

ΙΩΑΝΝΗΣ ΛΟΥΔΑΡΟΣ - ΧΡΙΣΤΙΝΑ ΚΡΑΤΗΜΕΝΟΥ

STAFF EVALUATION PROJECT

# ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ



# Εισαγωγή

## ΓΕΝΙΚΕΣ ΠΛΗΡΟΦΟΡΙΕΣ

**Η** παρακάτω αναφορά περιγράφει τον τρόπο λειτουργίας και τις υλοποιητικές τεχνικές που χρησιμοποιήθηκαν για την ολοκλήρωση του Project του μαθήματος “Βάσεις Δεδομένων”. Ξεκινώντας την αναφορά παραθέτουμε τα στοιχεία μας καθώς και συμπληρωματικό υλικό στο οποίο μπορείτε να ανατρέξετε για μια πιο ολοκληρωμένη εμπειρία. Μπορείτε πάντα να έχετε πρόσβαση στην τελευταία έκδοση του project σκανάροντας τον κωδικό QR στο εξώφυλλο ή κάνοντας κλικ [εδώ](#)



[Η Βάση Αναλυτικά](#)

Χρησιμοποιώντας της πλατφόρμα dbdocs έχουμε δημιουργήσει μια αναλυτική παρουσίαση του σχήματος της βάσης, με εκτενείς πληροφορίες για τα ορίσματα και τις συσχετίσεις μεταξύ των σχέσεων



[Διαδραστικό  
Σχεσιακό](#)

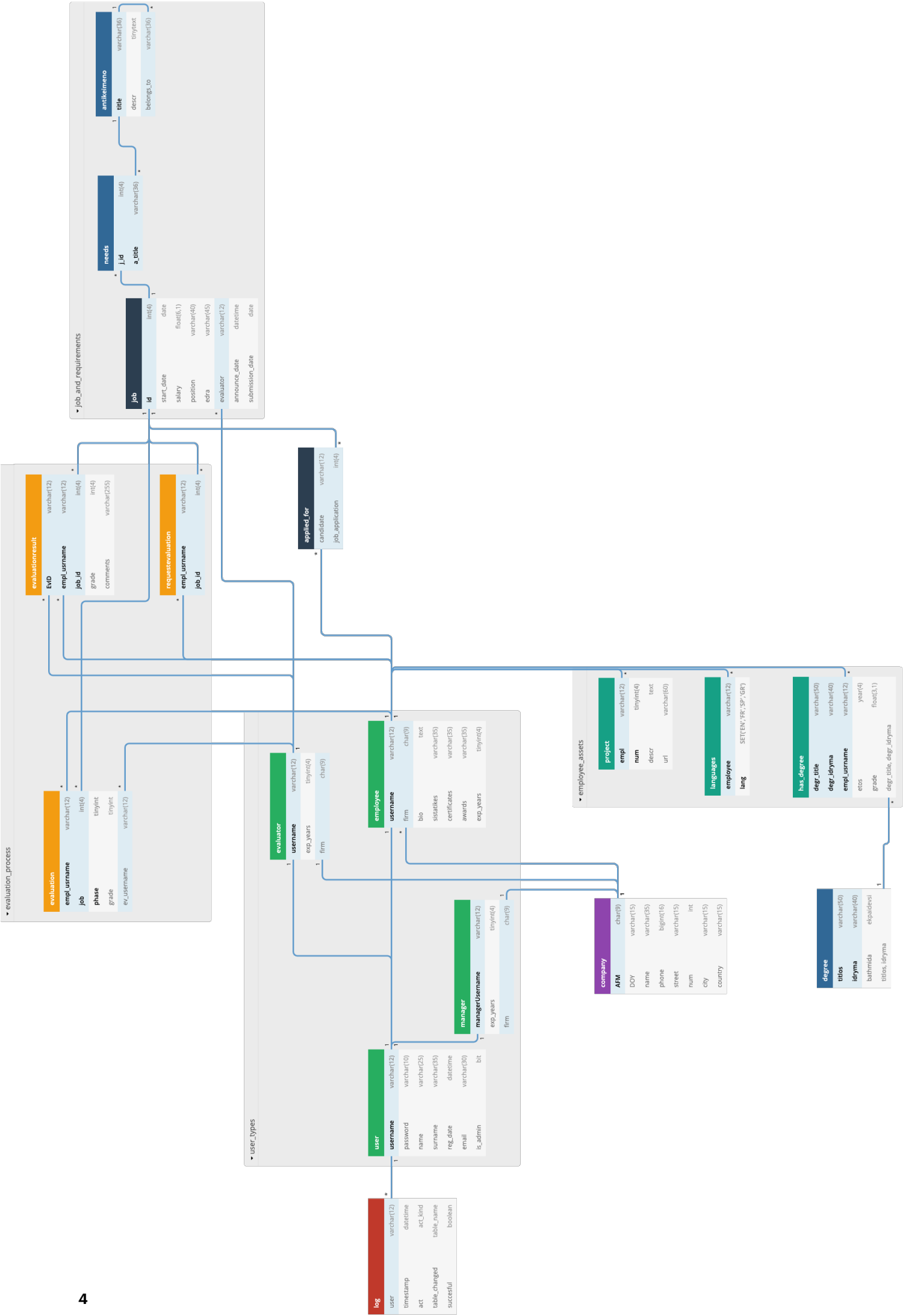
Αξιοποιώντας την δύναμη της γλώσσας dbml σας παραδίδουμε μέσω του εργαλείου dbdiagram ένα διαδραστικό σχεσιακή της βάσης μας για να κατανοήσετε καλύτερα την δομή της.

**Ιωάννης Λουδάρος**  
1067400  
**E-mail**  
**Web page**

**Χριστίνα Κρατημένου**  
1067495  
**E-mail**

# ΠΕΡΙΕΧΟΜΕΝΑ

<b>1. Σχεσιακό και παραδοχές</b>	.....4
i. Σχεσιακό	.....4
ii. Παραδοχές	.....5
<b>2. Δεδομένα εκκίνησης</b>	.....6
i. Create	.....6
ii. Insert	.....8
<b>3. Stored Procedures</b>	.....11
<b>4. Triggers</b>	.....13
<b>5. Διεπαφές</b>	.....15
i. Manager	.....15
ii. Evaluator	.....16
iii. Employee	.....17
iv. Administrator	.....18



## ΣΧΕΣΙΑΚΟ ΔΙΑΓΡΑΜΜΑ ΒΑΣΗΣ

Στην σελίδα 4 παραθέτουμε το σχεσιακό διάγραμμα της βάσης μας. Για την ευκολότερη κατανόηση του αλλά και για να έχετε πρόσβαση στις πληροφορίες πληθικότητας των αναφορών, μπορείτε να ανατρέξετε στον σύνδεσμο “[Διαδραστικό Σχεσιακό](#)”. Ο μηχανισμός αξιολόγησης διαφαίνεται στο πάνω μέρος (οι σχέσεις του σημειώνονται με πορτοκαλί).

## ΠΑΡΑΔΟΧΕΣ ΣΤΟΝ ΣΧΕΔΙΑΣΜΟ ΤΗΣ ΒΑΣΗΣ

Οι παραδοχές που έχουν παρθεί κατά τον σχεδιασμό της Βάσης σημειώνονται παρακάτω:

- Όταν ανακοινώνεται ένα job, ένας **employee** έχει την δυνατότητα να κάνει αίτηση αξιολόγησης. Η αίτηση παραμένει στον πίνακα **requestevaluation** μέχρι να ολοκληρωθούν όλα τα βήματα της αξιολόγησης του και να συμπληρωθεί το τελικό αποτέλεσμα στο **evaluationresult**.
- Θεωρήσαμε ότι ο ίδιος **employee** μπορεί να κάνει πολλές διαφορετικές αιτήσεις και ένας **evaluator** μπορεί να αξιολογεί πολλούς **employee** για την ίδια δουλειά. Για αυτό για να περιγράψουμε μοναδικά μια αξιολόγηση χρειαζόμαστε 3 στοιχεία: τα username του **employee** και του **evaluator**, και το id της δουλειάς.

# ΔΕΔΟΜΕΝΑ ΕΚΚΙΝΗΣΗΣ

Για τις ανάγκες της αξιολόγησης παραθέτουμε τον κώδικα SQL για την δημιουργία του σχήματος της βάσης (το σύνολο των create) καθώς και τα δεδομένα που “προϋπάρχουν” για τις ανάγκες της παρουσίασης.

```
CREATE TABLE `user` (  
  `username` varchar(12) PRIMARY KEY,  
  `password` varchar(10),  
  `name` varchar(25),  
  `surname` varchar(35),  
  `reg_date` datetime,  
  `email` varchar(30),  
  `is_admin` bit  
);
```

```
CREATE TABLE `employee` (  
  `username` varchar(12) PRIMARY KEY,  
  `firm` char(9),  
  `bio` text,  
  `sistatikes` varchar(35),  
  `certificates` varchar(35),  
  `awards` varchar(35),  
  `exp_years` tinyint(4)  
);
```

```
CREATE TABLE `manager` (  
  `managerUsername` varchar(12)  
PRIMARY KEY,  
  `exp_years` tinyint(4) NOT NULL,  
  `firm` char(9)  
);
```

```
CREATE TABLE `evaluator` (  
  `username` varchar(12) PRIMARY KEY,  
  `exp_years` tinyint(4),  
  `firm` char(9)  
);
```

```
CREATE TABLE `company` (  
  `AFM` char(9) PRIMARY KEY,  
  `DOY` varchar(15),  
  `name` varchar(35),  
  `phone` bigint(16),  
  `street` varchar(15),  
  `num` int,  
  `city` varchar(15),  
  `country` varchar(15)  
);
```

```
CREATE TABLE `antikeimeno` (  
  `title` varchar(36) PRIMARY KEY,  
  `descr` tinytext,  
  `belongs_to` varchar(36)  
);
```

```
CREATE TABLE `needs` (  
  `j_id` int(4) AUTO_INCREMENT,  
  `a_title` varchar(36) UNIQUE,  
  PRIMARY KEY (`j_id`, `a_title`)  
);
```

```
CREATE TABLE `project` (  
  `empl` varchar(12),  
  `num` tinyint(4) UNIQUE  
AUTO_INCREMENT,  
  `descr` text,  
  `url` varchar(60),  
  PRIMARY KEY (`empl`, `num`)  
);
```

```
CREATE TABLE `languages` (  
  `employee` varchar(12),  
  `lang` SET('EN','FR','SP','GR'),  
  PRIMARY KEY (`employee`, `lang`)  
);
```

```
CREATE TABLE `requestevaluation` (  
  `empl_username` varchar(12),  
  `job_id` int(4),  
  PRIMARY KEY (`empl_username`,  
  `job_id`)  
);
```

```
CREATE TABLE `job` (  
  `id` int(4) PRIMARY KEY  
AUTO_INCREMENT,  
  `start_date` date,  
  `salary` float(6,1),  
  `position` varchar(40),  
  `edra` varchar(45),  
  `evaluator` varchar(12),  
  `announce_date` datetime DEFAULT  
NOW(),  
  `submission_date` date  
);
```

```
CREATE TABLE `degree` (  
  `titlos` varchar(50),  
  `idryma` varchar(40),  
  `bathmida` ENUM ('LYKEIO', 'UNIV',  
  'MASTER', 'PHD'),  
  PRIMARY KEY (`titlos`, `idryma`)  
);
```

```
CREATE TABLE `has_degree` (  
  `user` varchar(12),  
  `degree` varchar(90),  
  PRIMARY KEY (`user`, `degree`)  
);
```

```
`degr_title` varchar(50),  
`degr_idryma` varchar(40),  
`empl_username` varchar(12),  
`etos` year(4),  
`grade` float(3,1),  
PRIMARY KEY (`degr_title`,  
`degr_idryma`, `empl_username`)  
);
```

```
CREATE TABLE `applied_for` (  
  `candidate` varchar(12),  
  `job_application` int(4)  
);
```

```
CREATE TABLE `evaluation` (  
  `empl_username` varchar(12),  
  `job` int(4),  
  `phase` tinyint,  
  `grade` tinyint,  
  `ev_username` varchar(12),  
  PRIMARY KEY (`empl_username`, `job`,  
  `phase`)  
);
```

```
CREATE TABLE `evaluationresult` (  
  `EVID` varchar(12),  
  `empl_username` varchar(12),  
  `job_id` int(4),  
  `grade` int(4),  
  `comments` varchar(255),  
  PRIMARY KEY (`EVID`,  
  `empl_username`)  
);
```

```
CREATE TABLE `log` (  
  `user` varchar(12),  
  `timestamp` datetime DEFAULT  
NOW(),  
  `act` ENUM ('insert', 'update', 'delete'),  
  `table_changed` ENUM ('job',  
  'employee', 'requestevaluation'),  
  `succesful` boolean  
);
```

```
ALTER TABLE `employee` ADD  
FOREIGN KEY (`username`)  
REFERENCES `user` (`username`);  
ALTER TABLE `employee` ADD  
FOREIGN KEY (`username`)  
REFERENCES `user` (`username`);
```

Για τον ευκολότερο χειρισμό των συσχετίσεων και την αποφυγή σφαλμάτων που θα μπορούσαν να προκύψουν από την λάθος σειρά δημιουργίας σχέσεων, αποφασίσαμε ότι είναι καλύτερο να τις έχουμε συγκεντρωμένες.

```
ALTER TABLE `employee` ADD  
FOREIGN KEY (`firm`) REFERENCES  
`company` (`AFM`);
```

```
ALTER TABLE `manager` ADD FOREIGN  
KEY (`managerUsername`) REFERENCES  
`user` (`username`);
```

```
ALTER TABLE `manager` ADD FOREIGN  
KEY (`firm`) REFERENCES `company`  
(`AFM`);
```

```
ALTER TABLE `evaluator` ADD FOREIGN  
KEY (`username`) REFERENCES `user`  
(`username`);
```

```
ALTER TABLE `evaluator` ADD FOREIGN  
KEY (`firm`) REFERENCES `company`  
(`AFM`);
```

```
ALTER TABLE `antikeimeno` ADD  
FOREIGN KEY (`belongs_to`) REFERENCES  
`antikeimeno` (`title`);
```

```
ALTER TABLE `needs` ADD FOREIGN  
KEY (`j_id`) REFERENCES `job` (`id`);
```

```
ALTER TABLE `needs` ADD FOREIGN  
KEY (`a_title`) REFERENCES  
`antikeimeno` (`title`);
```

```
ALTER TABLE `project` ADD FOREIGN  
KEY (`empl`) REFERENCES `employee`  
(`username`);
```

```
ALTER TABLE `languages` ADD  
FOREIGN KEY (`employee`) REFERENCES  
`employee` (`username`);
```

```
ALTER TABLE `requestevaluation` ADD  
FOREIGN KEY (`empl_username`) REFERENCES  
`employee` (`username`);
```

```
ALTER TABLE `requestevaluation` ADD  
FOREIGN KEY (`job_id`) REFERENCES  
`job` (`id`);
```

```
ALTER TABLE `job` ADD FOREIGN KEY  
(`evaluator`) REFERENCES `evaluator`  
(`username`);
```

```
ALTER TABLE `has_degree` ADD  
FOREIGN KEY (`empl_username`) REFERENCES  
`employee` (`username`);
```

```
ALTER TABLE `has_degree` ADD  
FOREIGN KEY (`degr_title`,  
`degr_idryma`) REFERENCES `degree`  
(`titlos`, `idryma`);
```

```
ALTER TABLE `applied_for` ADD  
FOREIGN KEY (`candidate`) REFERENCES  
`employee` (`username`);
```

```
ALTER TABLE `applied_for` ADD  
FOREIGN KEY (`job_application`) REFERENCES  
`job` (`id`);
```

```
ALTER TABLE `evaluation` ADD  
FOREIGN KEY (`empl_username`) REFERENCES  
`employee` (`username`);
```

```
ALTER TABLE `evaluation` ADD  
FOREIGN KEY (`job`) REFERENCES  
`job` (`id`);
```

```
ALTER TABLE `evaluation` ADD  
FOREIGN KEY (`ev_username`) REFERENCES  
`evaluator` (`username`);
```

```
ALTER TABLE `evaluationresult` ADD  
FOREIGN KEY (`EvID`) REFERENCES  
`evaluator` (`username`);
```

```
ALTER TABLE `evaluationresult` ADD  
FOREIGN KEY (`empl_username`) REFERENCES  
`employee` (`username`);
```

```
ALTER TABLE `evaluationresult` ADD  
FOREIGN KEY (`job_id`) REFERENCES  
`job` (`id`);
```

```
ALTER TABLE `log` ADD FOREIGN KEY  
(`user`) REFERENCES `user`  
(`username`);
```



Εισάγουμε τα αρχικά δεδομένα της βάσης με τις κατάλληλες εντολές insert. Φροντίζουμε να υπάρχει πληθώρα δεδομένων για να παρατηρήσουμε διάφορες περιπτώσεις που μπορεί να προκύψουν.

```
INSERT INTO user VALUES
('ilouaros','hnBHWK','Ioannis','Loudaros','
2015-11-27
9:14:16','ilouaros@upnet.gr',1);
INSERT INTO user VALUES
('xkratim','Jh87Xw','Xristina','Kratimenou','
2014-05-14
8:10:45','xkratim@upnet.gr',1);
INSERT INTO user VALUES
('sjames','WS6dZr','Sheila','James','2014-0
1-11 10:17:21','sjames@tech.com',0);
INSERT INTO user VALUES
('agray','BXtnBM','Andrew','Gray','2019-06
-12 9:36:11','agray@tech.com',0);
INSERT INTO user VALUES
('aanson','Sa6xfu','Angie','Anson','2018-07
17 11:05:07','aanson@logic.com',0);
INSERT INTO user VALUES
('agreenl','5cgQFw','Annette','Greenlove','
2017-12-17
10:30:17','agreenl@prev.com',0);
INSERT INTO user VALUES
('wgregg','jwCbKV','William','Gregg','2018
-03-10 08:17:14','wgregg@prev.com',0);
INSERT INTO user VALUES
('jmarsh','mYqCjU','James','Marsh','2019-0
4-10 09:18:48','jmarsh@prev.com',0);
INSERT INTO user VALUES
('amakint','MZ7cQS','Andrew','Makintosh','
2020-04-25
10:29:01','amakint@tech.com',0);
INSERT INTO user VALUES
('emarch','amPBkr','Erica','March','2019-02
-14 09:26:18','emarch@logic.com',0);
INSERT INTO user VALUES
('jstewart','4F88Bh','Jennifer','Stewart','201
8-05-02
11:12:18','jstewart@mod.com',0);
INSERT INTO user VALUES
('anelson','3XtQqc','Amy','Nelson','2016-0
8-19 10:41:09','anelson@mod.gr',0);
INSERT INTO user VALUES
('hkent','5vbPDt','Heather','Kent','2020-10-
28 14:10:05','hkent@mod.gr',0);
INSERT INTO user VALUES
('sknowles','4CZfwQ','Seance','Knowles','2
015-02-10
09:20:48','sknowles@logic.com',0);
INSERT INTO user VALUES
('cstinton','xKXefH','Christa','Stinton','202
0-09-28 08:39:21','cstinton@mod.gr',0);
```

```
INSERT INTO user VALUES
('abrown','dbnzzC','Alex','Brown','2019-08-
16 09:14:22','abrown@logic.com',0);
INSERT INTO user VALUES
('ddavis','wbNqyS','Devon','Davis','2018-11-
17 10:18:20','ddavis@logic.com',0);
INSERT INTO user VALUES
('ewhite','nRMUD3','Evan','White','2019-09
-09 09:14:04','ewhite@logic.com',0);
INSERT INTO user VALUES
('hthomas','5RCdFf','Hunter','Thomas','20
20-11-10
08:05:52','hthomas@logic.com',0);
INSERT INTO user VALUES
('kander','m7n6EH','Kai','Anderson','2018-1
0-17 08:35:51','kander@logic.com',0);

INSERT INTO company VALUES
('240277197','DOY G
Patrwn','Upnet','2610567846','Korinthou','
350','Patras','Greece');
INSERT INTO company VALUES
('114422384','DOY
Rethymnoy','Prevail','2831025645','Kount
ouriwth','59','Rethymno','Greece');
INSERT INTO company VALUES
('926898228','DOY Iwnias
Thes','Tech','2310152645','Bas
Gewrgiou','95','Thessaloniki','Greece');
INSERT INTO company VALUES
('160933162','DOY A
Athinwn','Modular','2105448112','L Basil
Sofias','244','Athens','Greece');
INSERT INTO company VALUES
('506365115','DOY
Ioanninwn','Logic','2651056412','Dwdwnh
s','161','Ioannina','Greece');

INSERT INTO manager VALUES
('hkent','10','160933162');
INSERT INTO manager VALUES
('hthomas','7','506365115');
INSERT INTO manager VALUES
('amakint','11','926898228');
INSERT INTO manager VALUES
('ilouaros','8','240277197');
INSERT INTO manager VALUES
('jmarsh','9','114422384');
```

```
INSERT INTO evaluator VALUES
('sknowles','5','506365115');
INSERT INTO evaluator VALUES
('kander','9','506365115');
INSERT INTO evaluator VALUES
('emarch','6','506365115');
INSERT INTO evaluator VALUES
('aanson','10','506365115');

INSERT INTO employee VALUES
('xkratim','240277197','bio','recommenda
t','certif','awar','3');
INSERT INTO employee VALUES
('sjames','926898228','bio','recommenda
t','certif','awar','6');
INSERT INTO employee VALUES
('agray','926898228','bio','recommenda
t','certif','awar','5');
INSERT INTO employee VALUES
('agreenl','114422384','bio','recommenda
t','certif','awar','10');
INSERT INTO employee VALUES
('wgregg','114422384','bio','recommenda
t','certif','awar','6');
INSERT INTO employee VALUES
('jstewart','160933162','bio','recommenda
t','certif','awar','8');
INSERT INTO employee VALUES
('anelson','160933162','bio','recommenda
t','certif','awar','7');
INSERT INTO employee VALUES
('cstinton','160933162','bio','recommenda
t','certif','awar','7');
INSERT INTO employee VALUES
('abrown','506365115','bio','recommenda
t','certif','awar','11');
INSERT INTO employee VALUES
('ddavis','506365115','bio','recommenda
t','certif','awar','9');
INSERT INTO employee VALUES
('ewhite','506365115','bio','recommenda
t','certif','awar','5');

INSERT INTO job VALUES
(null,'2021-03-05','2572.7','PR
Supervisor','Patras,
Greece','sknowles','2020-12-20
08:15:37','2020-12-30');
```



```

INSERT INTO job VALUES
(null,'2021-01-25','764.4','HR
Coordinator','Thessaloniki,
Greece','kander','2020-12-04
10:35:21','2020-12-30');
INSERT INTO job VALUES
(null,'2021-03-10','860.8','FR
Analyst','Thessaloniki,
Greece','kander','2020-12-04
10:35:54','2020-12-30');
INSERT INTO job VALUES
(null,'2021-02-01','4527.5','Manager','Reth
ymno, Greece','emarch','2020-12-12
9:10:25','2020-12-30');
INSERT INTO job VALUES
(null,'2021-02-10','952.3','PR
Supervisor','Rethymno,
Greece','emarch','2020-12-12
9:10:39','2020-12-30');
INSERT INTO job VALUES
(null,'2021-01-20','1200.2','CR
Embassador','Rethymno,
Greece','emarch','2020-12-12
9:11:19','2020-12-30');
INSERT INTO job VALUES
(null,'2021-02-01','956.6','UX
Designer','Athens,
Greece','aanson','2020-12-21
08:30:15','2020-12-30');
INSERT INTO job VALUES
(null,'2021-02-10','1526.1','PR
Supervisor','Athens,
Greece','aanson','2020-12-21
08:30:52','2020-12-30');
INSERT INTO job VALUES
(null,'2021-01-31','856.2','FR
Analyst','Ioannina,
Greece','sknowles','2020-12-20
08:15:45','2020-12-30');
INSERT INTO job VALUES
(null,'2021-02-07','3567.9','Manager','Ioan
nina, Greece','sknowles','2020-12-20
08:16:21','2020-12-30');

```

```

INSERT INTO antikeimeno VALUES
('Public Relations','-','null');
INSERT INTO antikeimeno VALUES
('Management','-','null');
INSERT INTO antikeimeno VALUES
('Design','-','null');
INSERT INTO antikeimeno VALUES
('Economics','-','Management');
INSERT INTO antikeimeno VALUES
('Social Media','In charge of the
company`s social media
presence','Public Relations');
INSERT INTO antikeimeno VALUES
('Soft Skills','In charge of group`s

```

```

dynamics and conflict
resolution','Management');
INSERT INTO antikeimeno VALUES
('Logistics','In charge of the company`s
economic management and
analysis','Management');
INSERT INTO antikeimeno VALUES
('Graphic Design','In charge of the
company`s logo design and website
appearance','Design');
INSERT INTO antikeimeno VALUES
('Photoshop','In charge of creating
posters, business cards etc.','Design');
INSERT INTO antikeimeno VALUES
('Human Resources','-','Management');

```

```

INSERT INTO degree VALUES
('Apolythrio Lykeio','17o Geniko Lykeio
Peristerio','LYKEIO');
INSERT INTO degree VALUES
('Apolythrio Lykeio','Arsakeio Lykeio
Thessalonikis','LYKEIO');
INSERT INTO degree VALUES
('Apolythrio Lykeio','5o Geniko Lykeio
Patrws','LYKEIO');
INSERT INTO degree VALUES ('Ptyxio
Organwshs kai Dioikhshs
Epixeirhsewn','Papa, Tmhma Dioikhshs
Epixeirhsewn','UNIV');
INSERT INTO degree VALUES ('Ptyxio
Oikonomikwn Episthmwn','APTh,
Oikonomikwn kai Politikwn
Spoudwn','UNIV');
INSERT INTO degree VALUES ('Ptyxio
Statistikhs kai Asfalistikhs','PaPei,
Statistikhs kai Asfalistikhs','UNIV');
INSERT INTO degree VALUES ('Master`s
Degree on Graphic Design','Deree, The
American College of Greece','MASTER');
INSERT INTO degree VALUES ('Master`s
Degree on Logistic Management','APTh,
Logistics k Dioikhsh A
Efodiasmou','MASTER');
INSERT INTO degree VALUES ('Master`s
Degree on New Technologies and
Marketing','UNIWA, Grafistikhs kai Opt
Epikoinwnias','MASTER');
INSERT INTO degree VALUES ('PhD in
Management, Finance and
Economics','Deree','PHD');
INSERT INTO degree VALUES
('Didaktoriko stis Oikonomikes
Episthmes','PaPe, Tmhma Oikonomikwn
Episthmwn','PHD');
INSERT INTO degree VALUES ('PhD in
Marketing Science', 'PaPei, Tmhma
Organwshs kai Dioikhshs','PHD');

```

```

INSERT INTO has_degree VALUES
('Ptyxio Organwshs kai Dioikhshs
Epixeirhsewn','Papa, Tmhma Dioikhshs
Epixeirhsewn','agree','2014','9.3');
INSERT INTO has_degree VALUES
('Apolythrio Lykeio','5o Geniko Lykeio
Patrws','agree','2009','18.6');
INSERT INTO has_degree VALUES
('Ptyxio Statistikhs kai
Asfalistikhs','PaPei, Statistikhs kai
Asfalistikhs','agray','2015','8.7');
INSERT INTO has_degree VALUES
('Master`s Degree on Logistic
Management','APTh, Logistics k
Dioikhsh A
Efodiasmou','agray','2017','9.5');
INSERT INTO has_degree VALUES
('Ptyxio Organwshs kai Dioikhshs
Epixeirhsewn','Papa, Tmhma Dioikhshs
Epixeirhsewn','sjames','2013','9.4');
INSERT INTO has_degree VALUES
('Apolythrio Lykeio','Arsakeio Lykeio
Thessalonikis','xkratim','2014','19.6');
INSERT INTO has_degree VALUES
('Master`s Degree on Graphic
Design','Deree, The American College
of Greece','wgregg','2017','9.8');
INSERT INTO has_degree VALUES
('Ptyxio Oikonomikwn
Episthmwn','APTh, Oikonomikwn kai
Politikwn
Spoudwn','jstewart','2008','10.0');
INSERT INTO has_degree VALUES
('Master`s Degree on Logistic
Management','APTh, Logistics k
Dioikhsh A
Efodiasmou','jstewart','2011','9.9');
INSERT INTO has_degree VALUES ('PhD
in Management, Finance and
Economics','Deree','jstewart','2014','9.8');
INSERT INTO has_degree VALUES
('Master`s Degree on Graphic
Design','Deree, The American College
of Greece','anelson','2015','10.0');
INSERT INTO has_degree VALUES
('Apolythrio Lykeio','Arsakeio Lykeio
Thessalonikis','cstinton','2019','20.0');
INSERT INTO has_degree VALUES
('Ptyxio Organwshs kai Dioikhshs
Epixeirhsewn','Papa, Tmhma Dioikhshs
Epixeirhsewn','abrown','2016','8.7');
INSERT INTO has_degree VALUES
('Master`s Degree on New
Technologies and Marketing','UNIWA,
Grafistikhs kai Opt
Epikoinwnias','abrown','2018','9.4');

```

```

INSERT INTO has_degree VALUES
('Ptyxio Statistikhs kai
Asfalistikhs','PaPei, Statistikhs kai
Asfalistikhs','ddavis','2011','9.8');
INSERT INTO has_degree VALUES
('PhD in Marketing Science', 'PaPei,
Tmhma Organwshs kai
Dioikhshs','ddavis','2016','9.1');
INSERT INTO has_degree VALUES
('Didaktoriko stis Oikonomikes
Episthmes','PaPe, Tmhma
Oikonomikwn
Episthmwn','ewhite','2016','10.0');

```

```

INSERT INTO languages VALUES
('xkratim','GR,EN');
INSERT INTO languages VALUES
('sjames','EN,FR');
INSERT INTO languages VALUES
('agray','EN,SP,FR');
INSERT INTO languages VALUES
('agreenl','EN,GR,FR');
INSERT INTO languages VALUES
('wgregg','EN');
INSERT INTO languages VALUES
('jstewart','EN,FR');
INSERT INTO languages VALUES
('anelson','EN,GR');
INSERT INTO languages VALUES
('cstinton','EN,FR');
INSERT INTO languages VALUES
('abrown','EN');
INSERT INTO languages VALUES
('ddavis','EN,FR,SP');
INSERT INTO languages VALUES
('ewhite','EN,SP');

```

```

INSERT INTO project VALUES
('xkratim',null,'Project
Description:','http://www.emp/xk/
projnum.com');
INSERT INTO project VALUES
('agray',null,'Project
Description:','http://www.emp/ag/
projnum.com');
INSERT INTO project VALUES
('agreenl',null,'Project
Description:','http://www.emp/agr/
projnum.com');
INSERT INTO project VALUES
('wgregg',null,'Project
Description:','http://www.emp/wg/
projnum.com');

```

```

INSERT INTO project VALUES
('anelson',null,'Project
Description:','http://www.emp/an/
projnum.com');
INSERT INTO project VALUES
('ddavis',null,'Project
Description:','http://www.emp/dd/
projnum.com');
INSERT INTO project VALUES
('ewhite',null,'Project
Description:','http://www.emp/ew/
projnum.com');

```

```

INSERT INTO needs VALUES
('0001','Social Media');
INSERT INTO needs VALUES
('0002','Soft Skills');
INSERT INTO needs VALUES
('0003','Economics');
INSERT INTO needs VALUES
('0004','Management');
INSERT INTO needs VALUES
('0005','Photoshop');
INSERT INTO needs VALUES
('0006','Design');
INSERT INTO needs VALUES
('0007','Graphic Design');
INSERT INTO needs VALUES
('0008','Public Relations');
INSERT INTO needs VALUES
('0009','Logistics');
INSERT INTO needs VALUES
('0010','Human Resources');

```

```

INSERT INTO applied_for VALUES
('sjames','0010');
INSERT INTO applied_for VALUES
('cstinton','0002');
INSERT INTO applied_for VALUES
('wgregg','0007');
INSERT INTO applied_for VALUES
('agray','0003');

```

```

INSERT INTO requestevaluation
VALUES ('ewhite','0009');
INSERT INTO requestevaluation
VALUES ('ddavis','0005');
INSERT INTO requestevaluation
VALUES ('agreenl','0006');
INSERT INTO requestevaluation
VALUES ('anelson','0007');

```

```

INSERT INTO evaluation VALUES
('ewhite','9','1','3','sknowles');
INSERT INTO evaluation VALUES
('ewhite','9','2','3','sknowles');
INSERT INTO evaluation VALUES
('ewhite','9','3','3','sknowles');
INSERT INTO evaluation VALUES
('ddavis','9','1','4','emarch');
INSERT INTO evaluation VALUES
('ddavis','9','2','3','emarch');
INSERT INTO evaluation VALUES
('agreenl','6','1','3','emarch');
INSERT INTO evaluation VALUES
('agreenl','6','2','2','emarch');
INSERT INTO evaluation VALUES
('agreenl','6','3','3','emarch');
INSERT INTO evaluation VALUES
('anelson','7','1','3','aanson');
INSERT INTO evaluation VALUES
('anelson','7','2','3','aanson');
INSERT INTO evaluation VALUES
('anelson','7','3','4','aanson');

```

```

INSERT INTO evaluationresult VALUES
('sknowles','ewhite','0009','9','comments
commentscommentscomments');
INSERT INTO evaluationresult VALUES
('emarch','agreenl','0006','8','commentssc
ommentscommentscomments');
INSERT INTO evaluationresult VALUES
('aanson','anelson','0007','10','comments
commentscommentscomments');

```

# STORED PROCEDURES

1. Το πρώτο Stored Procedure χρησιμοποιεί τον πίνακα user για να πάρει τα στοιχεία του **employee**, έπειτα συλλέγει στοιχεία για αυτόν από τους πίνακες **requestevaluation** (για να δει αν υπάρχουν ενεργές αιτήσεις), **evaluationresult** (για να δει τα αποτελέσματα των αξιολογήσεων του), **evaluation** (για να πάρει τις αξιολογήσεις που δεν έχουν ολοκληρωθεί ακόμη) **user** (για να συλλέξει το ονοματεπώνυμο του αξιολογητή). Την πληροφορία που χρειαζόμαστε για να βρούμε το ονοματεπώνυμο του αξιολογητή την εντοπίζουμε στο πεδίο EvID του **evaluationresult**. Παραθέτουμε τον κώδικα SQL παρακάτω. Υπάρχει πρόβλεψη για την αντιμετώπιση λανθασμένης εισαγωγής.

```
DELIMITER $
```

```
CREATE PROCEDURE seeevalresults (em_name  
VARCHAR(25),em_surname VARCHAR(35))  
BEGIN  
DECLARE em_username VARCHAR(12);  
DECLARE ev_name VARCHAR(25);  
DECLARE ev_username VARCHAR(25);
```

```
SELECT username INTO emp_username FROM user WHERE user.name  
=em_name AND user.username=em_surname;
```

```
IF(em_username != NULL) THEN
```

```
    IF( not exists(SELECT * FROM requestevaluation WHERE  
em_username=empl_username)) THEN
```

```
        SELECT 'Αι μωρ@ αφού δεν έχεις κάνει καν αίτηση  
ακόμη.' as '';
```

```
    ELSE
```

```
        SELECT 'Οι αιτήσεις σας κυρι@ τέτοι@ μου:' as '';
```

```
        SELECT * FROM requestevaluation WHERE  
empl_username=em_username;
```

```
        SELECT 'Τα αποτελέσματα σας κυρι@ τέτοι@ μου:' as '';
```

```
        SELECT evaluationresult.empl_username,  
evaluationresult.job_id, evaluationresult.grade,  
evaluationresult.comments, user.name, user.surname  
FROM evaluationresult  
WHERE empl_username=em_username  
INNER JOIN user on EvID = username;  
  
    END IF;
```

```
    IF( exists(select job,empl_username from evaluation  
where phase = 1 and job,empl_username not in  
(select job from evaluation where phase=3) THEN
```

```
        SELECT 'Αυτές οι αξιολογήσεις δεν έχουν  
ολοκληρωθεί ακόμη:' as '';
```

```
        SELECT evaluation.empl_username,  
evaluation.job, evaluation.grade, evaluation.phase, user.name,  
user.surname
```

```
FROM evaluationresult  
WHERE empl_username=em_username AND phase  
!=3 AND job,empl_username not in  
(select job from evaluation where phase=3)  
INNER JOIN user on EvID = username;
```

```
    END IF;
```

```
END IF;
```

```
END$  
DELIMITER ;
```

2. Γίνεται έλεγχος για το αν υπάρχουν οι τιμές των βαθμολογιών για κάθε φάση στον πίνακα **evaluation** και με την προϋπόθεση πως έχουν καταχωρηθεί (Η SELECT EXISTS επιστρέφει τιμή 1) , μεταφέρονται σε μεταβλητές και υπολογίζεται το άθροισμα τους. Μετά τον υπολογισμό, καταχωρείται στον πίνακα **evaluationresult**, ενώ σε περίπτωση που δεν βρεθεί βαθμολογία σε όλες τις φάσεις εμφανίζεται ανάλογο μήνυμα.

DELIMITER \$

```
CREATE PROCEDURE make_final (id_job INT(4), evaluatorin
VARCHAR(12), employeein varchar(12));
BEGIN

if(exists(select * from requestevaluation where empl_username =
employeein and job_id = id_job)) then
SET @g1 = SELECT EXISTS(SELECT grade FROM evaluation WHERE
phase=1 and empl_username = employeein and ev_username =
evaluatorin);
SET @g2 = SELECT EXISTS(SELECT grade FROM evaluation WHERE
phase=2 and empl_username = employeein and ev_username =
evaluatorin);
SET @g3 = SELECT EXISTS(SELECT grade FROM evaluation WHERE
phase=3 and empl_username = employeein and ev_username =
evaluatorin);
```

```
IF(@g1=1) AND (@g2=1) AND (g3=1); THEN
```

```
SET @fg1 = SELECT grade FROM evaluation WHERE phase=1 and
empl_username = employeein and ev_username = evaluatorin;
SET @fg2 = SELECT grade FROM evaluation WHERE phase=2 and
empl_username = employeein and ev_username = evaluatorin;
SET @fg3 = SELECT grade FROM evaluation WHERE phase=3
and empl_username = employeein and ev_username = evaluatorin;
```

```
INSERT INTO evaluationresult values
(evaluatorin,employeein,id_job,@fg1+@fg2+@fg3)
ELSE
SELECT 'Your evaluation grading is still in progress.' as ";
END IF;
```

```
else
```

```
SELECT 'Δεν έχει γίνει σχετική αίτηση.' as ";
```

```
end if;
END$
```

DELIMITER ;

3. Το τρίτο Stored Procedure χρησιμοποιεί τους πίνακες **evaluation** και **evaluationresult** για τα στοιχεία που χρειάζεται. Γίνεται έλεγχος για την ύπαρξη ή μη των ολοκληρωμένων βαθμολογήσεων και στην πρώτη περίπτωση εμφανίζονται οι οριστικοποιημένοι πίνακες. Αυτοί περιέχουν το username του κάθε υποψηφίου, καθώς και το ID της εκάστοτε θέσης μαζί με τις τελικές αξιολογήσεις. Για τις αξιολογήσεις σε εξέλιξη εμφανίζεται ανάλογο μήνυμα σε συνδυασμό με τις ημιτελείς βαθμολογήσεις. Στην περίπτωση που δεν υπάρχουν αξιολογήσεις για κάποια θέση εργασίας εμφανίζεται αντίστοιχο μήνυμα.

```
CREATE PROCEDURE `job_progress`(id int(4))
BEGIN
```

```
declare incomplete int;
declare applications int;
```

```
SELECT
COUNT(*)
INTO applications FROM
requestevaluation
WHERE
job_id = id;
```

```
if (applications=0) then
select "Δεν υπάρχουν αξιολογήσεις σε εξέλιξη κούκλ@ μου"
as ";
else
```

```
SELECT COUNT(*)
INTO incomplete
FROM evaluation
WHERE job=id and empl_username NOT IN
```

```
(SELECT empl_username
FROM evaluation
WHERE phase = 3);
```

```
if (incomplete>=0) then
select "Οριστικοποιημένοι Πίνακες" as ";
select empl_username as 'Username', job_id as 'Job', grade as 'Grade'
from evaluationresult
where job_id=id
order by grade desc;
```

```
end if;
```

```
if (incomplete>0)
then
select "Αξιολόγηση σε εξέλιξη. Εκρεμμούν:" as ",incomplete as ";
end if;
```

```
end if;
END
```

# TRIGGERS

1. Τα πρώτα Triggers αφορούν την ενημέρωση του πίνακα **log**. Ο πίνακας αυτός ενημερώνεται μετά την διαγραφή, την ενημέρωση και την προσθήκη στοιχείων στους πίνακες **job**, **employee** και **requestevaluation**. Στον log προστίθεται το είδος της ενέργειας που εφαρμόστηκε στον πίνακα, μία τιμή που καθορίζει την επιτυχία ή την αποτυχία της ενέργειας (1 και 0 αντίστοιχα), το username του ατόμου που την εκτέλεσε, το timestamp, καθώς και το όνομα του πίνακα στον οποίο εφαρμόστηκε η ενέργεια. Έτσι, δημιουργήθηκαν ζεύγη Trigger για κάθε συμβάν (insert, update, delete), ένα πριν (για να δηλώνει το συμβάν) και ένα έπειτα το συμβάν (ώστε να πιστοποιεί την επιτυχημένη ολοκλήρωση του) σε καθέναν από τους πίνακες που αναφέρθηκαν παραπάνω. Παραθέτουμε ένα τέτοιο ζεύγος ως παράδειγμα

```
DELIMITER $

CREATE TRIGGER job_log_bupdate BEFORE UPDATE
ON job
FOR EACH ROW
BEGIN

set @lastlogged = now();
set @getuser = replace(user(), '@localhost:');
insert into log values(@getuser,@lastlogged,'update','job',false);

END$

CREATE TRIGGER job_log_aupdate AFTER UPDATE
ON job
FOR EACH ROW
BEGIN

update log
set succesful = true where timestamp = @lastlogged;

END$
DELIMITER ;
```

2. Το δεύτερο Trigger αποτρέπει την ανανέωση των πεδίων ΑΦΜ και ΔΟΥ της κάθε εταιρείας (σχέση **company**). Ενώ στον χρήστη θα εμφανίζεται η δυνατότητα τροποποίησης των πεδίων και η ενημέρωση θα δείχνει πως εκτελείται, στην πραγματικότητα οι νέες τιμές τους (NEW.AFM, NEW.DOY) θα συνεχίζουν να είναι οι παλιές (OLD.AFM, OLD.DOY).

```
DELIMITER $

CREATE TRIGGER company_info AFTER UPDATE ON
company
FOR EACH ROW
BEGIN

if (OLD.DOY != NEW.DOY OR OLD.AFM != NEW.AFM
OR OLD.NAME != NEW.NAME) then
set NEW.DOY = OLD.DOY;

set NEW.AFM = OLD.AFM;
set NEW.NAME = OLD.NAME;

select 'Οι αλλαγές αγνοήθηκαν' as '';
end if;

END$
DELIMITER ;
```

3. Το τρίτο Trigger εμφανίζει σφάλμα sql κάθε φορά που κάποιος χρήστης που δεν είναι administrator, προσπαθεί να αλλάξει στοιχεία στο προφίλ του που δεν έχει δικαίωμα αλλαγής. Για να καθοριστεί αν κάποιος χρήστης έχει δικαιώματα διαχειριστή χρησιμοποιείται το ειδικό πεδίο **user.is\_admin** .

```
CREATE PROCEDURE `job_progress`(id int(4))  
BEGIN
```

```
declare incomplete int;  
declare applications int;
```

```
SELECT  
    COUNT(*)  
INTO applications FROM  
    requestevaluation  
WHERE  
    job_id = id;
```

```
if (applications=0) then  
    select "Δεν υπάρχουν αξιολογήσεις σε εξέλιξη  
κούκλ@ μου" as " "  
else
```

```
SELECT COUNT(*)  
INTO incomplete  
FROM evaluation  
WHERE job=id and empl_username NOT IN  
    (SELECT empl_username  
    FROM evaluation  
    WHERE phase = 3);
```

```
if (incomplete>=0) then  
    select "Οριστικοποιημένοι Πίνακες" as " "  
    select empl_username as 'Username', job_id as 'Job', grade as  
    'Grade'  
    from evaluationresult  
    where job_id=id  
    order by grade desc;
```

```
end if;
```

```
if (incomplete>0)  
then  
    select "Αξιολόγηση σε εξέλιξη. Εκρεμμούν:" as  
    ",incomplete as " "  
end if;
```

```
end if;  
END
```