

Βάσεις Δεδομένων II (Ε)

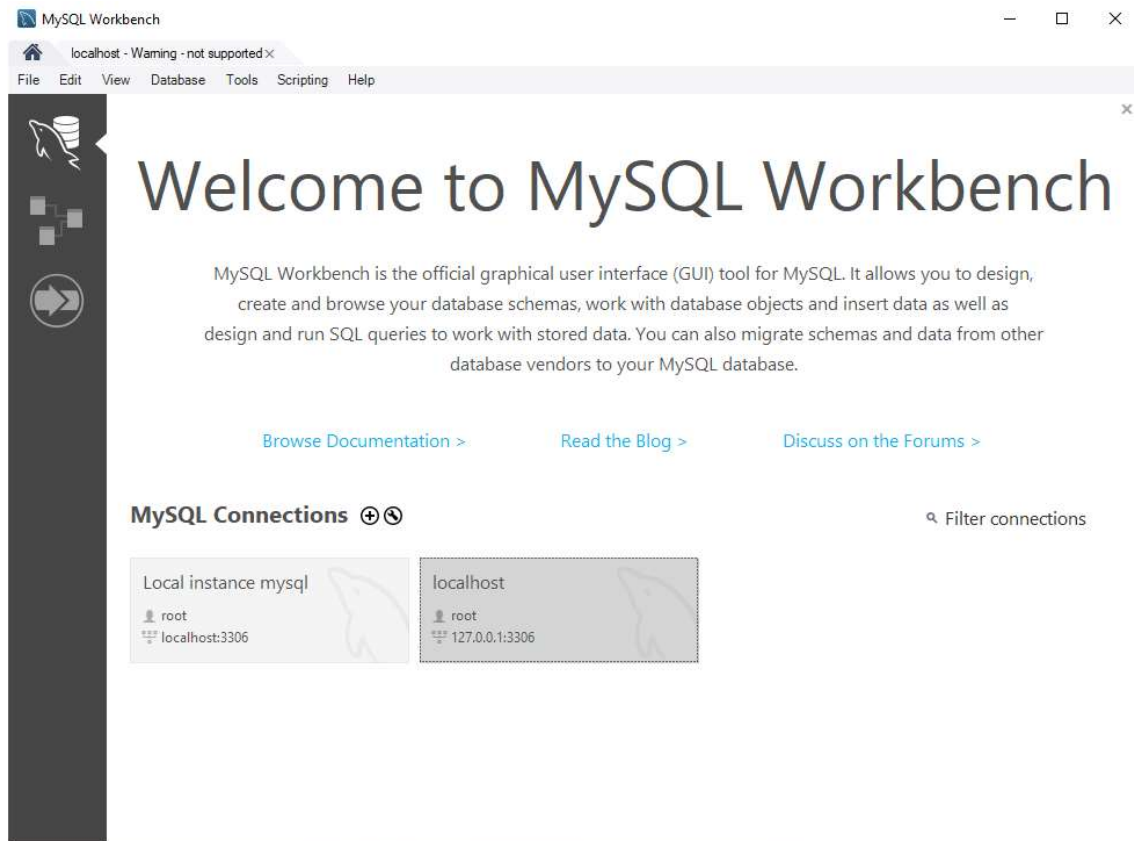
A.M.: ice18390094

Ονοματεπώνυμο: Άγγελος Τζώρτζης

Εργαστηριακή άσκηση 1

Τμήμα: [00] Χωρίς παρακολούθηση

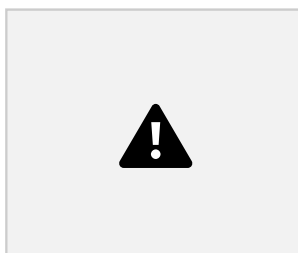
1. Σύνδεση στην MySQL του συστήματος μας:



2. Έλεγχος αν υπάρχει ήδη η ΒΔ personnel και διαγραφή της αν υπάρχει και δημιουργία της εκ νέου:

```
show databases;  
drop database if exists personnel;  
create database personnel;
```

Εδώ φαίνονται οι ΒΔ του συστήματος μας:



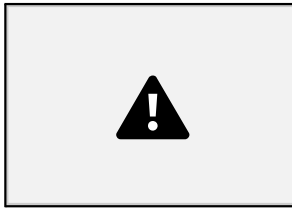
3. Επιλογή της personnel για χρήση:

```
use personnel;
```

4. Εμφανίζουμε τους πίνακες του personnel και τους διαγράφουμε εάν υπάρχουν:

```
show tables;
```

Εδώ φαίνονται οι πίνακες στην ΒΔ personnel:



Εφόσον δημιουργήσαμε την ΒΔ personnel εκ νέου δεν υπάρχουν πίνακες μέσα σε αυτήν για να διαγράψουμε.

5. Δημιουργία των πινάκων DEPT, JOB, EMP χωρίς την προσθήκη κλειδιών. Λαμβάνονται υπόψη οι τύποι δεδομένων από τον πίνακα 1.

```
create table DEPT(  
    DEPTNO numeric(2),  
    DNAME varchar(24),  
    LOC char(23)  
);  
create table JOB(  
    JOBCODE numeric(3),  
    JOB_DESCR varchar(24),  
    SAL numeric(10, 2)  
);  
create table EMP(  
    EMPNO numeric(4),  
    ENAME varchar(24),  
    JOBNO numeric(3),  
    DEPTNO numeric(2),  
    COMM numeric(10, 2)  
);
```

```
describe DEPT;  
describe JOB;  
describe EMP;
```

Δομή των πινάκων:

DEPT:



JOB:



EMP:



Οι δομές των πινάκων είναι όπως περιμέναμε και τα δεδομένα σύμφωνα με τον πίνακα 1.

6. Εισαγωγή των δεδομένων στους πίνακες:

```
insert into DEPT (DEPTNO, DNAME, LOC) values  
  (50, 'SALES', 'ATHENS'),  
  (60, 'ACCOUNTING', 'ATHENS'),  
  (70, 'PAYROLL', 'VOLOS');
```

```
insert into JOB (JOBCODE, JOB_DESCR, SAL) values  
  (100, 'SALESMAN', 2000),  
  (200, 'ANALYST', 2000),  
  (300, 'DBA', 3000);
```

```
insert into EMP (EMPNO, ENAME, JOBNO, DEPTNO, COMM) values  
  (10, 'CODD', 100, 50, NULL),  
  (20, 'NAVATHE', 200, 50, 450),  
  (30, 'ELMASRI', 300, 60, NULL),  
  (40, 'DATE', 100, 50, NULL);
```

```
select * from DEPT;  
select * from JOB;  
select * from EMP;
```

Περιεχόμενα των πινάκων:

DEPT:



JOB:



EMP:

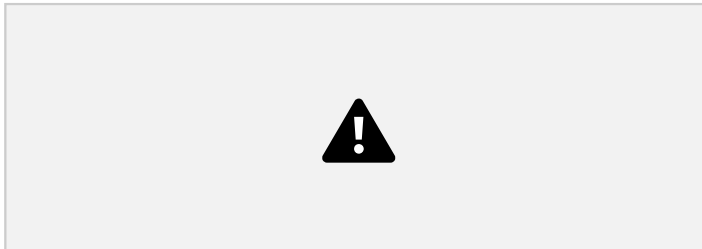


Τα περιεχόμενα των πινάκων είναι όπως τα περιμέναμε.

7. Προσθήκη στον πίνακα EMP τη στήλη DATE με τύπο δεδομένων HIREDATE:

```
alter table EMP add HIREDATE date;  
describe EMP;
```

Νέα δομή του πίνακα EMP:



Προστέθηκε το νέο πεδίο στον ήδη υπάρχοντα πίνακα.

8. Ενημέρωση της στήλης HIREDATE με τις ημερομηνίες πρόσληψης των υπαλλήλων:

```
update EMP set HIREDATE = '2001-01-10' where EMPNO = 10;  
update EMP set HIREDATE = '1999-02-25' where EMPNO = 20;  
update EMP set HIREDATE = '2000-03-17' where EMPNO = 30;  
update EMP set HIREDATE = '1989-06-07' where EMPNO = 40;
```

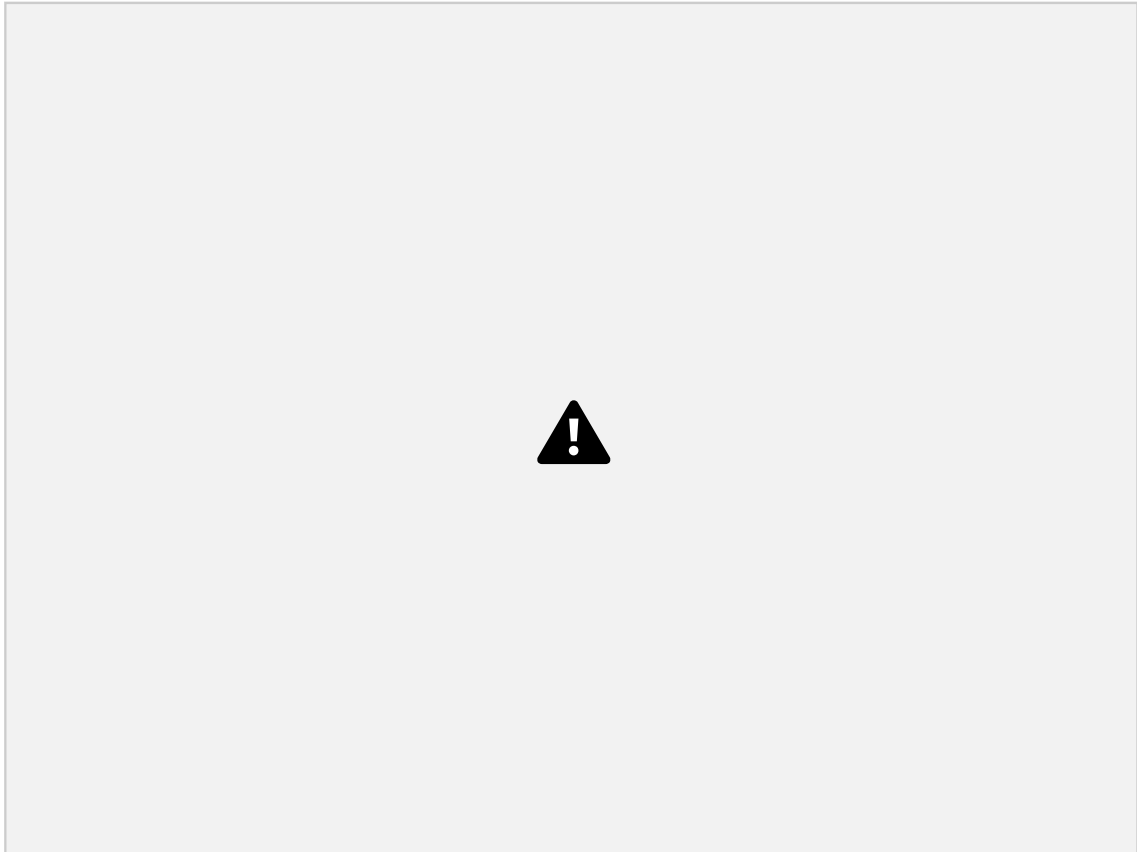
```
select * from EMP;
```

Πίνακας EMP με τα νέα δεδομένα:

	EMPNO	ENAME	JOBNO	DEPTNO	COMM	HIREDATE
▶	10	CODD	100	50	NULL	2001-01-10
	20	NAVATHE	200	50	450.00	1999-02-25
	30	ELMASRI	300	60	NULL	2000-03-17
	40	DATE	100	50	NULL	1989-06-07

Τα νέα δεδομένα στην στήλη HIREDATE είναι σωστά.

Για να μπορούν να γίνουν τα update έπρεπει να απενεργοποιήσω την εξής επιλογή στο MySQL workbench:



9. Ορίζουμε τον τύπο δεδομένων της στήλης JOB_DESCR σε varchar(30).

```
alter table JOB modify JOB_DESCR varchar(30);  
describe JOB;
```

Νέα δομή του πίνακα JOB:



Βλέπουμε ότι έχει ενημερωθεί ο τύπος δεδομένων της στήλης JOB_DESCR.

10. Κατάργηση της στήλης LOC από τον πίνακα DEPT:

```
alter table DEPT drop LOC;  
describe DEPT;
```

Νέα δομή του πίνακα DEPT:



Βλέπουμε ότι έχει διαγραφεί η στήλη LOC.

11. Μετονομασία της στήλης DNAME σε DEPT_NAME και αλλαγή σε τύπο δεδομένων varchar(25) σε μία εντολή:

```
alter table DEPT change DNAME DEPT_NAME varchar(25);  
describe DEPT;
```

Νέα δομή του πίνακα DEPT:



Βλέπουμε ότι έχει αλλάξει το όνομα και ο τύπος δεδομένων της στήλης DNAME.

12. Πρόσθεση του περιορισμού πρωτεύοντος κλειδιού στον πίνακα DEPT:

```
alter table DEPT add constraint pk_DEPTNO primary key (DEPTNO);  
describe DEPT;
```

Νέα δομή του πίνακα DEPT:



Βλέπουμε ότι η στήλη DEPTNO έχει την ιδιότητα primary key και ότι πλέον δεν μπορεί να έχει και την τιμή null καθώς αυτή είναι ιδιότητα των πρωτεύοντων κλειδιών.

13. Κατάργηση του περιορισμού πρωτεύοντος κλειδιού στον πίνακα DEPT:

```
alter table DEPT drop primary key;  
describe DEPT;  
show create table DEPT;
```

Νέα δομή του πίνακα DEPT:



Βλέπουμε ότι η στήλη DEPTNO δεν έχει την ιδιότητα primary key πλέον.

Εμφάνιση εντολής create table για τον πίνακα DEPT (μπορεί να χρειαστεί μεγέθυνση για να φανεί καθαρά η εικόνα):



14. Προσθήκη περιορισμού πρωτεύοντος κλειδιού και περιορισμούς ξένου κλειδιού σε όλους τους πίνακες:

```
alter table DEPT add constraint pk_DEPTNO primary key (DEPTNO);  
alter table JOB add constraint pk_JOBCODE primary key (JOBCODE);  
alter table EMP add constraint pk_EMPNO primary key (EMPNO);  
alter table EMP add constraint fk_JOBNO foreign key (JOBNO) references  
JOB(JOBCODE);  
alter table EMP add constraint fk_DEPTNO foreign key (DEPTNO)  
references DEPT(DEPTNO);
```

```
describe DEPT;  
describe JOB;  
describe EMP;
```

Εμφάνισης της νέας δομής των πινάκων:
DEPT:



JOB:



EMP:

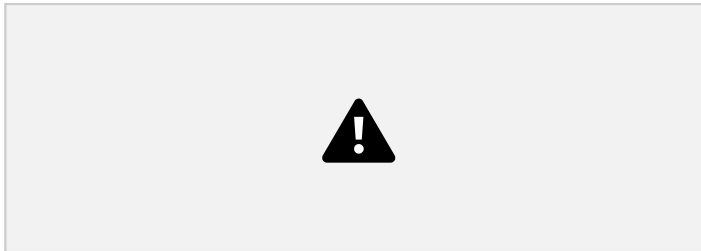


Βλέπουμε την προσθήκη πρωτεύοντος κλειδιού σε κάθε πίνακα καθώς και ξένου κλειδιού στις στήλες που προέρχονται από άλλο πίνακα.

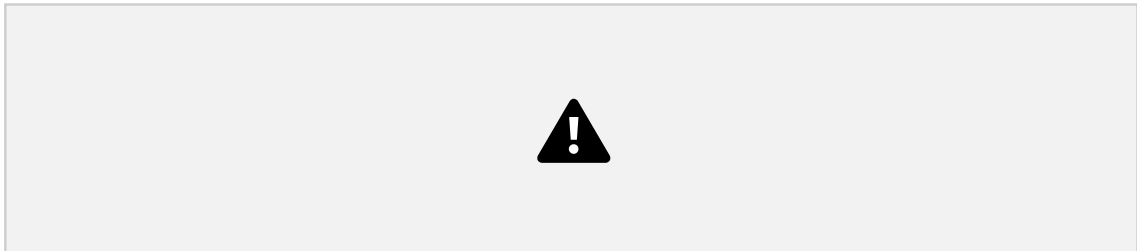
15. Κατάργηση περιορισμών ξένων κλειδιών στον πίνακα EMP:

```
alter table EMP drop foreign key fk_JOBNO;  
alter table EMP drop foreign key fk_DEPTNO;  
describe EMP;  
show create table EMP;
```

Νέα δομή του πίνακα EMP:



Εμφάνιση εντολής create table για τον πίνακα DEPT (μπορεί να χρειαστεί μεγέθυνση για να φανεί καθαρά η εικόνα):



Βλέπουμε ότι δεν υπάρχει στήλη με την ιδιότητα foreign στον πίνακα.

16. Προσθέτουμε DEFAULT τιμή στον πίνακα DEPT, θέτοντας στην στήλη DEPT_NAME την τιμή DEVELOPMENT:

```
alter table DEPT alter DEPT_NAME set default 'DEVELOPMENT';  
insert into EMP (EMPNO, ENAME, JOBNO, DEPTNO, COMM) values (50,  
'CODD', 100, 50, NULL); -- Στο παράδειγμα της εργασίας υπάρχει ήδη  
υπάλληλος με κωδικό 10 οπότε το αλλάζουμε σε 50.  
insert into DEPT (DEPTNO) values (10), (20);  
insert into DEPT values (30, 'SALES');  
select * from EMP;  
select * from DEPT;
```

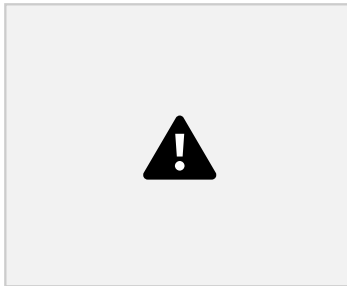
Εισάγαμε νέες τιμές στους πίνακες για να δείξουμε την λειτουργία του περιορισμού DEFAULT.

Νέα περιεχόμενα των πινάκων EMP, DEPT:

EMP:



DEPT:



Βλέπουμε πώς έγινε η εισαγωγή στον πίνακα DEPT χωρίς να οριστεί DEPT_NAME και ορίστηκε αυτόματα ως DEVELOPMENT.

17. Εμφάνιση της λίστας περιορισμών της ΒΔ personnel:

```
select * from INFORMATION_SCHEMA.TABLE_CONSTRAINTS where  
CONSTRAINT_SCHEMA = 'personnel';
```



18. Δημιουργία πίνακα PROJECT με στήλες P_ID και P_NAME.

```
create table PROJECT(  
    P_ID int not null auto_increment,  
    P_NAME varchar(255),  
    primary key(P_ID)  
) engine=InnoDB default charset=utf8;  
describe PROJECT;
```

Εμφάνιση της δομής το πίνακα PROJECT:



Βλέπουμε ότι οι στήλες του πίνακα έχουν τις ιδιότητες που ορίσαμε για αυτές.

19. Καταχωρούμε νέες εγγραφές ώστε να φανει η λειτουργία των περιορισμών:

```
insert into PROJECT (P_NAME) values  
    ('FRONTEND'),  
    ('BACKEND'),  
    ('DATABASE');  
select * from PROJECT;
```

Εμφάνιση των περιεχομένων του πίνακα PROJECT:

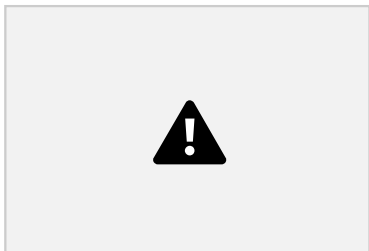


Βλέπουμε ότι χωρίς την εισαγωγή του P_ID εισάγεται μόνο του και αυξάνεται με κάθε νέα εγγραφή.

20. Επαναλαμβάνουμε το παραπάνω βήμα ορίζοντας ως αρχική τιμή του AUTO_INCREMENT το 200:

```
alter table PROJECT auto_increment = 200;  
insert into PROJECT (P_NAME) values  
    ('SALES PROJECTION'),  
    ('ANNUAL ANALYTICS');  
select * from PROJECT;
```

Εμφάνιση του νέου περιεχομένου του πίνακα PROJECT:



Βλέπουμε ότι το P_ID των νέων εγγραφών ξεκίνησαν από το 200 και αυξάνονται κατά 1 από εκεί.