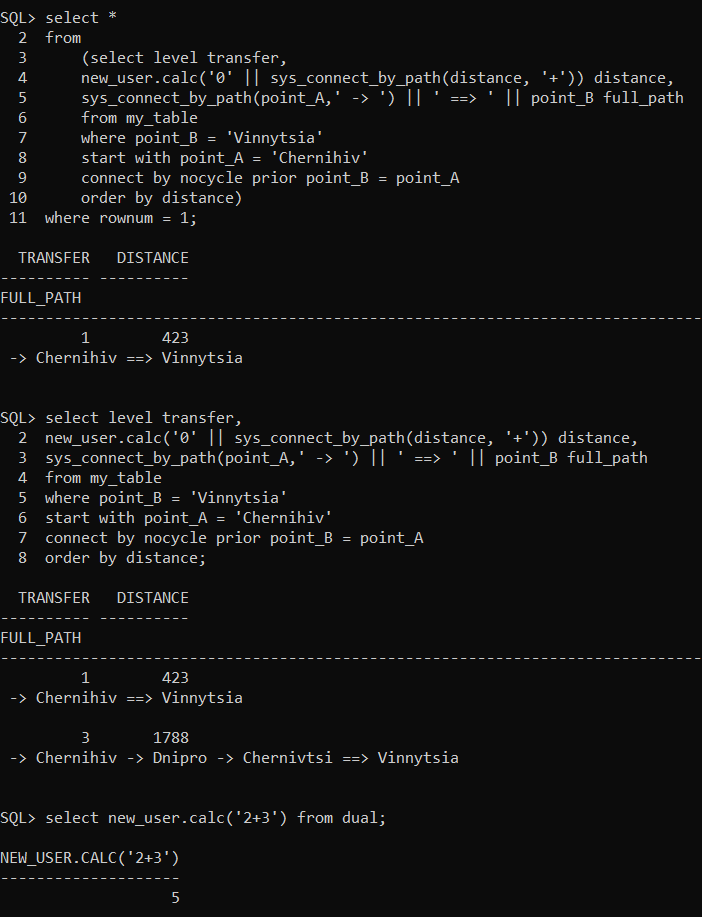
**return v\_return;**

**end;**

**/**

****

**select new\_user.calc('2+3') from dual;**

****

**select \***

**from**

**(select level transfer,**

**new\_user.calc('0' || sys\_connect\_by\_path(distance, '+')) distance,**

**sys\_connect\_by\_path(point\_A,' -> ') || ' ==> ' || point\_B full\_path**

**from my\_table**

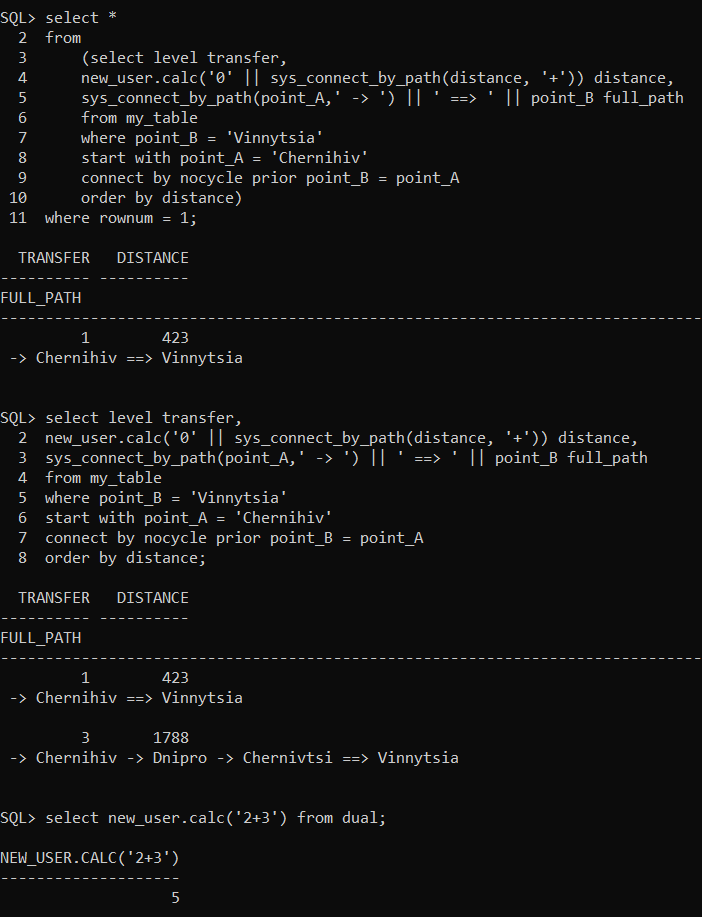
**where point\_B = 'Vinnytsia'**

**start with point\_A = 'Chernihiv'**

**connect by nocycle prior point\_B = point\_A**

**order by distance)**

**where rownum = 1;**

****

Перевірка:

**select level transfer,**

**new\_user.calc('0' || sys\_connect\_by\_path(distance, '+')) distance,**

**sys\_connect\_by\_path(point\_A,' -> ') || ' ==> ' || point\_B full\_path**

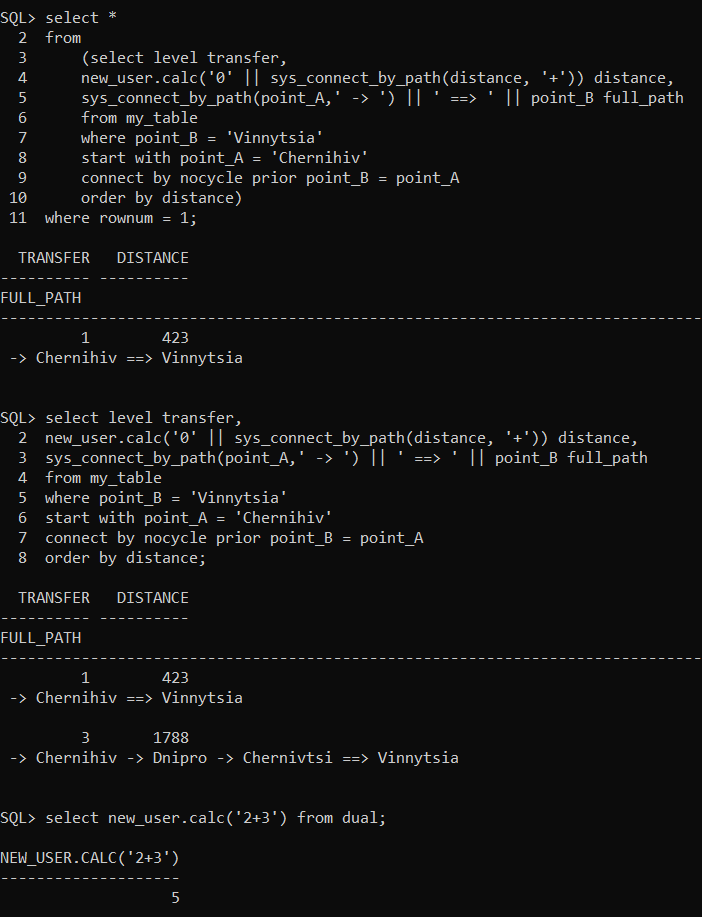
**from my\_table**

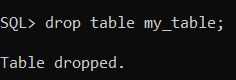
**where point\_B = 'Vinnytsia'**

**start with point\_A = 'Chernihiv'**

**connect by nocycle prior point\_B = point\_A**

**order by distance;**

****

Видаліть таблицю.

**drop table my\_table;**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*4.3 (підказки)*

*Як отримати довжину маршруту?*

*sys\_connect\_by\_path дозволяє отримати шлях від поточного рядка до кореневого; '0' || sys\_connect\_by\_path (length, '+') для кожного вузла дасть нам рядок із сумою відстаней до кореня (наприклад '0 + 1 + 1 + 2');*

*Обчислити рядок '0 + 1 + 1 + 2' можна декількома способами;*

*На думку Oracle-гуру TOM KYTE кращий спосіб:*

*SELECT XMLQUERY (‘0+1+1+2' RETURNING CONTENT).getnumberval () AS x FROM dual;*

*Проблема - не працює в Oracle 10 XE через ліцензійні обмеження (в 11 ХЕ - працює)*

*4.3 Короткий погляд в бік процедур, що зберігаються*

*Рішення з* [*http://stackoverflow.com/questions/1267025/how-tocalculate-value-of-string-in-oracle*](http://stackoverflow.com/questions/1267025/how-tocalculate-value-of-string-in-oracle)

*SQL> CREATE OR REPLACE FUNCTION calc(pi\_val VARCHAR2) RETURN NUMBER IS*

*2 v\_return NUMBER;*

*3 BEGIN*

*4 EXECUTE IMMEDIATE 'select '||pi\_val||' from dual' INTO v\_return;*

*5 RETURN v\_return;*

*6 END;*

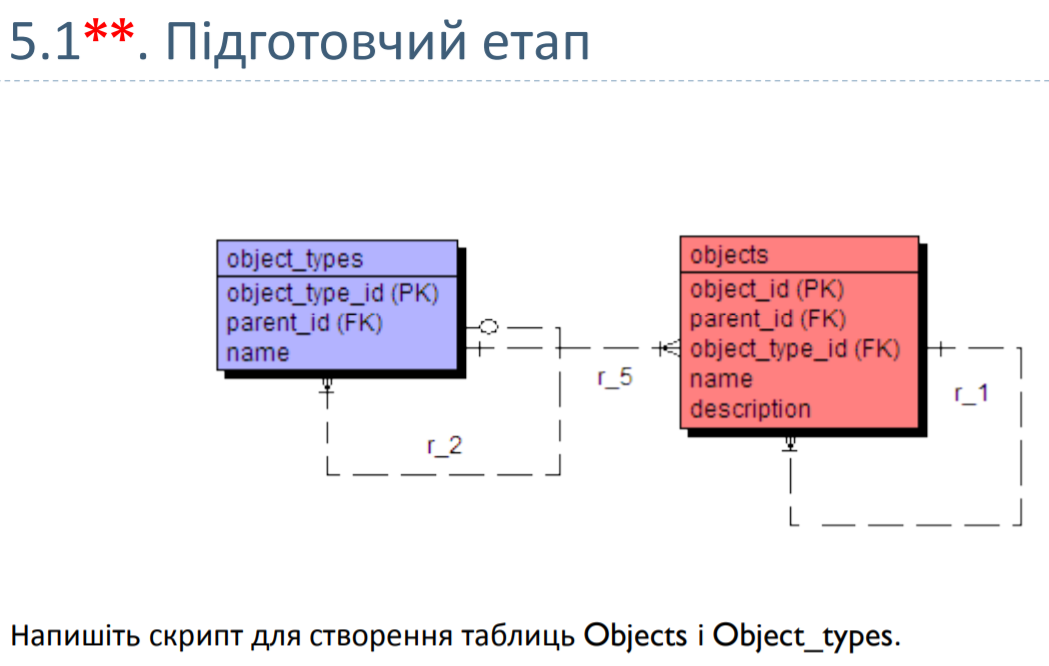
*7 /*

*SQL> SELECT calc('2\*6\*10') FROM dual; CALC('2\*6\*10') -------------- 120*

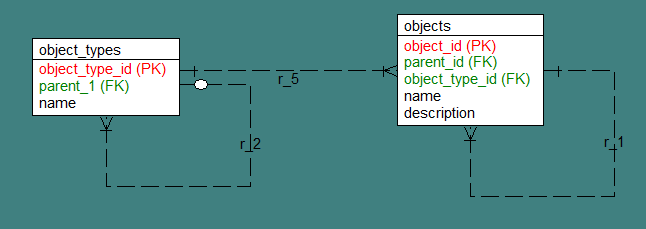
*Обмеження: рішення є потенційною загрозою безпеці. Дивіться обговорення за посиланням*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5.1\*\*. Підготовчий етап



Напишіть скрипт для створення таблиць Objects і Object\_types.



**Create table object\_types11 (**

**"object\_type\_id" Number(20) NOT NULL,**

**"parent\_id" Number(20) default null,**

**"name" Varchar2 (20),**

**primary key ("object\_type\_id")**

**);**

**Create table objects11 (**

**"object\_id" Number(20) NOT NULL ,**

**"parent\_id" Number(20) default null,**

**"object\_type\_id" Number(20) NOT NULL,**

**"name" Varchar2 (20),**

**"description" Varchar2 (20),**

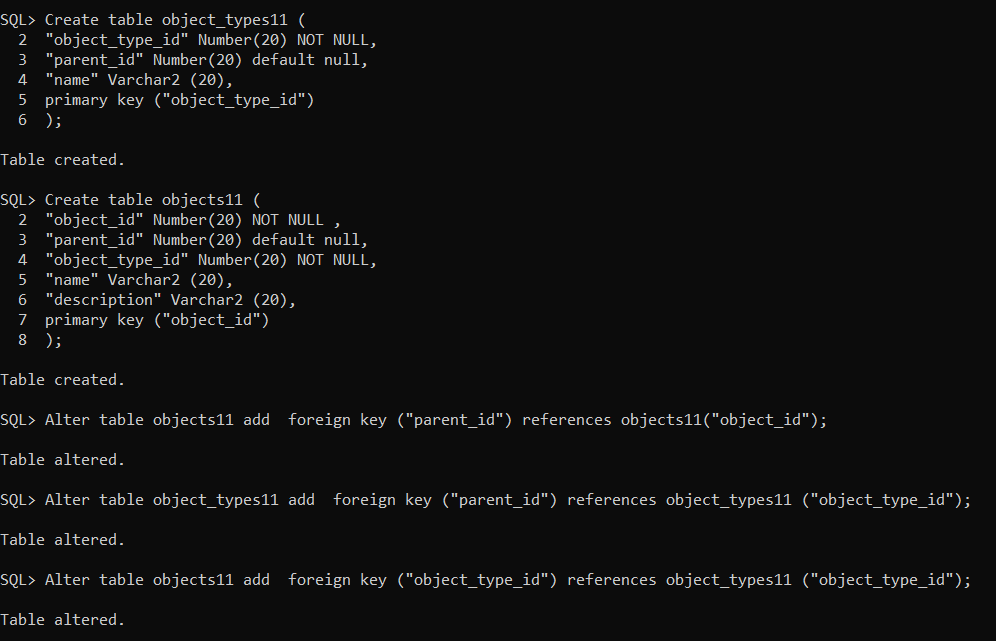
**primary key ("object\_id")**

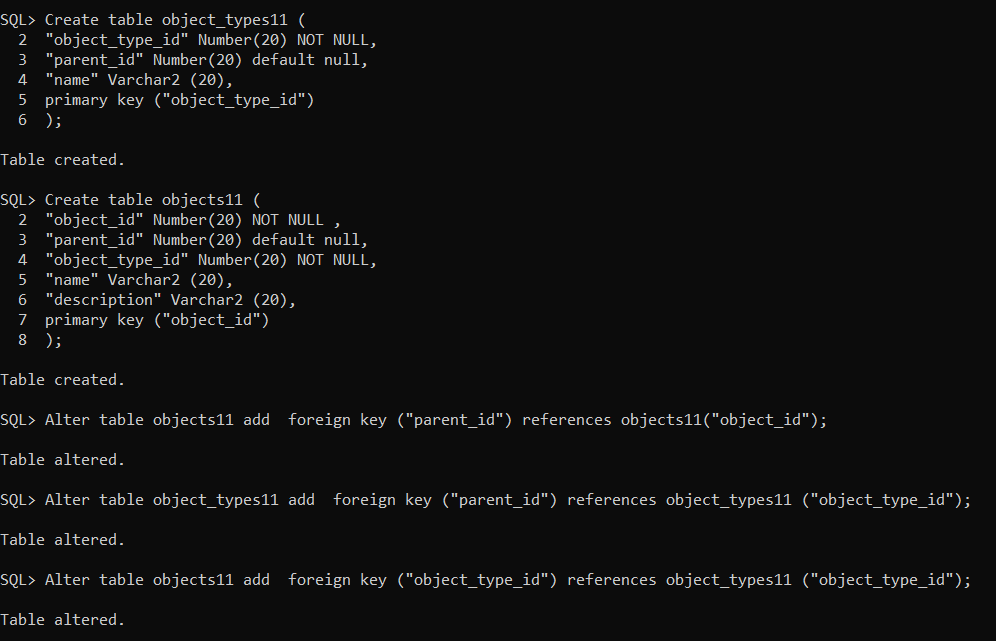
**);**

**Alter table objects11 add foreign key ("parent\_id") references objects11("object\_id");**

**Alter table object\_types11 add foreign key ("parent\_id") references object\_types11 ("object\_type\_id");**

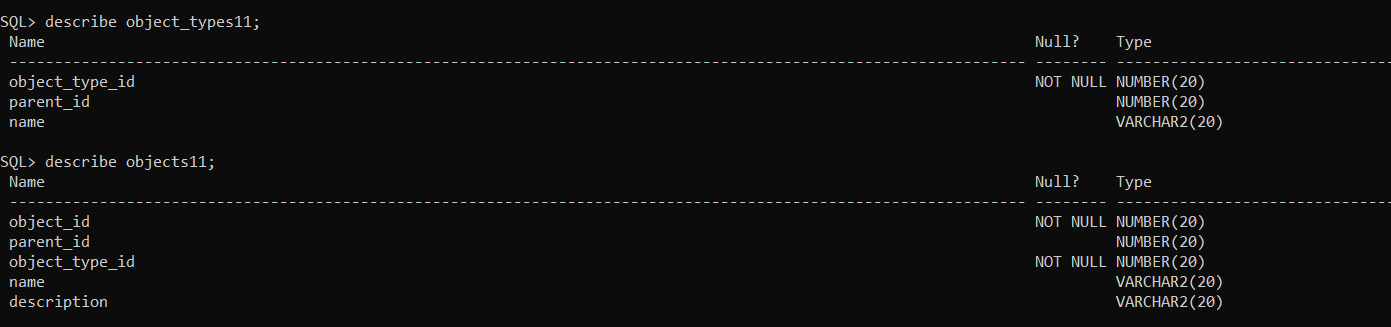
**Alter table objects11 add foreign key ("object\_type\_id") references object\_types11 ("object\_type\_id");**

****

****

**describe object\_types11;**

**describe objects11;**

****

5.2\*\*. Предметна область

Опис предметної області: місто, вулиця, будинок, обладнання.

Дані таблиці Objects описуються деревом:



Дані таблиці Object\_types описуються деревом:



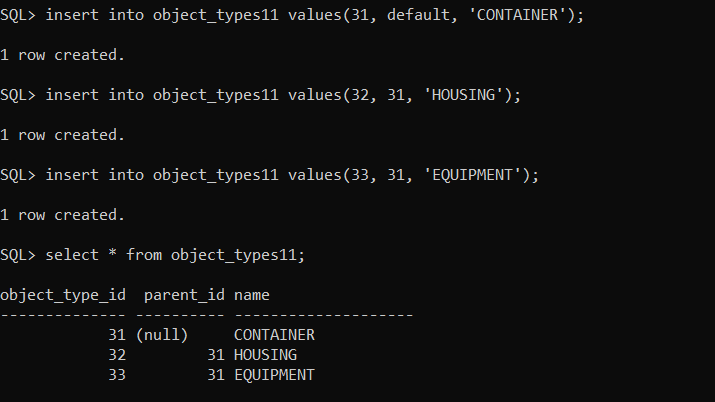
Напишіть скрипт для заповнення таблиць Objects і Object\_types.

**insert into object\_types11 values(31, default, 'CONTAINER');**

**insert into object\_types11 values(32, 31, 'HOUSING');**

**insert into object\_types11 values(33, 31, 'EQUIPMENT');**

**select \* from object\_types11;**



**insert into objects11 values(101, default, 32, 'SUMY', 'HOUSING');**

**insert into objects11 values(100, 101, 31, null, 'CONTAINER');**

**insert into objects11 values(102, 101, 32, 'SUPRUN STREET', 'HOUSING');**

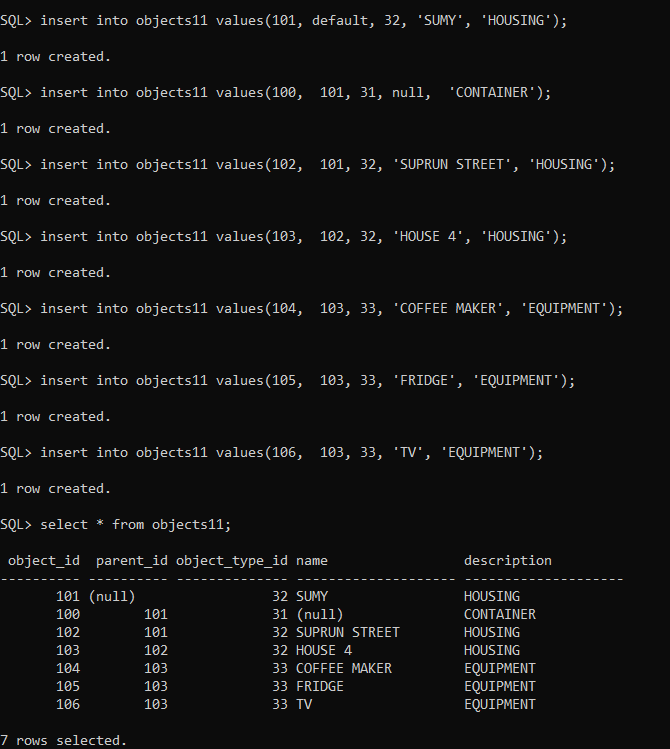
**insert into objects11 values(103, 102, 32, 'HOUSE 4', 'HOUSING');**

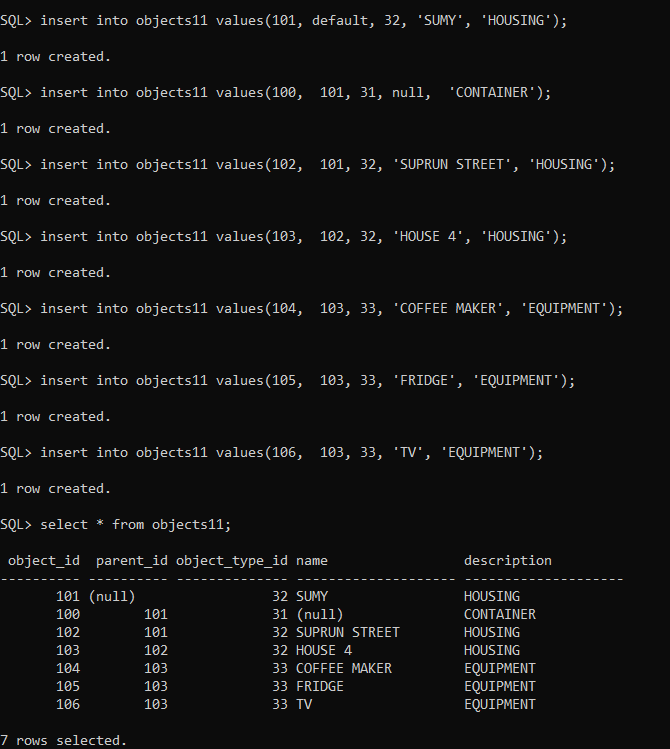
**insert into objects11 values(104, 103, 33, 'COFFEE MAKER', 'EQUIPMENT');**

**insert into objects11 values(105, 103, 33, 'FRIDGE', 'EQUIPMENT');**

**insert into objects11 values(106, 103, 33, 'TV', 'EQUIPMENT');**

**select \* from objects11;**

****



5.3\*\*. Завдання

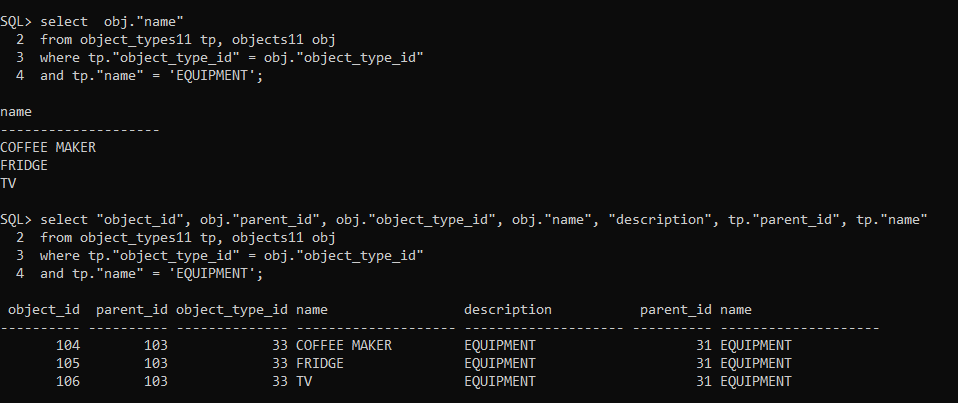
1. Виведіть назви об'єктів, що належать до типу «Обладнання»;

**select obj."name"**

**from object\_types11 tp, objects11 obj**

**where tp."object\_type\_id" = obj."object\_type\_id"**

**and tp."name" = 'EQUIPMENT';**

****

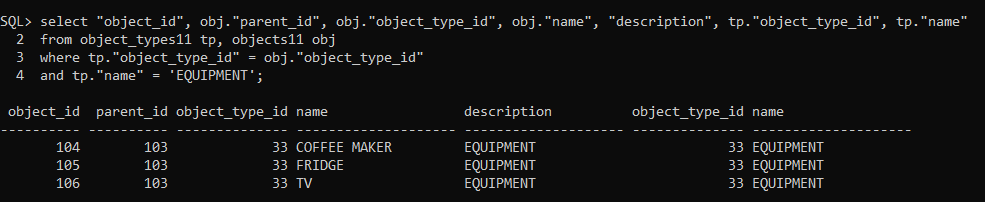
Перевірка:

**select "object\_id", obj."parent\_id", obj."object\_type\_id", obj."name", "description", tp."object\_type\_id", tp."name"**

**from object\_types11 tp, objects11 obj**

**where tp."object\_type\_id" = obj."object\_type\_id"**

**and tp."name" = 'EQUIPMENT';**

****

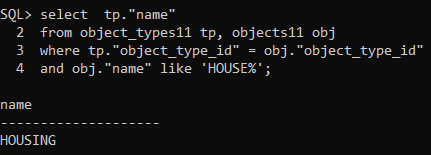
2. Виведіть назву типу об'єкта з ім'ям «Будинок»;

**select tp."name"**

**from object\_types11 tp, objects11 obj**

**where tp."object\_type\_id" = obj."object\_type\_id"**

**and obj."name" like 'HOUSE%';**

****

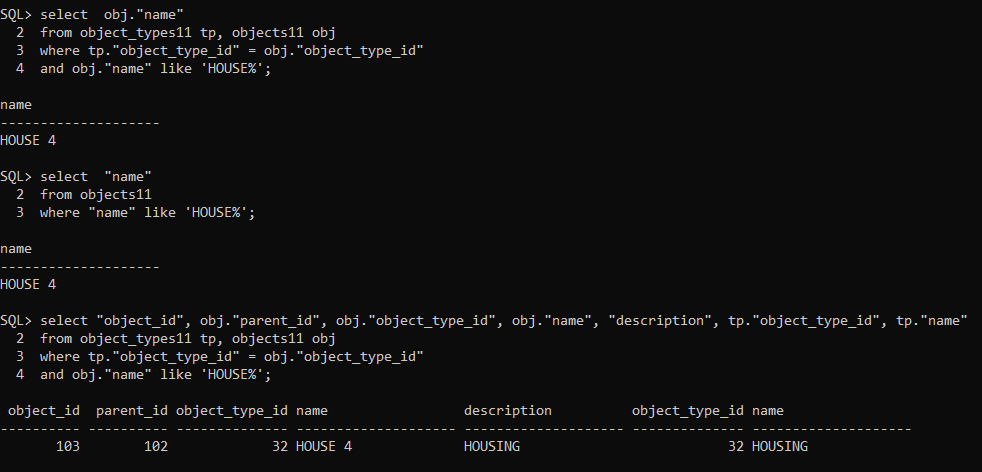
Перевірка:

**select "object\_id", obj."parent\_id", obj."object\_type\_id", obj."name", "description", tp."object\_type\_id", tp."name"**

**from object\_types11 tp, objects11 obj**

**where tp."object\_type\_id" = obj."object\_type\_id"**

**and obj."name" like 'HOUSE%';**

****

3. Виведіть назви об'єктів, які мають тип, що успадкований від типу «Контейнер»;

**select obj."name"**

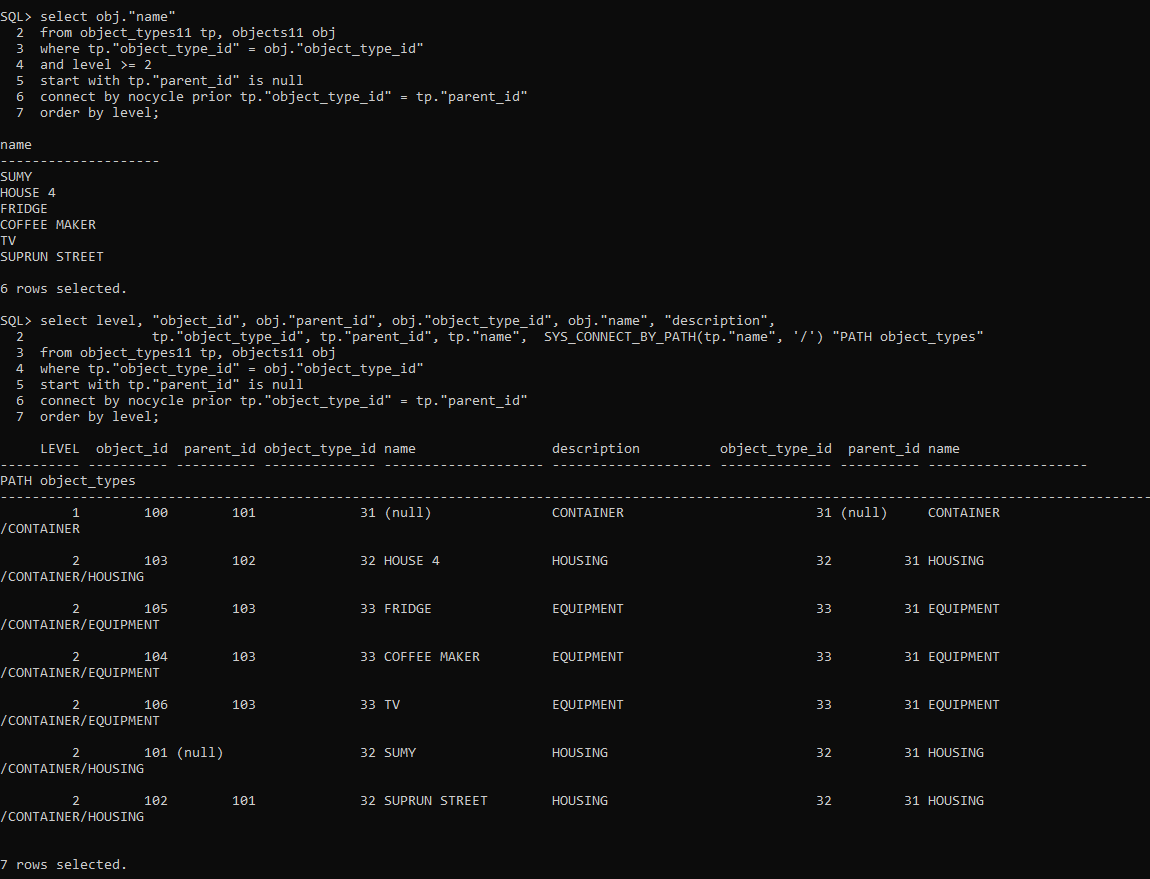
**from object\_types11 tp, objects11 obj**

**where tp."object\_type\_id" = obj."object\_type\_id"**

**and level >= 2**

**start with tp."parent\_id" is null**

**connect by nocycle prior tp."object\_type\_id" = tp."parent\_id"**

**order by level;**

Перевірка:

**select level, "object\_id", obj."parent\_id", obj."object\_type\_id", obj."name", "description",**

**tp."object\_type\_id", tp."parent\_id", tp."name", SYS\_CONNECT\_BY\_PATH(tp."name", '/') "PATH object\_types"**

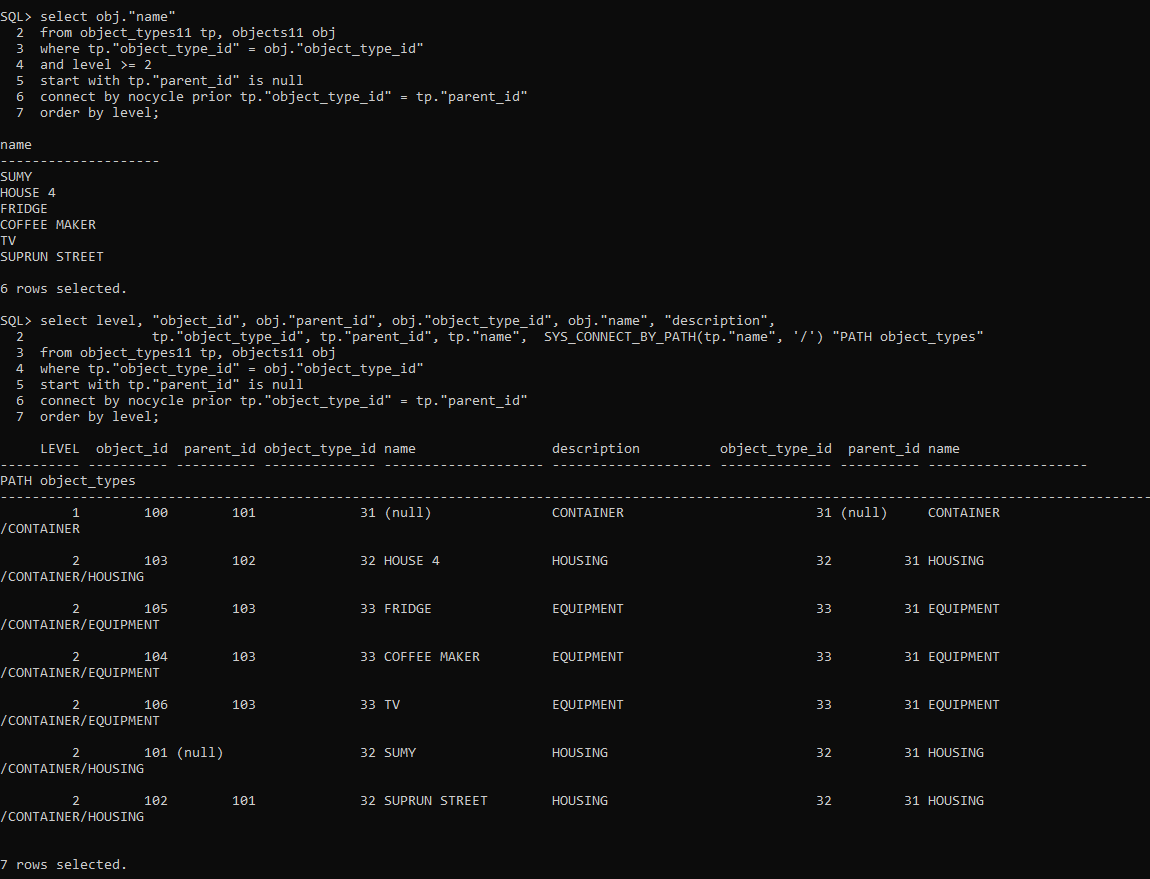
**from object\_types11 tp, objects11 obj**

**where tp."object\_type\_id" = obj."object\_type\_id"**

**start with tp."parent\_id" is null**

**connect by nocycle prior tp."object\_type\_id" = tp."parent\_id"**

**order by level;**



4. Вивести гілку дерева у вигляді: -> Суми -> вул. Супруна -> Будинок 4 -> Кавоварка.

**select SYS\_CONNECT\_BY\_PATH(obj."name", ' -> ') "PATH object name"**

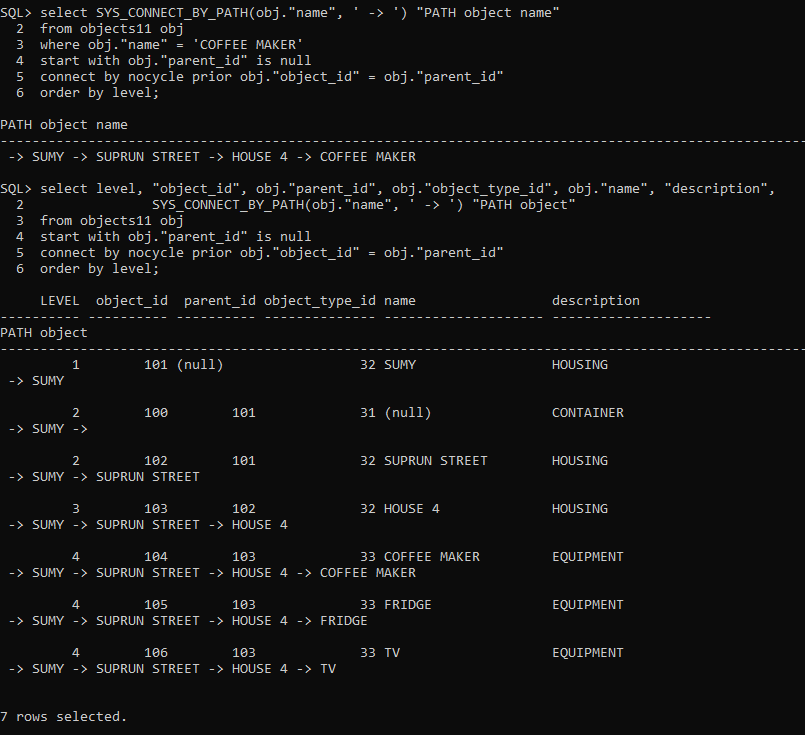
**from objects11 obj**

**where obj."name" = 'COFFEE MAKER'**

**start with obj."parent\_id" is null**

**connect by nocycle prior obj."object\_id" = obj."parent\_id"**

**order by level;**



Перевірка:

**select level, "object\_id", obj."parent\_id", obj."object\_type\_id", obj."name", "description",**

**SYS\_CONNECT\_BY\_PATH(obj."name", ' -> ') "PATH object"**

**from objects11 obj**

**start with obj."parent\_id" is null**

**connect by nocycle prior obj."object\_id" = obj."parent\_id"**

**order by level;**

