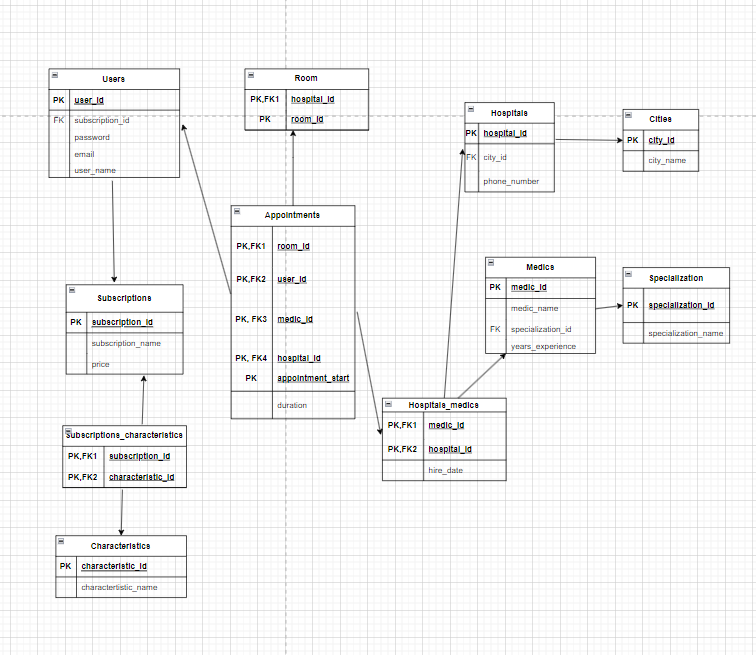
Lucan Cristian Stefan grupa 252 SGBD Tema 0

1. Identificați diagrama conceptuală definită în proiectul prezentat la materia Baze de Date din anul I. Dacă nu aveți o astfel de diagramă, atunci definiți un model simplificat pentru acest exercițiu.

a. adăugați ca poză diagrama conceptuală identificată;



b. adaptați cerințele exercițiilor 36 și 42 pentru diagrama conceptuală utilizată la

punctul a (formulați cerința în limbaj natural, apoi rezolvați cererea propusă în SQL).

Ex. 36 (“\_tema” la final):

SPOOL D:\an2\_sem1\SGBD\tema1\_36.sql;

SELECT 'DROP TABLE ' || table\_name || ';'

FROM user\_tables

WHERE table\_name LIKE upper('%\_tema');

SPOOL OFF;

Fisier tema1\_36.sql:

'DROPTABLE'||TABLE\_NAME||';'

--------------------------------------------------------------------------------------------------------------------------------------------

DROP TABLE APPOINTMENTS\_TEMA;

DROP TABLE CHARACTERISTICS\_TEMA;

DROP TABLE CITIES\_TEMA;

DROP TABLE HOSPITALS\_MEDICS\_TEMA;

DROP TABLE HOSPITALS\_TEMA;

DROP TABLE MEDICS\_TEMA;

DROP TABLE ROOMS\_TEMA;

DROP TABLE SPECIALIZATIONS\_TEMA;

DROP TABLE SUBSCRIPTIONS\_TEMA;

DROP TABLE SUBS\_CHARACT\_TEMA;

DROP TABLE USERS\_TEMA;

11 rows selected.

Ex. 42(“\_tema” la final)

SPOOL D:\an2\_sem1\SGBD\tema1\_42.sql;

SELECT

'INSERT INTO specializations\_tema VALUES

(' || specialization\_id || ', '|| specialization\_name || ')'

AS "Inserare date"

FROM specializations;

SPOOL OFF;

Fisier tema1\_42.sql

Inserare date

----------------------------------------------------------------------------------------------------------

INSERT INTO specializations\_tema VALUES

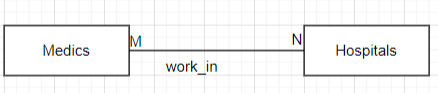
(1, ortoped)

2. Identificați în diagrama Entitate-Relație utilizată în proiectul prezentat la materia Baze de Date din anul I, o relație din tip many-to-many. Dacă nu aveți o astfel de relație în acest proiect, atunci definiți una folosind tema aleasă în acel proiect:

a.descrieți entitățile și relația dintre acestea;

Intre tabelele **Medics** si **Hospitals** exista o relatie many-to-many, deoarece un spital poate avea mai multi medici, iar un medic poate lucra la mai multe spitale.

b. Realizați diagrama Entitate-Relație doar pentru această parte din proiect;



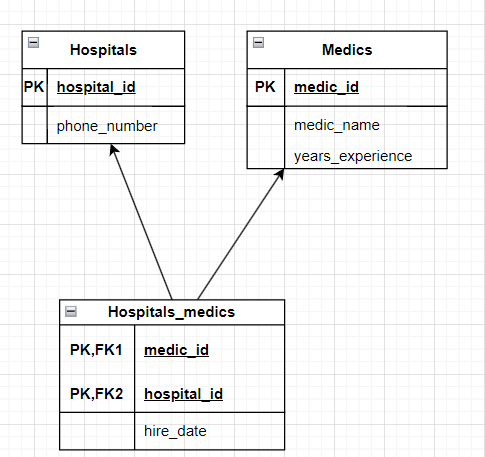
c. Descrieți modul de transformare al acestei relații în diagrama conceptuală,

precizând toate cheile primare, cheile externe și alte atribute esențiale;

In diagrama conceptuala va aparea un al treilea tabel, **Hospitals\_medics**

Care va avea o cheie primara compusa din cheia primara din **Medics** si cheia primara din **Hospitals**, fiecare dintre acestea fiind si cheie externa ce isi referentiaza tabelul din care provine (medic\_id referentiaza **Medics** si hospital\_id referentiaza **Hospitals**). O alta data importanta ce va apartine tabelului nou este hire\_date care descrie data angajarii medicului in respectivul spital.

d. realizați diagrama conceptuală doar pentru această parte din proiect;



e. pe baza diagramei conceptuale de la punctul d, definiți în SQL tabelele și toate

constrângerile necesare;

CREATE TABLE hospitals\_t(

hospital\_id NUMBER,

phone\_number char(10) NOT NULL UNIQUE,

CONSTRAINT hospital\_pk\_t PRIMARY KEY(hospital\_id)

);

CREATE TABLE medics\_t(

medic\_id NUMBER,

medic\_name VARCHAR2(20) NOT NULL UNIQUE,

years\_exp NUMBER NOT NULL,

CONSTRAINT medic\_pk\_t PRIMARY KEY(medic\_id)

);

CREATE TABLE hospitals\_medics\_t(

hospital\_id NUMBER,

medic\_id NUMBER,

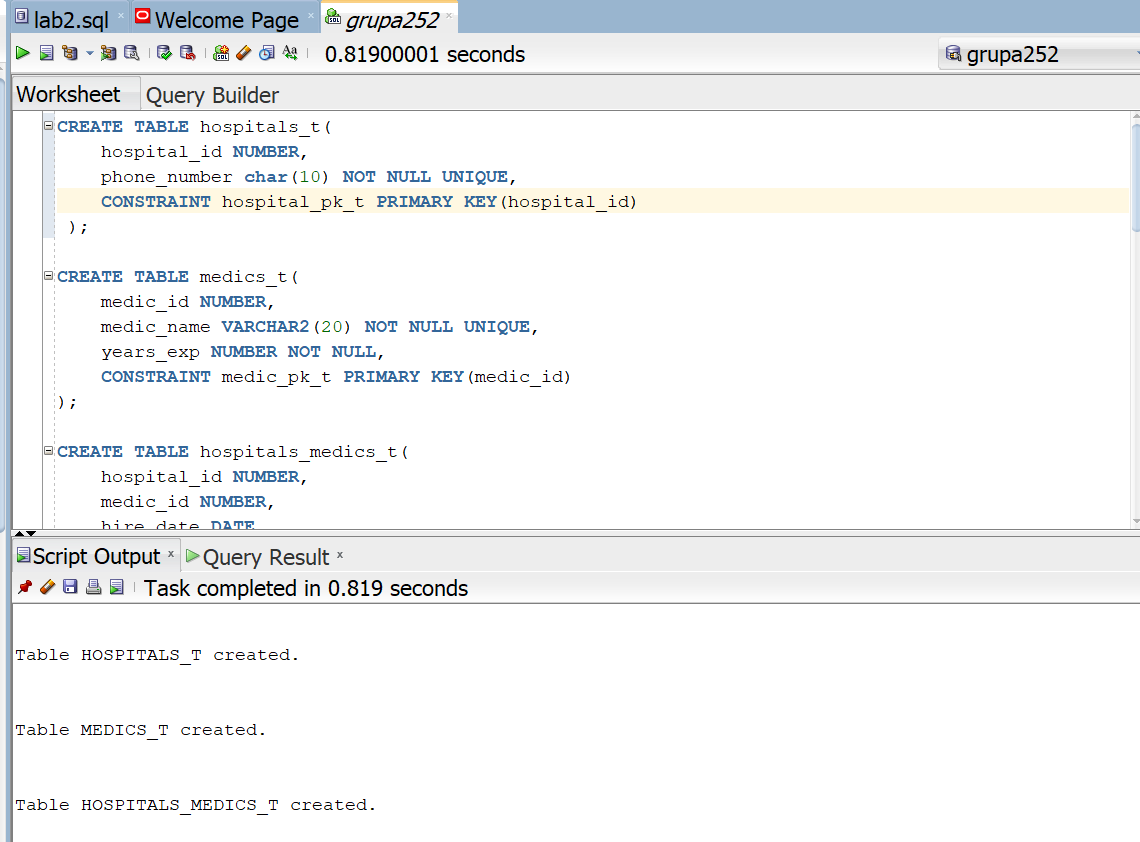
hire\_date DATE,

CONSTRAINT hospitals\_medics\_pk\_t PRIMARY KEY(hospital\_id, medic\_id),

CONSTRAINT hosp\_med\_hosp\_fk\_t FOREIGN KEY(hospital\_id) REFERENCES hospitals\_t(hospital\_id),

CONSTRAINT hosp\_med\_med\_fk\_t FOREIGN KEY(medic\_id) REFERENCES medics\_t(medic\_id)

);



f. adaptați una dintre cerințele exercițiilor 25-34 pentru diagrama obținută la punctul d (formulați cerința în limbaj natural, inserați 5-10 înregistrări în fiecare tabelă utilizată apoi rezolvați cererea propusă în SQL).

Inserari:

INSERT ALL

INTO hospitals VALUES (1, '1111111111')

INTO hospitals VALUES (2, '2222222222')

INTO hospitals VALUES (3, '3333333333')

INTO hospitals VALUES (4, '4444444444')

INTO hospitals VALUES (5, '5555555555')

SELECT 1 FROM DUAL;

INSERT ALL

INTO medics VALUES (1, 'ALEX', 1)

INTO medics VALUES (2, 'Andrew', 2)

INTO medics VALUES (3, 'Joe', 2)

INTO medics VALUES (4, 'Albert', 5)

INTO medics VALUES (5, 'Chris', 5)

SELECT 1 FROM DUAL;

INSERT ALL

INTO hospitals\_medics VALUES (1, 1, SYSDATE)

INTO hospitals\_medics VALUES (1, 3, SYSDATE)

INTO hospitals\_medics VALUES (2, 3, SYSDATE)

INTO hospitals\_medics VALUES (2, 2, SYSDATE)

INTO hospitals\_medics VALUES (2, 4, SYSDATE)

INTO hospitals\_medics VALUES (3, 5, SYSDATE)

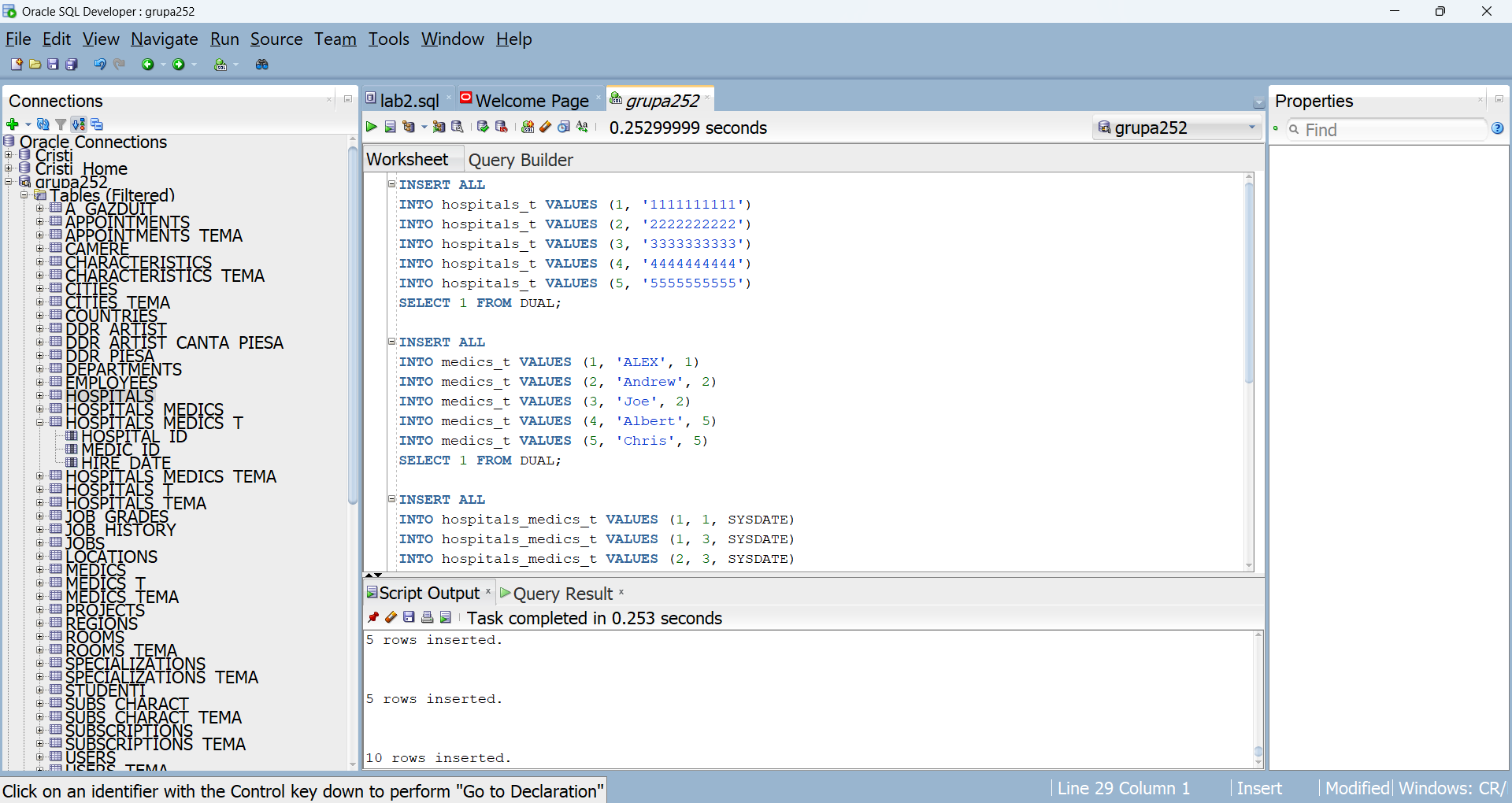
INTO hospitals\_medics VALUES (4, 4, SYSDATE)

INTO hospitals\_medics VALUES (4, 5, SYSDATE)

INTO hospitals\_medics VALUES (5, 1, SYSDATE)

INTO hospitals\_medics VALUES (3, 2, SYSDATE)

SELECT 1 FROM DUAL;



Adaptare 26: Să se obţină numărul spitalelor care au cel puţin 2 medici.

SELECT COUNT(\*)

FROM (SELECT hospital\_id, COUNT(medic\_id)

FROM hospitals\_medics

GROUP BY hospital\_id HAVING COUNT(medic\_id) >= 2);

