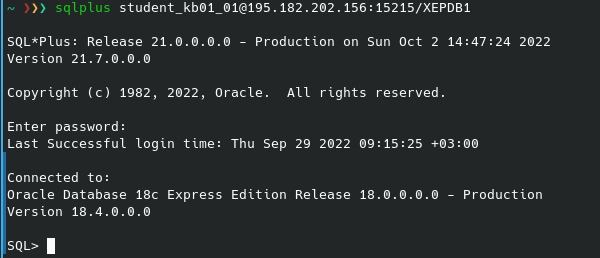
# Лабораторна робота 2

Завдання 1 Приєднатися до бази  
 Завантажте архів із sql\*plus, який додано завдання роботи, роз архівуйте та запустіть SQL.bat. Якщо Ви використовує ОС сімейства Linux скористайтесь інструкцією та адаптуйте файл для себе. У лабораторній роботі використовуються таблиці зі схеми SCOTT (див. рис 1.1). Сценарій їх створення та заповнення додано до роботи (init\_script.sql). За допомогою клієнта приєднайтеся до бази та виконайте цей сценарій.

Щоб підключитися до бази даних використаю наступну команду

sqlplus student\_kb01\_01@195.182.202.156:15215/XEPDB1

Рис. 1 – Успішне підключення до бази даних.

Після перевірки, що БД працює запустимо init\_script.sql додавши <start> секцію до нашої команди приєднання.

sqlplus [student\_kb01\_01@195.182.202.156](mailto:student_kb01_01@195.182.202.156):15215/XEPDB1 @Downloads/init\_script.sql

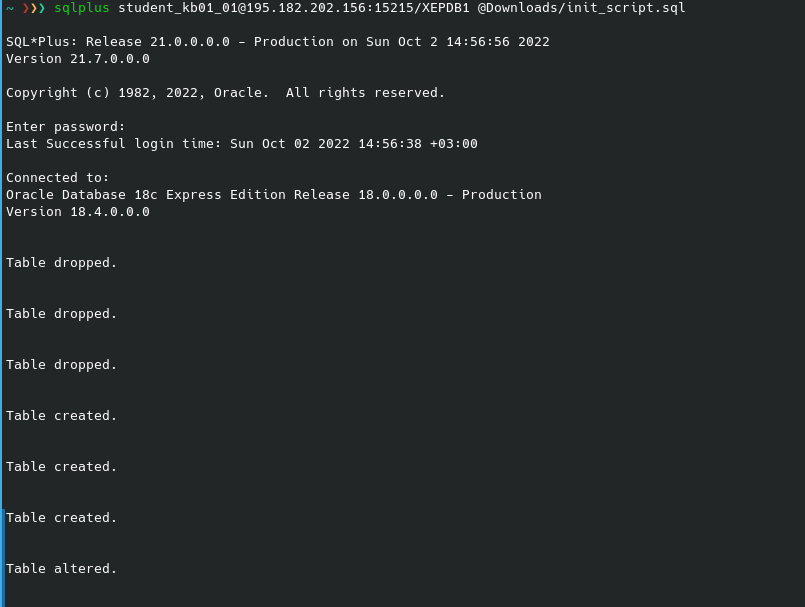
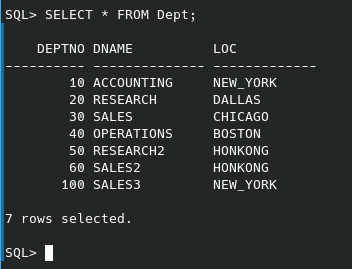


Рис. 2 – Запуск init скрипту.

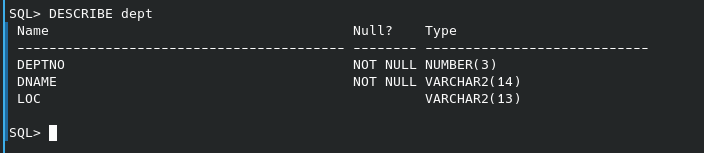
Завдання 2. Прості запити  
 Виконайте запити у консолі. Додайте у звіт текс завдання та скірншот результату. Якщо виводиться повідомлення про помилку «ні таблиці»- її потрібно заздалегідь створити. Для цього завантажте в середу і виконайте файл Init\_script.sql із попереднього завдання.

1. Виведемо всі записи з таблиці DEPT

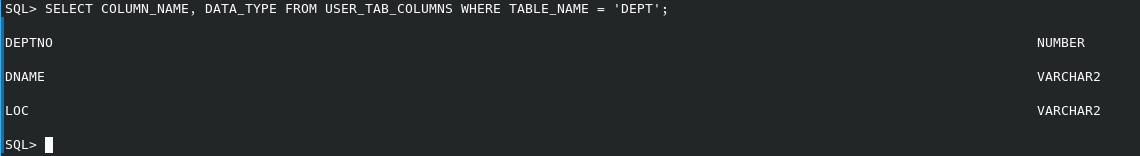
SELECT \* FROM Dept;



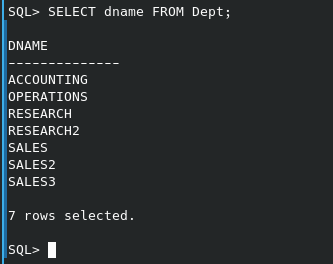
2. Переглянемо структуру таблиці DEPT  
Це команда SQL\*plus:  
 DESCRIBE dept



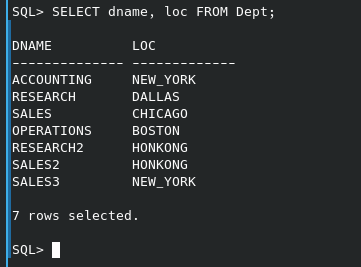
А так це можна зробити запитом:



3. Виведемо всі назви відділів з таблиці Dept  
 SELECT dname FROM Dept;

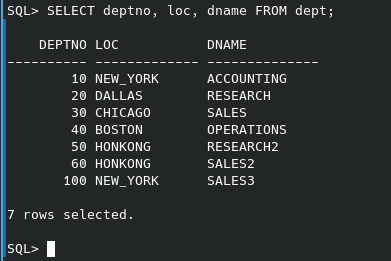


4. Виведемо всі назви відділів та їх розташування з таблиці Dept  
 SELECT dname, loc FROM Dept;



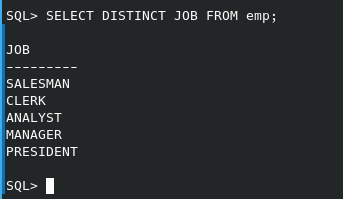
5. Зміна порядку колонок.  
SELECT [DISTINCT] {\*, column [alias],...}  
FROM table;  
 Запит із завдання 2.1 виводить стовбці у порядку deptno, dname, loc. Скоригуйте запит, так щоб він виводив стовпці в порядку deptno, loc, dname.

SELECT deptno, loc, dname FROM dept;



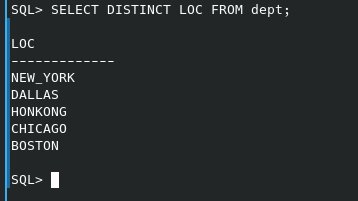
6. Запит із використанням DISTINCT  
SELECT DISTINCT {\*, column [alias],...}  
FROM table;  
6.1. Складіть запит, який поверне всі можливі посади співробітників без повторень

SELECT DISTINCT JOB FROM emp;



6.2. Складіть запит, який поверне усі можливі міста розміщення відділів

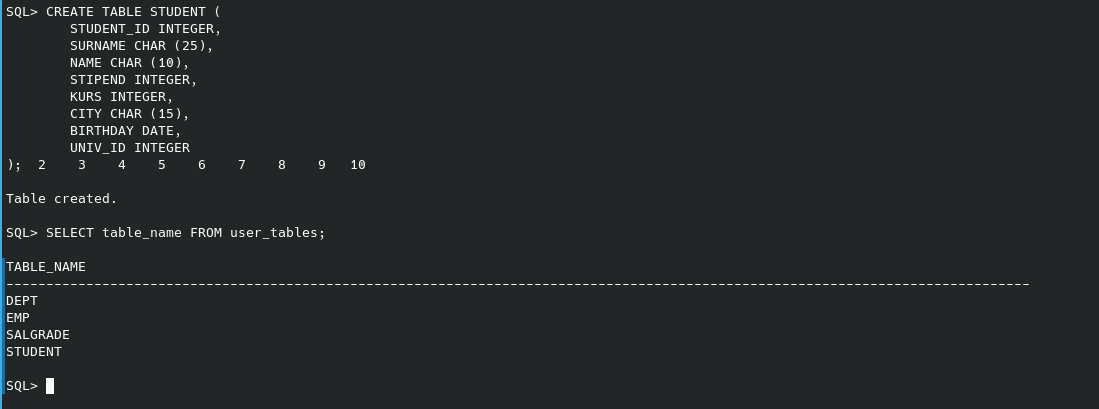
SELECT DISTINCT LOC FROM dept;



Завдання 4. Створення таблиці

Завдання  
CREATE TABLE STUDENT (  
 STUDENT\_ID INTEGER,  
 SURNAME CHAR (25),  
 NAME CHAR (10),  
 STIPEND INTEGER,  
 KURS INTEGER,  
 CITY CHAR (15),  
 BIRTHDAY DATE,  
 UNIV\_ID INTEGER  
);

Створення таблиці STUDENT:



Візьміть за основу запит на створення таблиці STUDENT і створіть таблицю STUDENT1:  
 Пара значень (Прізвище, Курс) є унікальною

Стипендія не може бути меншою за 5000

Student\_id не може бути порожнім.

За замовчанням UNIV\_ID - 0

Створення таблиці STUDENT1 відповідно до вказівок:

CREATE TABLE STUDENT1 (

STUDENT\_ID INTEGER NOT NULL,

SURNAME CHAR (25) UNIQUE,

NAME CHAR (10),

STIPEND INTEGER, CHECK(STIPEND>=5000),

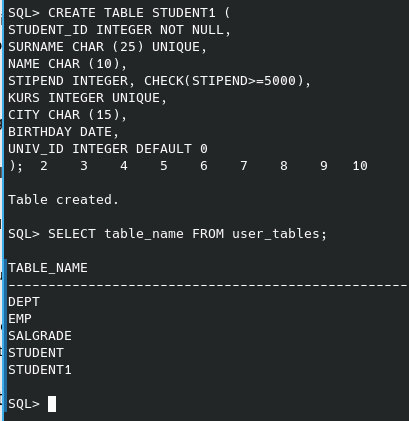
KURS INTEGER UNIQUE,

CITY CHAR (15),

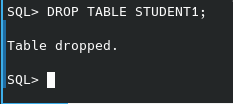
BIRTHDAY DATE,

UNIV\_ID INTEGER DEFAULT 0

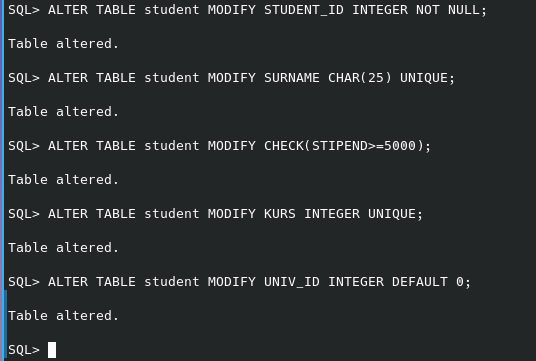
);



Видаліть таблицю Student1;

Видалення таблиці:

Завдання на «4» та «5». Таблиця STUDENT створена. Напишіть запити на її зміну (alter table). Додайте всі ті самі обмеження, що і у таблиці STUDENT1.

Запити на зміну таблиці student: