

Πανεπιστήμιο Ιωαννίνων
Τμήμα Μαθηματικών
Μάθημα: Βάσεις Δεδομένων (741)
Εργαστηριακό Τεστ

21/12/2004

Έχουμε την βάση της σχολής που αποτελείται από τους παρακάτω πίνακες.

Όνομα πίνακα: Students	
Γνώρισμα	Τύπος Δεδομένων
Name	Συμβολοσειρά 20 χαρακτήρων
Surname	Συμβολοσειρά 40 χαρακτήρων
Id	Ακέραιος
Start_date	Ημερομηνία
Primary key:	Id

Όνομα πίνακα: Lessons	
Γνώρισμα	Τύπος Δεδομένων
Name	Συμβολοσειρά 20 χαρακτήρων
Code	Μικρός ακέραιος
Credits	Προκαθορισμένες τιμές: 3 ή 4 ή 5
Primary key:	Code

Όνομα πίνακα: Grades	
Γνώρισμα	Τύπος Δεδομένων
Id φοιτητή	Ακέραιος
Code μαθήματος	Μικρός Ακέραιος
Grade	Οι τιμές από 0 έως 10
Primary key:	Id, Code

Γίνεται καταχώρηση δεδομένων και έχουμε:

Students			
Όνομα	Επώνυμο	Αριθμός μητρώου	Ημερομηνία
Γιάννης	Καραμήτρος	4100	1/10/1995
Μανώλης	Βελάς	6700	1/10/1999
Άρης	Τερζός	7005	1/10/2000

Lessons		
Όνομα	Κωδικός	Διδακτικές μονάδες
Βάσεις Δεδομένων	741	3
Προγραμματισμός Λογικής	544	3
Απειροστικός Λογισμός I	111	5

Grades		
Αριθμός μητρώου	Κωδικός μαθήματος	Βαθμός
4100	741	7
6700	741	5
6700	544	4
7005	111	6
7005	741	5
4100	544	8
4100	111	3
6700	111	7
7005	544	9

Ερωτήματα:

- 1) i) Δημιουργήστε μια όψη για την εύρεση των φοιτητών που μπήκαν στην σχολή από το 1995 έως και το 1999.
ii) Δημιουργήστε μια όψη για την εύρεση των μαθημάτων με 3 διδακτικές μονάδες.
(1 μονάδα)

```
CREATE VIEW STUD_1995_1999(
    NAME,
    SURNAME,
    ID,
    START_DATE)
AS
select * from students
where start_date between '1.1.1995' and '1.1.1999'
```

- 2) Δημιουργήστε μια όψη για την εύρεση των αριθμών μητρώων των φοιτητών και των ονομάτων των μαθημάτων που έχουν περάσει οι φοιτητές.
Προσοχή: Ορίστε την όψη με τέτοιο τρόπο ώστε να έχουμε σαν όνομα της στήλης των αριθμών μητρώων την λέξη ΜΗΤΡΩΟ και σαν όνομα της στήλης μαθημάτων, ΜΑΘΗΜΑΤΑ.
(1 μονάδα)

```
CREATE VIEW MITROA_MATHIMATA(
MITROO,
MATHIMATA)
AS
select id,name from grades,lessons
where grades.code=lessons.code and grade>=5
```

- 3) Δημιουργήστε μια όψη για την εύρεση των αριθμών μητρώων των φοιτητών και την βαθμολογία των φοιτητών στα μαθήματα με 5 διδακτικές μονάδες.
(1 μονάδα)

```
CREATE VIEW BATHMOLOGIA_5_CREDITS(
ID,
GRADE)
AS
select grades.id, grade from grades,lessons
where grades.code=lessons.code and credits=5
```

- 4) Δημιουργήστε μια trigger για τον πίνακα Grades, για πριν την εισαγωγή δεδομένων ώστε κατά την εισαγωγή δεδομένων στον πίνακα αν δεν υπάρχει καθόλου βαθμός να καταχωρείται το μηδέν για βαθμός.
(1 μονάδα)

```
CREATE TRIGGER GRADES_BI0 FOR GRADES
ACTIVE BEFORE INSERT POSITION 0
AS
begin
if(new.grade is NULL) then
new.grade=0;
End
```

- 5) Δημιουργήστε μια trigger για τον πίνακα Grades, για την ενημέρωση δεδομένων ώστε κατά την ενημέρωση δεδομένων στον πίνακα Grades, να γίνεται είτε ενημέρωση κανονικά, ή να εμφανίζει μήνυμα λάθους αν ο φοιτητής δεν έχει δηλώσει το μάθημα (αν δεν υπάρχει εγγραφή στον πίνακα Grades).
(1 μονάδα)

```
CREATE TRIGGER GRADES_BU0 FOR GRADES
ACTIVE BEFORE UPDATE POSITION 1
AS
begin
if(exists(select 1 from grades where code=new.code and id=new.id)) then
begin
```

```
end
else
exception none;
end
```

- 6) Δημιουργήστε μια trigger ώστε όταν σβήνετε έναν φοιτητή από τον πίνακα Students, να σβήνονται και όλοι οι βαθμοί του από τον πίνακα Grades.
(1 μονάδα)

```
CREATE TRIGGER STUDENTS_AD0 FOR STUDENTS
ACTIVE AFTER DELETE POSITION 0
AS
begin
delete from grades where id=students.id;
end
^
```

- 7) i) Δημιουργήστε μια stored procedure που να έχει ως δεδομένα εισόδου τον αριθμό μητρώου ενός φοιτητή και να εμφανίζει τα στοιχεία του.
ii) Επίσης δημιουργήστε άλλες δύο stored procedures που να εισάγουν και να διαγράφουν δεδομένα από τον πίνακα Students αντίστοιχα.
(1 μονάδα)

```
i)
CREATE PROCEDURE STOIXEIA (
  IDA INTEGER)
RETURNS (
  RNAME VARCHAR(20),
  RSURNAME VARCHAR(40),
  IDN INTEGER,
  RSTARTS_DATE DATE)
AS
begin
select name,surname,id,start_date from students where id=:ida into
rname,rsurname,idn,rstarts_date;
suspend;
end
```

```
ii)
CREATE PROCEDURE INSERTT (
  NAMEA VARCHAR(20),
  SURA VARCHAR(40),
  IDA INTEGER,
```

```

    STARTA DATE)
AS
begin
insert into students values(:namea,:sura,:ida,:starta);
    suspend;
end

```

```

CREATE PROCEDURE DELETEE (
    NAMEA VARCHAR(20),
    SURNAMEA VARCHAR(40),
    IDA INTEGER,
    STARTA DATE)
AS
begin
delete from students where name=:namea and surname=:surnamea and id=:ida and
start_date=:starta;
    suspend;
end

```

- 8) Δημιουργήστε μια stored procedure που να πληκτρολογείτε τον αριθμό μητρώου του φοιτητή και να εμφανίζεται ένα μήνυμα που να πληροφορεί τον χρήστη για τον αριθμό των μαθημάτων των οποίων έχει περαστεί η βαθμολογία στον πίνακα Grades (δεν μας ενδιαφέρει αν ο φοιτητής έχει περάσει το μάθημα).
(1.5 μονάδα)

```

CREATE EXCEPTION TWO 'Έχει δηλώσει δύο μαθήματα';

```

```

CREATE EXCEPTION NONE 'Δεν έχει δηλώσει κανένα μάθημα';

```

```

CREATE EXCEPTION ONE 'Έχει δηλώσει ένα μάθημα';

```

```

CREATE EXCEPTION THREE 'Έχει δηλώσει τρία μαθήματα';

```

```

CREATE PROCEDURE HOW_MANY (
    IDN INTEGER)
AS
DECLARE VARIABLE COUNT_LESSONS INTEGER;
begin
select count(id) from grades where id=:idn into count_lessons;
if(count_lessons=0) then
exception none;
if(count_lessons=1) then
exception one;

```

```
if(count_lessons=2) then  
exception two;  
if(count_lessons=3) then  
exception three;  
suspend;  
end
```

- 9) i) Δώστε τα δικαιώματα SELECT, INSERT για τον πίνακα Students στον χρήστη JOHN.
ii) Έπειτα δώστε το δικαίωμα SELECT στην πρώτη stored procedure του ερωτήματος (7).
(0.5 μονάδα)

i)
Grant select, insert on students to JOHN;

ii)
grant select on student to STOIXEIA;

- 10) i) Δημιουργήστε έναν ρόλο με το όνομα ΕΙΣΑΓΩΓΗ.
ii) Έπειτα δώστε σε αυτόν τον ρόλο το δικαίωμα INSERT, UPDATE, DELETE για τον πίνακα STUDENTS.
iii) Δώστε στον χρήστη JOHN το δικαίωμα του ρόλου ΕΙΣΑΓΩΓΗ Με την δυνατότητα να μπορεί ο χρήστης να μεταφέρει τον ρόλο και σε άλλους χρήστες.
(1 μονάδα)

i)
create Role EISAGOGI;

ii)
grant insert, update, delete on students to EISAGOGI;

iii)
grant eisagogi to john with admin option;