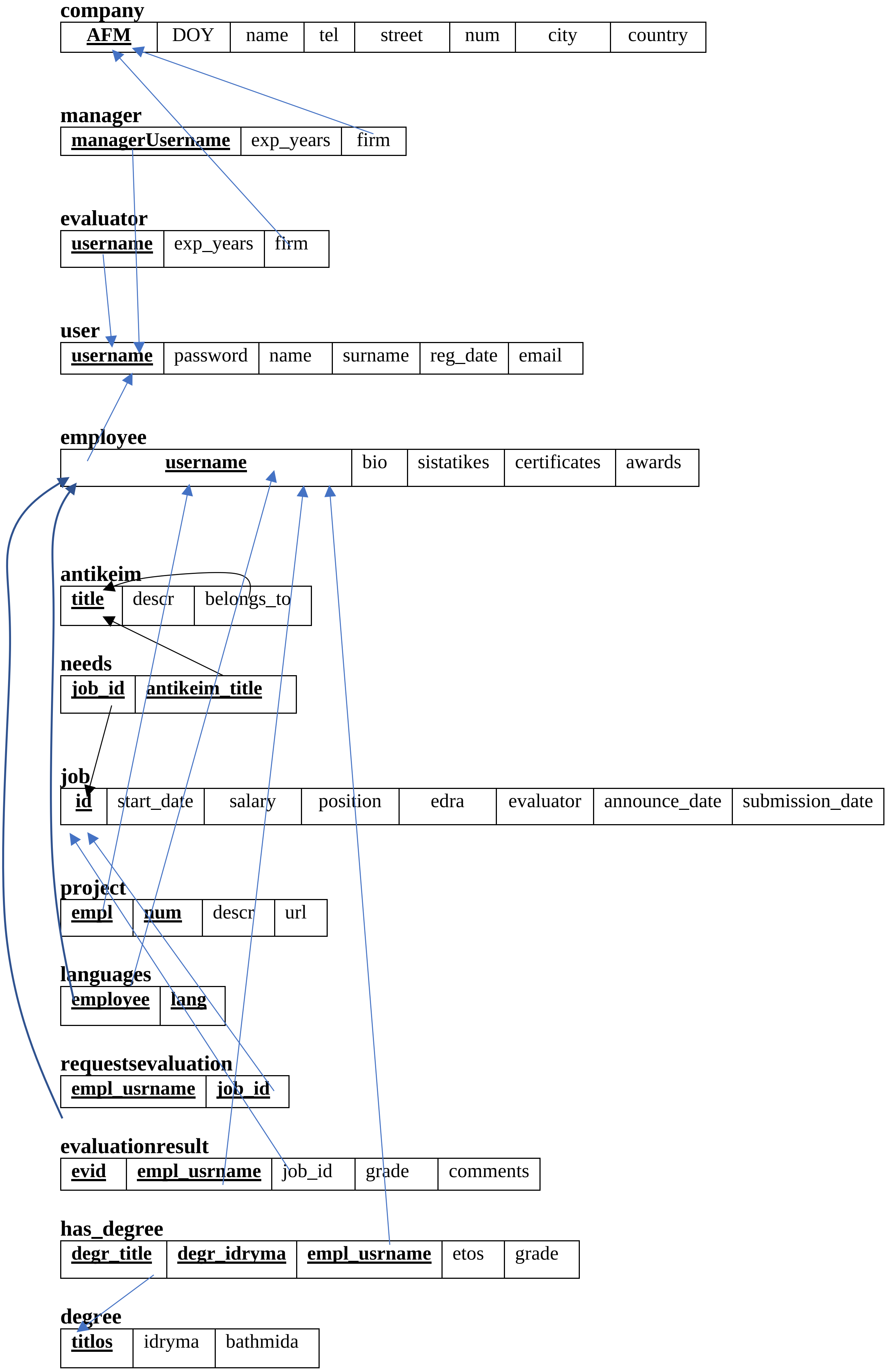
PROJECT ΣΤΟ ΕΡΓΑΣΤΗΡΙΟ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ 2020-2021

ΣΠΟΥΔΑΣΤΕΣ:

ΕΡΜΟΛΑΟΣ ΒΑΣΙΛΑΡΟΣ ΑΜ 1047093

ΒΑΣΙΛΗΣ ΤΡΙΠΟΛΙΔΗΣ ΑΜ 236365

1 Σχεσιακό Μοντέλο



company

AFM	DOY	name	tel	street	num	city	country

manager

managerUsername	exp_years	firm

evaluator

username	exp_years	firm

user

username	password	name	surname	reg_date	email

employee

username	bio	sistatikes	certificates	awards

antikeim

title	descr	belongs_to

needs

job_id	**antikeim_title**

job

id	start_date	salary	position	edra	evaluator	announce_date	submission_date

project

empl	**num**	descr	url

languages

employee	**lang**

requestsevaluation

empl_usrname	**job_id**

evaluationresult

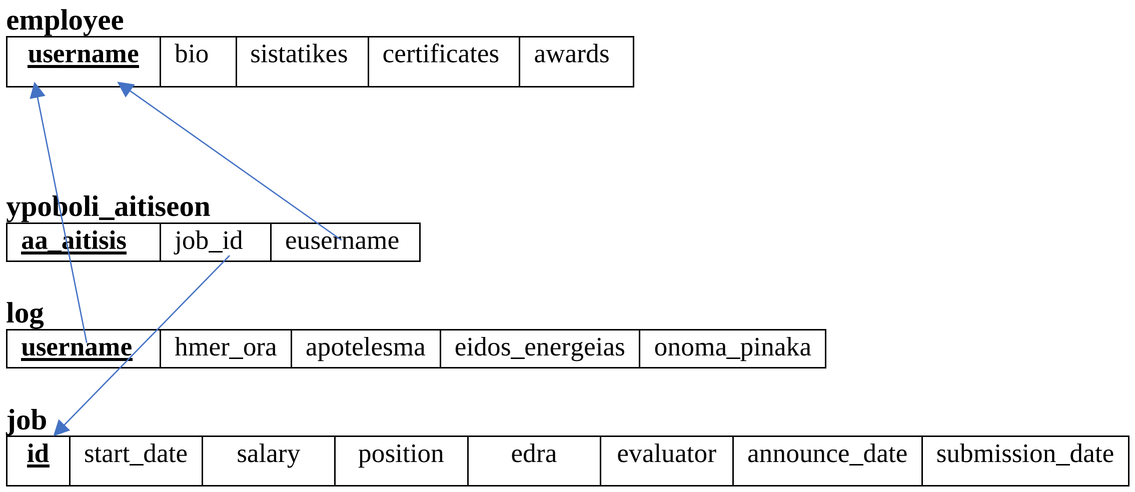
evid	**empl_usrname**	job_id	grade	comments

has_degree

degr_title	**degr_idryma**	**empl_usrname**	etos	grade

degree

titlos	idryma	bathmida



employee

username	bio	sistatikes	certificates	awards

ypoboli_aitiseon

aa_aitisis	job_id	eusername

log

username	hmer_ora	apotelesma	eidos_energeias	onoma_pinaka

job

id	start_date	salary	position	edra	evaluator	announce_date	submission_date

2 Πίνακες

2.1 Εντολές Δημιουργίας Πινάκων

```
mysql> drop database if exists staffevaluation;

Query OK, 17 rows affected (1.12 sec)


mysql> create database staffevaluation;

Query OK, 1 row affected (0.03 sec)


mysql> use staffevaluation;

Database changed

mysql> create table company(AFM char(9) primary key not null, DOY varchar(15), name varchar(35), phone bigint(16),street varchar(15),num tinyint(4), city varchar(15),country varchar(15));

Query OK, 0 rows affected, 2 warnings (0.01 sec)


mysql> create table antikeim(title varchar(35) primary key, descr tinytext, belongs_to varchar(36),constraint b foreign key(belongs_to) references antikeim(title));

Query OK, 0 rows affected (0.01 sec)


mysql> create table user(username varchar(12) primary key,password varchar(10),name varchar(25),surname varchar(35), reg_date datetime, email varchar(20),constraint c foreign key(username) references manager(username) on delete cascade on update cascade);

Query OK, 0 rows affected (0.01 sec)


mysql> create table manager(managerusername varchar(12),exp_years tinyint(4),firm char(9),primary key(managerusername,firm), constraint a1 foreign key(firm) references company(AFM), constraint a2 foreign key(managerusername) references user(username));

Query OK, 0 rows affected, 1 warning (0.01 sec)


mysql> create table employee(username varchar(12) primary key,bio text,sistatikes varchar(35), certificates varchar(35), awards varchar(35));
```

```
Query OK, 0 rows affected (0.01 sec)

mysql> create table evaluator(username varchar(12) primary key, exp_years tinyint(4), firm char(9), constraint d foreign key(username) references manager(username) on delete cascade on update cascade, constraint e foreign key(firm) references company(AFM) on delete cascade on update cascade);

Query OK, 0 rows affected, 1 warning (0.01 sec)

mysql> create table job(id int(4) primary key, start_date date, salary float(6,1),position varchar(40), edra varchar(45), evaluator varchar(12), announce_date datetime, submission_date date,constraint f foreign key(evaluator) references evaluator(username) on delete cascade on update cascade);

Query OK, 0 rows affected, 2 warnings (0.01 sec)

mysql> create table project(empl varchar(12),num tinyint(4),descr text, url varchar(60),primary key(empl,num), constraint f foreign key(num) references needs(job_id) on delete cascade on update cascade);

Query OK, 0 rows affected, 1 warning (0.01 sec)

mysql> create table needs(job_id int(4),antikeim_title varchar(36),primary key(job_id,antikeim_title), constraint g foreign key(antikeim_title) references antikeim(title) on delete cascade on update cascade);

Query OK, 0 rows affected, 1 warning (0.01 sec)

mysql> create table requestsevaluation(empl_username varchar(12),job_id int(4),primary key(empl_username,job_id),constraint h foreign key(empl_username) references employee(username) on delete cascade on update cascade, constraint j foreign key(job_id) references job(job_id) on delete cascade on update cascade);

Query OK, 0 rows affected, 1 warning (0.03 sec)

mysql> create table languages(employee varchar(12) primary key, lang set('EN','FR','SP','GR'), constraint k foreign key(employee) references employee(username) on delete cascade on update cascade);

Query OK, 0 rows affected (0.01 sec)

mysql> create table degree(titlos varchar(50), idryma varchar(40),primary key(titlos,idryma));
```

```
Query OK, 0 rows affected (0.01 sec)


mysql> create table has_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),constraint m foreign key(degr_title) references degree(title),constraint n foreign key(empl_username) references employee(username));

Query OK, 0 rows affected, 2 warnings (0.01 sec)


mysql> create table evaluationresult(Evld int(4) not null auto_increment, empl_username varchar(12), job_id int(4), grade int(4), comments varchar(255),primary key(evld,empl_username), constraint p foreign key(empl_username) references employee(username), constraint q foreign key(job_id) references job(job_id));

Query OK, 0 rows affected, 3 warnings (0.01 sec)


mysql> create table evaluation_process(aa int primary key,job_id int,

    -> bathmos_synenteyxis ENUM('0','1','2','3','4'),sxolia_synenetyxis text,xaraktirismos_synenteyxis ENUM('1','2','3'),

    -> bathmos_report ENUM('0','1','2','3','4'), sxolia_report text,xaraktirismos_report ENUM('1','2','3'),

    -> bathmos_loipon ENUM('0','1','2'),sxolia_loipon text,xaraktirismos_loipon ENUM('1','2','3'),

    -> username_eval varchar(12),

    -> username_emp varchar(12),

    -> teliko_apotelesma float (3,1),

    -> constraint z1 foreign key(username_eval) references evaluator(username),

    -> constraint z2 foreign key(username_emp) references employee(username),

    -> constraint z3 foreign key(job_id) references job(job_id));

Query OK, 0 rows affected, 1 warning (0.01 sec)


mysql> create table ypoboli_aitiseon(aa_aitisis int primary key auto_increment,job_id int, eusername varchar(12), constraint t foreign key(job_id) references job(job_id),constraint z foreign key(eusername) references employee(username));

Query OK, 0 rows affected (0.01 sec)
```

```
mysql> create table log(username varchar(12),hmer_ora datetime,apotelesma varchar(20),eidos_energeias varchar(20),onoma_pinaka varchar(20));
Query OK, 0 rows affected (0.01 sec)
```

2.2 Εντολές Εισαγωγής Εγγραφών

```
insert into company values('123456123','A Patras','Kataskeyasti Patras','2610-394509','Korinthou',10,'Patras','Greece');
insert into company values('123456888','B Patras','Emporio Sidiroy Patras','2610-324317','Avlonos',20,'Patras','Greece');
insert into company values('100045608','Pyrgou','Emporio Domion Ylikon','23610-52455','Korinthou',87,'Patras','Greece');
```

```
insert into user values('admin','admin','Manos','Giannakopoulos','2020-1-1','manos@gmail.com');
insert into user values('arg','000','Nikos','Argyriadis','2020-12-1','argiriad@gmail.com');
insert into user values('manolop','657','Kostas','Manolopoulos','2020-11-1','manolop@gmail.com');
insert into user values('giannakoy','123','Maria','Giannakou','2020-11-1','gianakou@gmail.com');
insert into user values('dimop','233','Giorgos','Dimopoulos','2019-11-10','dimop$hotmail.com');
insert into user values('delig','999','Andreas','Deligiannis','2018-6-10','delig$hotmail.com');
insert into user values('dalak','551','Anna','Dalaka','2017-1-9','dalak$hotmail.com');
insert into user values('papan','112','Nikos','Papanikolaou','2020-1-1','argiriad@hotmail.com');
insert into user values('papag','555','Kostas','Papageorgiou','2020-9-1','papag@hotmail.com');
insert into user values('papad','563','Vasilis','Papdionysiou','2020-8-1','papad@gmail.com');
insert into user values('xristop','345','Eleni','Papanikolaou','2020-51-1','xristop@gmail.com');
insert into user values('xorafop','346','Anna','Papageorgiou','2020-7-1','xorafop@gmail.com');
insert into user values('karela','347','Sofia','Karela','2020-4-1','karela@gmail.com');
```

```
insert into manager values('papan',10,'123456123');
insert into manager values('papag',8,'123456888');
insert into manager values('papad',7,'100045608');

insert into employee values('arg','Exei megali Proyphresia se networks','exei polles sistatikes','arg_certificates.pdf','arg_awards.pdf');
insert into employee values('manolop','Exei mikri Proyphresia se networks','exei liges sistatikes','manolop_certificates.pdf','manolop_awards.pdf');
insert into employee values('giannakoy','Exei sxetika megali Proyphresia se Databases','exei arrketes sistatikes','giannakoy_certificates.pdf','giannakoy_awards.pdf');
insert into employee values('dimop','Exei Proyphresia se Databases','exei sistatikes apo proigoumenes douleis','dimop_certificates.pdf','dimop_awards.pdf');
insert into employee values('delig','Exei doulepsei poly se ierarxikes Databases','exei liges sistatikes','delig.pdf','delig.pdf');
insert into employee values('dalak','Exei empeiria se Databases kai Networks','Polles sistatikes','dalak_certificates.pdf','dalak_awards.pdf');

insert into employee values('karela','Enai i pio kali evaluator','Polles systatikes','karela_certificates.pdf','karela_awards.pdf');
insert into employee values('xristop','Enai i pio kali evaluator','Liges systatikes','xristop_certificates.pdf','xristop_awards.pdf');
insert into employee values('xorafop','Enai i pio kali evaluator','Polles systatikes','xorafop_certificates.pdf','xorafop_awards.pdf');

insert into evaluator values('xristop',10,'123456123');
insert into evaluator values('xorafop',20,'123456888');
insert into evaluator values('karela',15,'100045608');
insert into evaluator values('karela',15,'123456888');

insert into antikeim values('Networks','Root Element',NULL);
insert into antikeim values('Local Area Networks','Second Level Element','Networks');
insert into antikeim values('Wide Area Networks','Second Level Element','Networks');
insert into antikeim values('Metropolitan Area Networks','Second Level Element','Networks');
insert into antikeim values('Databases','Root Element',NULL);
insert into antikeim values('MySQL','Second Level Element','Databases');
insert into antikeim values('SQL Server','Second Level Element','Databases');
insert into job values(1,'2020-1-1',1500,'Programmer','Patra','xristop','2020-1-2','2020-1-5');
insert into job values(2,'2020-2-1',1600,'Programmer','Athina','xristop','2020-2-3','2020-2-7');
insert into job values(3,'2020-3-1',1000,'Analyst','Pyrgos','xorafop','2020-3-6','2020-3-8');
insert into job values(4,'2020-4-1',1100,'Analyst','Thessaloniki','xorafop','2020-4-7','2020-4-10');
insert into job values(5,'2020-5-1',1600,'HelpDesk','Pyrgos','karela','2020-3-6','2020-5-8');
insert into job values(6,'2020-6-11',2000,'Assistant Manager','Patra','xorafop','2020-6-17','2020-6-27');
insert into job values(7,'2020-7-13',2200,'Assistant Manager','Patra','xristop','2020-7-17','2020-7-21');

insert into project values('arg',1,'Kataskeyi Topikou Diktyou','http:://www.ceid.upatras.gr');
insert into project values('arg',2,'Kataskeyi WAN','http:://www.ceid.upatras.gr');
insert into project values('manolop',1,'Kataskeyi sxesiakou RDBMS','http:://www.ceid.upatras.gr');
insert into project values('manolop',2,'Kataskeyi ierarxikou RDBMS','http:://www.ceid.upatras.gr');
insert into project values('arg',3,'Kataskeyi Topikou Diktyou','http:://www.diktya.gr');
insert into project values('arg',4,'Kataskeyi WAN','http:://www.ceid.upatras.gr');
insert into project values('manolop',5,'Kataskeyi sxesiakou RDBMS','http:://www.ece.upatras.gr');
insert into project values('manolop',6,'Kataskeyi dentrikou RDBMS','http:://www.ece.upatras.gr');
insert into project values('delig',1,'Epektasi Database Etaireias','http:://www.database.gr');
insert into project values('dalak',1,'Epektasi Diktyou Etaireias','http:://www.diktya.gr');

insert into needs values(1,'Networks');
insert into needs values(2,'Networks');
insert into needs values(3,'Local Area Networks');
insert into needs values(3,'Wide Area Networks');
insert into needs values(4,'Local Area Networks');
insert into needs values(5,'Databases');

insert into needs values(5,'SQL Server');


insert into evaluationresult values(",'arg',1,5,'Ta pige metria');

insert into evaluationresult values(",'manolop',2,9,'Ta pige arista');

insert into evaluationresult values(",'giannakoy',3,3,'Ta pige xalia');

insert into evaluationresult values(",'dimop',1,10,'Aristi eikona');

insert into evaluationresult values(",'delig',2,6,'Ta pige sxerika kala');

insert into evaluationresult values(",'dalak',3,7,'Ta pige kala');


insert into degree values('AEI','Panepistimio Patras');

insert into degree values('Master','Panepistimio Patras');

insert into degree values('PHD','Panepistimio Patras');

insert into degree values('AEI','Panepistimio Athinas');

insert into degree values('Master','Panepistimio Athinas');

insert into degree values('PHD','Panepistimio Athinas');


insert into has_degree values('AEI','Panepistimio Athinas','arg',2015,6.5);

insert into has_degree values('Master','Panepistimio Patras','arg',2018,6.5);

insert into has_degree values('AEI','Panepistimio Athinas','manolop',2019,6);

insert into has_degree values('PHD','Panepistimio Athinas','giannakoy',2019,8);

insert into has_degree values('AEI','Panepistimio Athinas','dimop',2020,5);

insert into has_degree values('Master','Panepistimio Patras','delig',2018,7.5);

insert into has_degree values('Master','Panepistimio Patras','dalak',2017,8.5);

insert into has_degree values('AEI','Panepistimio Patras','dalak',2015,5.5);
insert into languages values('arg','EN');

insert into languages values('manolop','GR');

insert into languages values('giannakoy','FR,GR');

insert into languages values('dimop','SP,GR');

insert into languages values('delig','EN,SP');

insert into languages values('dalak','EN,GR');


insert into requestsevaluation values('arg',1);

insert into requestsevaluation values('arg',2);

insert into requestsevaluation values('manolop',2);

insert into requestsevaluation values('manolop',3);

insert into requestsevaluation values('giannakoy',3);

insert into requestsevaluation values('giannakoy',4);

insert into requestsevaluation values('dimop',5);

insert into requestsevaluation values('dimop',6);

insert into requestsevaluation values('delig',1);

insert into requestsevaluation values('delig',6);

insert into requestsevaluation values('delig',3);

insert into requestsevaluation values('dalak',2);

insert into requestsevaluation values('dalak',4);

insert into requestsevaluation values('dalak',5);


insert into ypoboli_aitiseon values(",6,'arg');

insert into ypoboli_aitiseon values(",7,'manolop');
insert into ypoboli_aitiseon values(",6,'arg');

insert into ypoboli_aitiseon values(",7,'manolop');

insert into ypoboli_aitiseon values(",8,'manolop');


insert into evaluation_process values(1,1,3,'Metria Synenetyxi',3,3,'O dieythintis exei metria eikona',3,3,'Exei mono ptyxio kai den exei systatikes oute diakriseis',null,'karela','arg',null);

insert into evaluation_process values(2,2,3,'Kali Synenetyxi',2,2,'O dieythintis exei thetiki gnomi',2,1,'Exei Master kai 2 systatikes alla oxi diakriseis',1,'karela','arg',null);

insert into evaluation_process values(3,3,4,'Aristi Synenetyxi',3,4,'O dieythintis exei aristi gnomi',3,3,'Exei PHD ptyxio kai polles systatikes kai evresitexnia',1,'karela','manolop',null);

insert into evaluation_process values(4,1,4,'Poly Kali Synenetyxi',3,4,'O dieythintis ton proteinei gia tin thesi ayti',3,4,'Exei Master,PHD',null,'karela','manolop',null);

insert into evaluation_process values(5,3,2,null,3,null,'O dieythintis den ton proteinei gia tin thesi',3,4,'Exei Master,PHD',null,'karela','arg',null);

insert into evaluation_process values(6,1,1,'Elleipis Synenetyxi',1,2,'O dieythintis den exei kalh eikona',1,2,'Exei Ptyxio',2,'karela','giannakoy',null);
```

2.3 Μέρος Γ. Stored Procedures

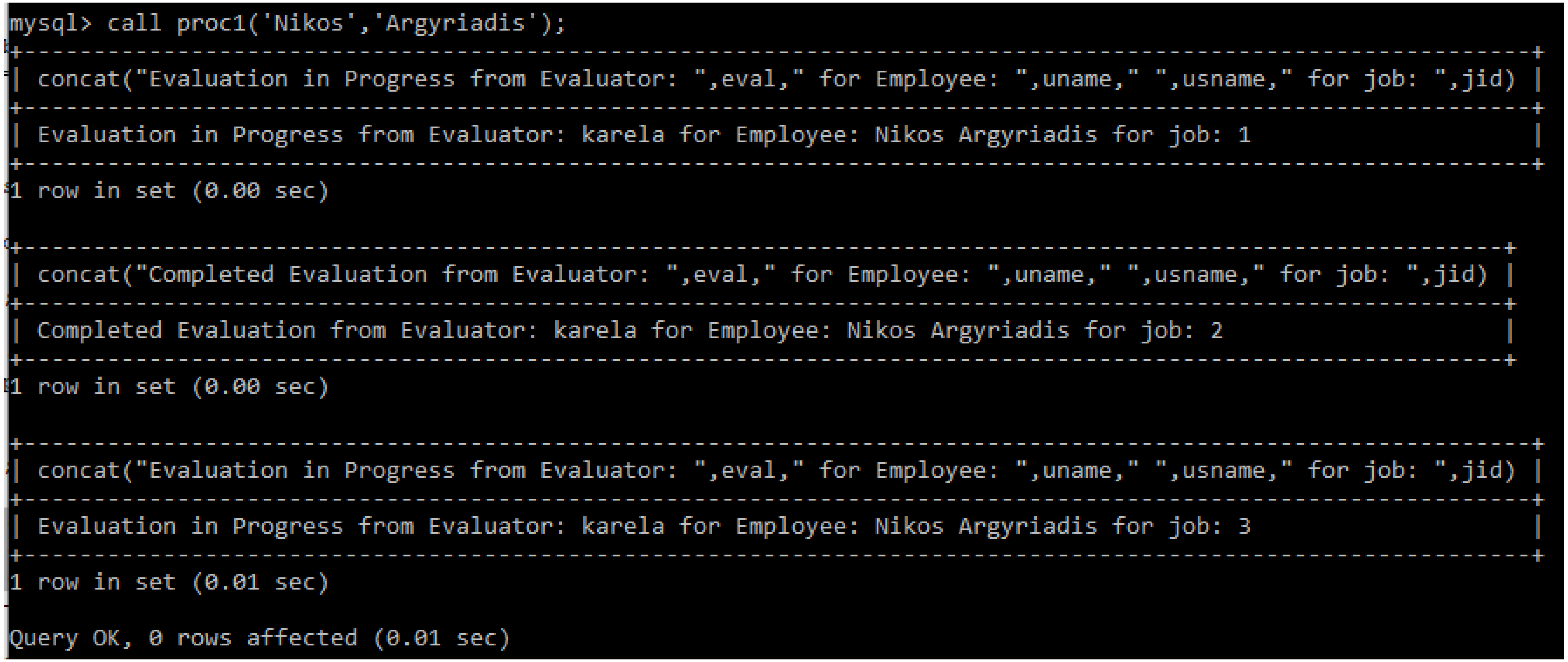
2.3.1 Stored Procedure 1

```
drop procedure if exists proc1;

delimiter $

create procedure proc1(IN ename varchar(25),IN esurname varchar(35))

begin

   declare nf int;

   declare ar_aitisis int;

   declare jid int;

   declare empl varchar(12);
```

```
    declare eval varchar(12);

    declare xs text;

    declare xr text;

    declare xl text;

    declare uname varchar(25);

    declare usname varchar(35);


    declare bcursor cursor for select distinct
evaluation_process.aa,evaluation_process.job_id,evaluation_process.username_emp,evaluation_process.username_eval,evaluation_process.xaraktirismos_synenteyxis,xaraktirismos_report,xaraktirismos_loipon,user.name,user.surname

    from user,employee,evaluation_process,evaluator,requestsevaluation

    where user.username=employee.username and employee.username=requestsevaluation.empl_username and

     employee.username=evaluation_process.username_emp and

    evaluator.username=evaluation_process.username_eval and user.name=ename and user.surname=esurname;

    declare continue handler for not found set nf=1;

    set nf=0;

    open bcursor;

    repeat

            fetch bcursor into ar_aitisis,jid,empl,eval,xs,xr,xl,uname,usname;

            if (nf=0) then


                    if (xs is not null) and (xr is not null) and (xl is not null) then

                            select concat("Completed Evaluation from Evaluator: ",eval," for Employee: ",uname," ",usname," for job: ",jid);

                    else

                            select concat("Evaluation in Progress from Evaluator: ",eval," for Employee: ",uname," ",usname," for job: ",jid);

                    end if;
            end if;
        until (nf=1)
            end repeat;
end$
delimiter ;
```

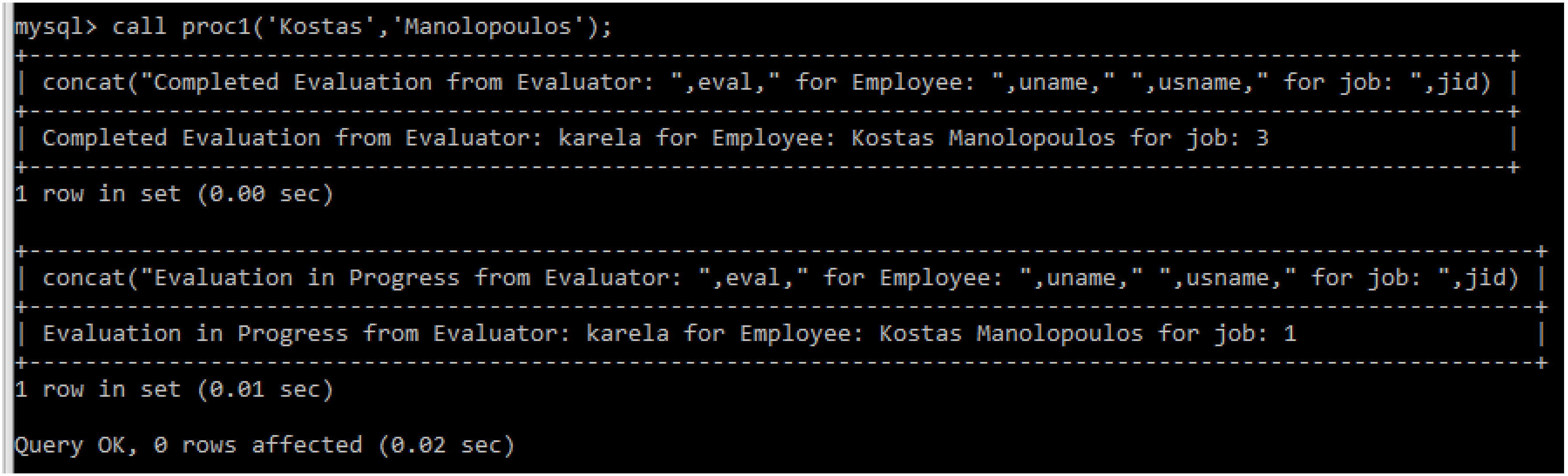
Αποτελέσματα Εκτέλεσης 1ης Stored Procedure

1η εκτέλεση



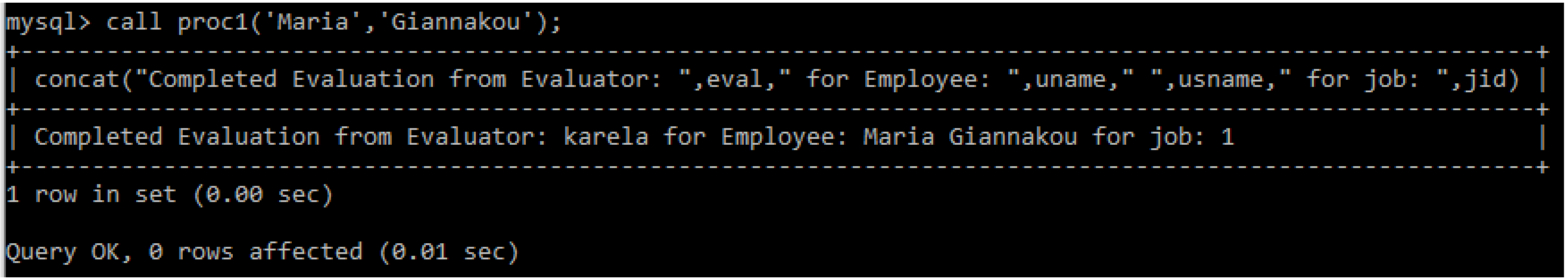
```
mysql> call proc1('Nikos','Argyriadis');
+-------------------------------------------------------------------------------------------------------------+
| concat("Evaluation in Progress from Evaluator: ",eval," for Employee: ",uname," ",usname," for job: ",jid) |
+-------------------------------------------------------------------------------------------------------------+
| Evaluation in Progress from Evaluator: karela for Employee: Nikos Argyriadis for job: 1                    |
+-------------------------------------------------------------------------------------------------------------+
1 row in set (0.00 sec)

+-----------------------------------------------------------------------------------------------------------+
| concat("Completed Evaluation from Evaluator: ",eval," for Employee: ",uname," ",usname," for job: ",jid) |
+-----------------------------------------------------------------------------------------------------------+
| Completed Evaluation from Evaluator: karela for Employee: Nikos Argyriadis for job: 2                    |
+-----------------------------------------------------------------------------------------------------------+
1 row in set (0.00 sec)

+-------------------------------------------------------------------------------------------------------------+
| concat("Evaluation in Progress from Evaluator: ",eval," for Employee: ",uname," ",usname," for job: ",jid) |
+-------------------------------------------------------------------------------------------------------------+
| Evaluation in Progress from Evaluator: karela for Employee: Nikos Argyriadis for job: 3                    |
+-------------------------------------------------------------------------------------------------------------+
1 row in set (0.01 sec)

Query OK, 0 rows affected (0.01 sec)
```

2η εκτέλεση



```
mysql> call proc1('Kostas','Manolopoulos');
+----------------------------------------------------------------------------------------------------------+
| concat("Completed Evaluation from Evaluator: ",eval," for Employee: ",uname," ",usname," for job: ",jid) |
+----------------------------------------------------------------------------------------------------------+
| Completed Evaluation from Evaluator: karela for Employee: Kostas Manolopoulos for job: 3                 |
+----------------------------------------------------------------------------------------------------------+
1 row in set (0.00 sec)

+------------------------------------------------------------------------------------------------------------+
| concat("Evaluation in Progress from Evaluator: ",eval," for Employee: ",uname," ",usname," for job: ",jid) |
+------------------------------------------------------------------------------------------------------------+
| Evaluation in Progress from Evaluator: karela for Employee: Kostas Manolopoulos for job: 1                 |
+------------------------------------------------------------------------------------------------------------+
1 row in set (0.01 sec)

Query OK, 0 rows affected (0.02 sec)
```

3^{η} εκτέλεση



```
mysql> call proc1('Maria','Giannakou');
+----------------------------------------------------------------------------------------------------------+
| concat("Completed Evaluation from Evaluator: ",eval," for Employee: ",uname," ",usname," for job: ",jid) |
+----------------------------------------------------------------------------------------------------------+
| Completed Evaluation from Evaluator: karela for Employee: Maria Giannakou for job: 1                     |
+----------------------------------------------------------------------------------------------------------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.01 sec)
```

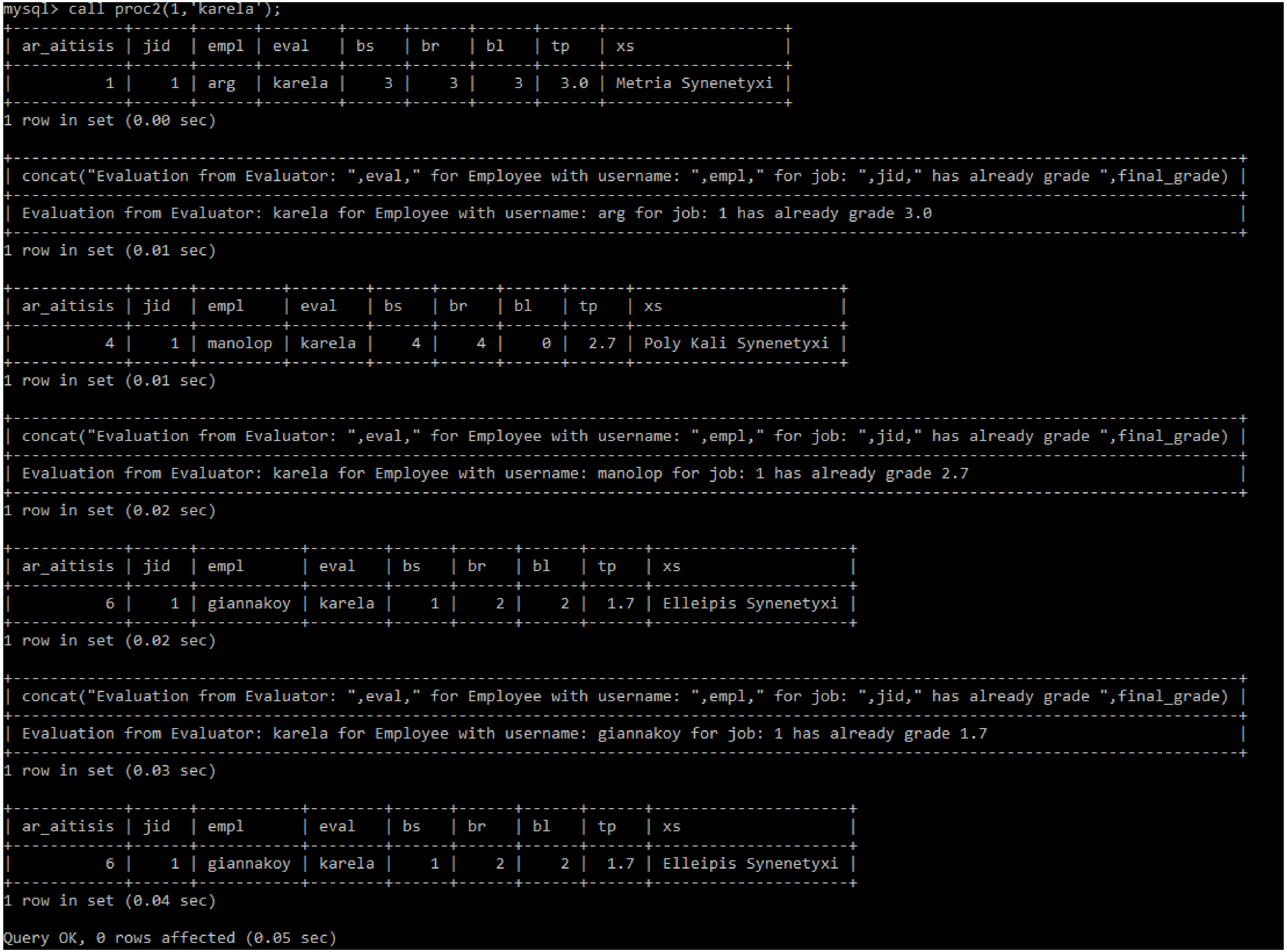
2.3.2 Stored Procedure 2

```
drop procedure if exists proc2;
delimiter $
create procedure proc2(IN jbdid int,IN evalusr varchar(12))
begin
  declare nf int;
  declare ar_aitisis int;
  declare jid int;
  declare empl varchar(12);
  declare eval varchar(12);
  declare bs int;
  declare br int;
  declare bl int;
  declare uname varchar(25);
  declare usname varchar(35);
  declare final_grade float(3,1);
  declare tp float(3,1);
  declare xs text;
  declare bcursor cursor for select distinct evaluation_process.aa,evaluation_process.job_id,evaluation_process.username_emp,evaluation_process.username_eval,evaluation_process.bathmos_synenteyxis,evaluation_process.bathmos_report,evaluation_process.bathmos_loipon,evaluation_process.teliko_apotelesma,evaluation_process.sxolia_synenetyxis from user,employee,evaluation_process,evaluator,requestsevaluation where user.username=employee.username and employee.username=requestsevaluation.empl_username and employee.username=evaluation_process.username_emp and evaluation_process.username_eval=evalusr and evaluation_process.job_id=jbdid;
  declare continue handler for not found set nf=1;
  set nf=0;
  open bcursor;
  repeat
```

```
    fetch bcursor into ar_aitisis,jid,empl,eval,bs,br,bl,tp,xs;
    select ar_aitisis,jid,empl,eval,bs,br,bl,tp,xs;
    if (nf=0) then
        if (bs is not null) and (br is not null) and (bl is not null) and (tp is null) then
            set final_grade=(bs+br+bl)/3;
            update evaluation_process set teliko_apotelesma=final_grade where evaluation_process.username_eval=evalusr and evaluation_process.job_id=jbdid
            and evaluation_process.username_emp=empl;
            insert into evaluationresult values(",empl,jid,final_grade,xs);
            select concat("Final Evaluation Grade from Evaluator: ",eval," for Employee with username: ",empl," for job: ",jid," is: ",final_grade);
        else
        set final_grade=(bs+br+bl)/3;
            if (bs is not null) and (br is not null) and (bl is not null) and (tp is not null) then
                select concat("Evaluation from Evaluator: ",eval," for Employee with username: ",empl," for job: ",jid," has already grade ",final_grade);
            end if;
        end if;
    end if;
  until (nf=1)
    end repeat;
end$ delimiter ;
```

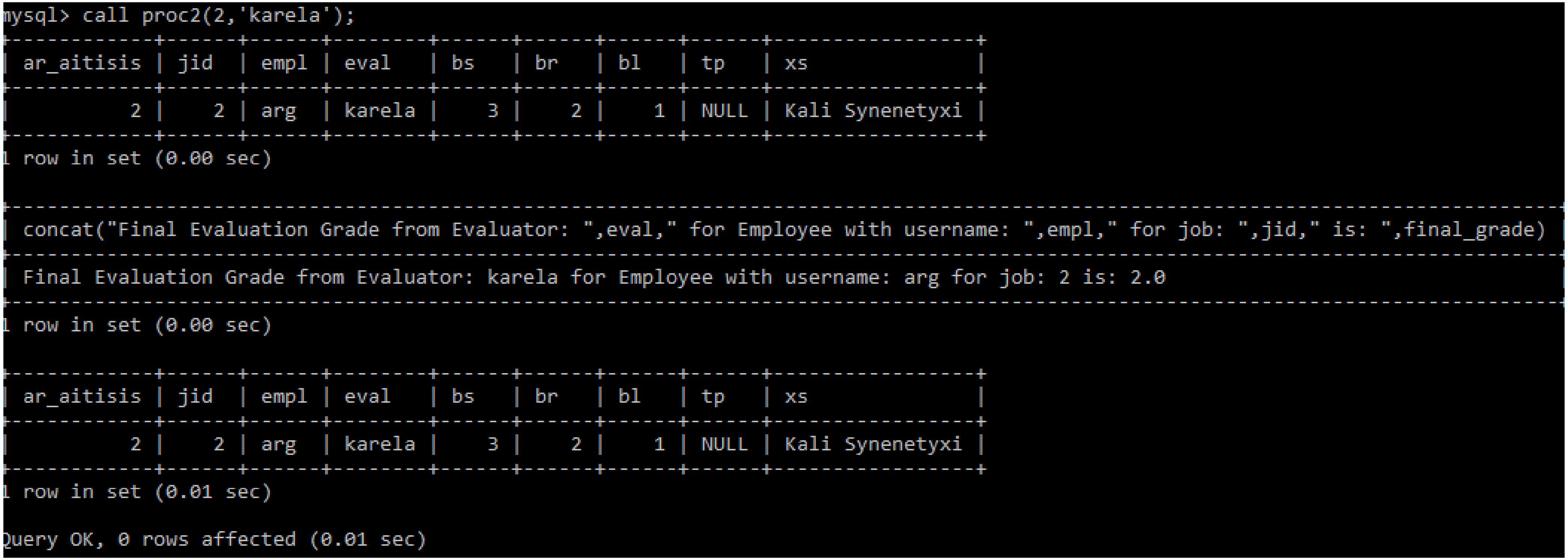
Αποτελέσματα Εκτέλεσης 2ης Stored Procedure

1η εκτέλεση



```
mysql> call proc2(1,'karela');
+------------+------+------+--------+------+------+------+------+-------------------+
| ar_aitisis | jid  | empl | eval   | bs   | br   | bl   | tp   | xs                |
+------------+------+------+--------+------+------+------+------+-------------------+
|          1 |    1 | arg  | karela |    3 |    3 |    3 |  3.0 | Metria Synenetyxi |
+------------+------+------+--------+------+------+------+------+-------------------+
1 row in set (0.00 sec)

+------------------------------------------------------------------------------------------------------------------------------------+
| concat("Evaluation from Evaluator: ",eval," for Employee with username: ",empl," for job: ",jid," has already grade ",final_grade) |
+------------------------------------------------------------------------------------------------------------------------------------+
| Evaluation from Evaluator: karela for Employee with username: arg for job: 1 has already grade 3.0                                 |
+------------------------------------------------------------------------------------------------------------------------------------+
1 row in set (0.01 sec)

+------------+------+---------+--------+------+------+------+------+----------------------+
| ar_aitisis | jid  | empl    | eval   | bs   | br   | bl   | tp   | xs                   |
+------------+------+---------+--------+------+------+------+------+----------------------+
|          4 |    1 | manolop | karela |    4 |    4 |    0 |  2.7 | Poly Kali Synenetyxi |
+------------+------+---------+--------+------+------+------+------+----------------------+
1 row in set (0.01 sec)

+------------------------------------------------------------------------------------------------------------------------------------+
| concat("Evaluation from Evaluator: ",eval," for Employee with username: ",empl," for job: ",jid," has already grade ",final_grade) |
+------------------------------------------------------------------------------------------------------------------------------------+
| Evaluation from Evaluator: karela for Employee with username: manolop for job: 1 has already grade 2.7                             |
+------------------------------------------------------------------------------------------------------------------------------------+
1 row in set (0.02 sec)

+------------+------+-----------+--------+------+------+------+------+---------------------+
| ar_aitisis | jid  | empl      | eval   | bs   | br   | bl   | tp   | xs                  |
+------------+------+-----------+--------+------+------+------+------+---------------------+
|          6 |    1 | giannakoy | karela |    1 |    2 |    2 |  1.7 | Elleipis Synenetyxi |
+------------+------+-----------+--------+------+------+------+------+---------------------+
1 row in set (0.02 sec)

+------------------------------------------------------------------------------------------------------------------------------------+
| concat("Evaluation from Evaluator: ",eval," for Employee with username: ",empl," for job: ",jid," has already grade ",final_grade) |
+------------------------------------------------------------------------------------------------------------------------------------+
| Evaluation from Evaluator: karela for Employee with username: giannakoy for job: 1 has already grade 1.7                           |
+------------------------------------------------------------------------------------------------------------------------------------+
1 row in set (0.03 sec)

+------------+------+-----------+--------+------+------+------+------+---------------------+
| ar_aitisis | jid  | empl      | eval   | bs   | br   | bl   | tp   | xs                  |
+------------+------+-----------+--------+------+------+------+------+---------------------+
|          6 |    1 | giannakoy | karela |    1 |    2 |    2 |  1.7 | Elleipis Synenetyxi |
+------------+------+-----------+--------+------+------+------+------+---------------------+
1 row in set (0.04 sec)

Query OK, 0 rows affected (0.05 sec)
```

2η εκτέλεση



```
mysql> call proc2(2,'karela');
+------------+------+------+--------+------+------+------+------+-----------------+
| ar_aitisis | jid  | empl | eval   | bs   | br   | bl   | tp   | xs              |
+------------+------+------+--------+------+------+------+------+-----------------+
|          2 |    2 | arg  | karela |    3 |    2 |    1 | NULL | Kali Synenetyxi |
+------------+------+------+--------+------+------+------+------+-----------------+
1 row in set (0.00 sec)

+------------------------------------------------------------------------------------------------------------------------------+
| concat("Final Evaluation Grade from Evaluator: ",eval," for Employee with username: ",empl," for job: ",jid," is: ",final_grade) |
+------------------------------------------------------------------------------------------------------------------------------+
| Final Evaluation Grade from Evaluator: karela for Employee with username: arg for job: 2 is: 2.0                             |
+------------------------------------------------------------------------------------------------------------------------------+
1 row in set (0.00 sec)

+------------+------+------+--------+------+------+------+------+-----------------+
| ar_aitisis | jid  | empl | eval   | bs   | br   | bl   | tp   | xs              |
+------------+------+------+--------+------+------+------+------+-----------------+
|          2 |    2 | arg  | karela |    3 |    2 |    1 | NULL | Kali Synenetyxi |
+------------+------+------+--------+------+------+------+------+-----------------+
1 row in set (0.01 sec)

Query OK, 0 rows affected (0.01 sec)
```

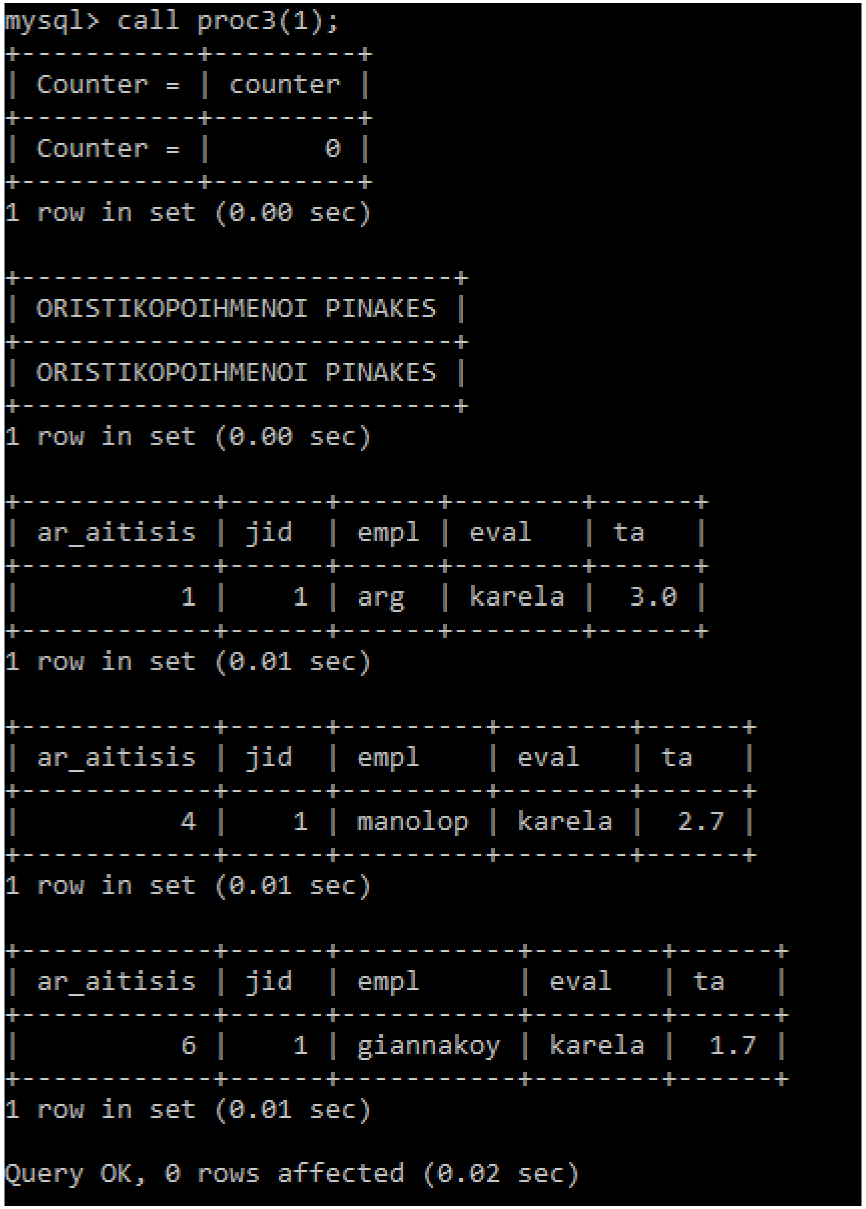
2.3.3 Stored Procedure 3

```
drop procedure if exists proc3;

delimiter $

create procedure proc3(IN jbdid int)

begin

   declare nf int;

   declare ar_aitisis int;

   declare jid int;

   declare empl varchar(12);

   declare eval varchar(12);

   declare bs int;

   declare br int;

   declare bl int;

   declare uname varchar(25);

   declare usname varchar(35);

   declare final_grade float(3,1);

   declare ta float(3,1);

   declare counter int;

   declare bcursor cursor for select distinct evaluation_process.aa,evaluation_process.job_id,evaluation_process.username_emp, evaluation_process.username_eval, evaluation_process.teliko_apotelesma from user,employee,evaluation_process,evaluator,requestsevaluation where user.username=employee.username and employee.username=requestsevaluation.empl_username and employee.username=evaluation_process.username_emp and evaluation_process.job_id=jbdid;

    declare bcursor2 cursor for select distinct evaluation_process.aa,evaluation_process.job_id, evaluation_process.username_emp,evaluation_process.username_eval, evaluation_process.teliko_apotelesma from user,employee,evaluation_process,evaluator,requestsevaluation where user.username=employee.username and employee.username=requestsevaluation.empl_username and employee.username=evaluation_process.username_emp and evaluation_process.job_id=jbdid order by evaluation_process.teliko_apotelesma desc;

            declare bcursor3 cursor for select distinct evaluation_process.aa,evaluation_process.job_id, evaluation_process.username_emp, evaluation_process.username_eval, evaluation_process.teliko_apotelesma from user, employee, evaluation_process,evaluator, requestsevaluation where user.username=employee.username and employee.username=requestsevaluation.empl_username and employee.username=evaluation_process.username_emp and   evaluation_process.job_id=jbdid;
```

```
 declare continue handler for not found set nf=1;
set nf=0;
set counter=0;
open bcursor;
repeat
        fetch bcursor into ar_aitisis,jid,empl,eval,ta;
        if (nf=0) then
                if (ta is null) then
                    set counter=counter+1;
                end if;
        end if;
  until (nf=1)
end repeat;
        select 'Counter =',counter;
        if (counter=0) then
                select 'ORISTIKOPOIHMENOI PINAKES';
                set nf=0;
                open bcursor2;
                repeat
                        fetch bcursor2 into ar_aitisis,jid,empl,eval,ta;
                        if (nf=0) then
                                if (ta is not null) then
                                        select ar_aitisis,jid,empl,eval,ta;
                                end if;
                        end if;
        until (nf=1)
        end repeat;
    end if;
  if (counter<>0) then
        select 'AXIOLOGHSH SE EXELIXI. EKKREMOUN AKOMA: ',counter;
        set nf=0;
        open bcursor3;
        repeat
            fetch bcursor3 into ar_aitisis,jid,empl,eval,ta;
            select ar_aitisis,jid,empl,eval,ta;
            select '---------------------------------';
            if (nf=0) then
                if (ta is not null) then
                    select ar_aitisis,jid,empl,eval,ta;
                end if;
            end if;
        until (nf=1)
        end repeat;
    end if;
end$
delimiter ;
```

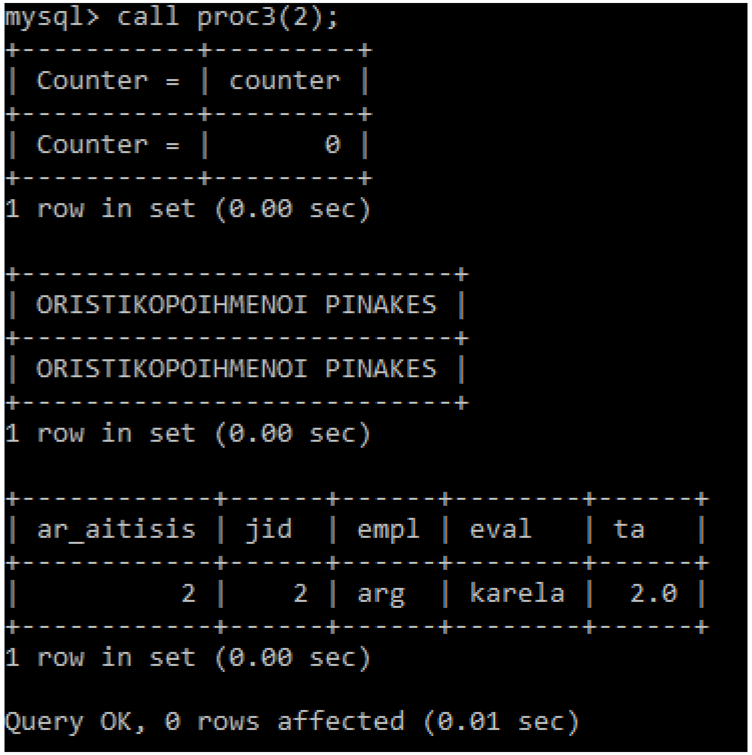
Αποτελέσματα Εκτέλεσης 3ης Stored Procedure

1η εκτέλεση



```
mysql> call proc3(1);
+-----------+---------+
| Counter = | counter |
+-----------+---------+
| Counter = |       0 |
+-----------+---------+
1 row in set (0.00 sec)

+--------------------------+
| ORISTIKOPOIHMENOI PINAKES |
+--------------------------+
| ORISTIKOPOIHMENOI PINAKES |
+--------------------------+
1 row in set (0.00 sec)

+------------+------+------+--------+------+
| ar_aitisis | jid  | empl | eval   | ta   |
+------------+------+------+--------+------+
|          1 |    1 | arg  | karela |  3.0 |
+------------+------+------+--------+------+
1 row in set (0.01 sec)

+------------+------+---------+--------+------+
| ar_aitisis | jid  | empl    | eval   | ta   |
+------------+------+---------+--------+------+
|          4 |    1 | manolop | karela |  2.7 |
+------------+------+---------+--------+------+
1 row in set (0.01 sec)

+------------+------+-----------+--------+------+
| ar_aitisis | jid  | empl      | eval   | ta   |
+------------+------+-----------+--------+------+
|          6 |    1 | giannakoy | karela |  1.7 |
+------------+------+-----------+--------+------+
1 row in set (0.01 sec)

Query OK, 0 rows affected (0.02 sec)
```

$2^η$ εκτέλεση



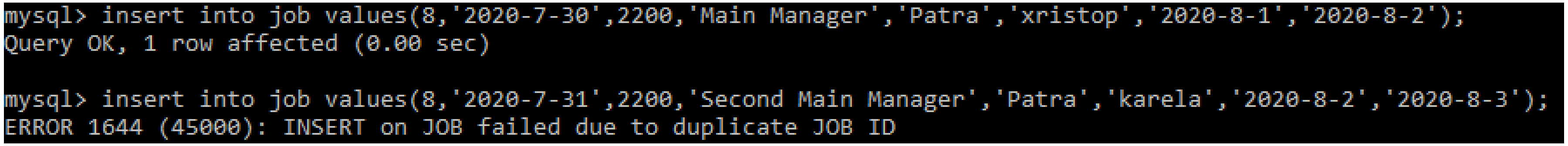
```
mysql> call proc3(2);
+-----------+---------+
| Counter = | counter |
+-----------+---------+
| Counter = |       0 |
+-----------+---------+
1 row in set (0.00 sec)

+-------------------------+
| ORISTIKOPOIHMENOI PINAKES |
+-------------------------+
| ORISTIKOPOIHMENOI PINAKES |
+-------------------------+
1 row in set (0.00 sec)

+------------+------+------+--------+------+
| ar_aitisis | jid  | empl | eval   | ta   |
+------------+------+------+--------+------+
|          2 |    2 | arg  | karela |  2.0 |
+------------+------+------+--------+------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.01 sec)
```

2.4 Μέρος Δ. Triggers

2.4.1 Trigger 1 στον πίνακα job

```
drop trigger if exists jobtrig1;

delimiter $

create trigger jobtrig1 before insert on job

for each row

begin

        DECLARE jid int;

        SET jid = (SELECT job.id FROM job WHERE job.id = NEW.id);


    IF jid IS NOT NULL THEN

        insert into log values(NEW.evaluator,now(),'Failure','insert','job');

                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'INSERT on JOB failed due to duplicate JOB ID';

        ELSE

                insert into log values(NEW.evaluator,now(),'Success','insert','job');

    END IF;

end$

delimiter ;
```

<u>Αποτελέσματα 2 Εκτελέσεων 1ου Trigger του πίνακα Job</u>

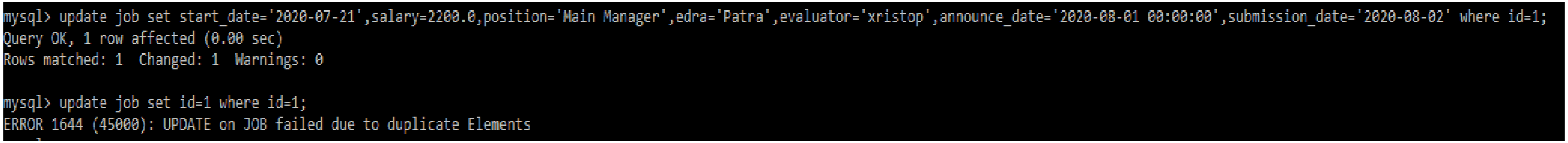


```
mysql> insert into job values(8,'2020-7-30',2200,'Main Manager','Patra','xristop','2020-8-1','2020-8-2');
Query OK, 1 row affected (0.00 sec)

mysql> insert into job values(8,'2020-7-31',2200,'Second Main Manager','Patra','karela','2020-8-2','2020-8-3');
ERROR 1644 (45000): INSERT on JOB failed due to duplicate JOB ID
```

2.4.2 Trigger 2 στον πίνακα job

```
drop trigger if exists jobtrig2;

delimiter $

create trigger jobtrig2 before update on job

for each row

begin

    IF new.id=old.id and new.start_date=old.start_date and new.salary=old.salary and new.position=old.position and new.edra=old.edra and new.evaluator=old.evaluator

        and new.announce_date=old.announce_date and new.submission_date=old.submission_date then

        insert into log values(NEW.evaluator,now(),'Failure','update','job');

                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on JOB failed due to duplicate Elements';

        ELSE

                insert into log values(NEW.evaluator,now(),'Success','update','job');

    END IF;

        IF (old.id <> new.id) THEN

                insert into log values(NEW.evaluator,now(),'Failure','update','Job');

                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on Jib failed due to EXISTING JOB ID';

        end if;

end$

delimiter ;
```

Αποτελέσματα 2 Εκτελέσεων 2^{ου} Trigger του πίνακα Job

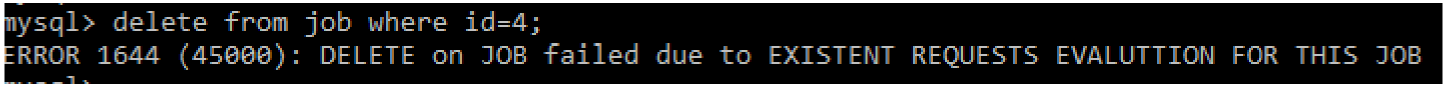


```
mysql> update job set start_date='2020-07-21',salary=2200.0,position='Main Manager',edra='Patra',evaluator='xristop',announce_date='2020-08-01 00:00:00',submission_date='2020-08-02' where id=1;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> update job set id=1 where id=1;
ERROR 1644 (45000): UPDATE on JOB failed due to duplicate Elements
```

2.4.3 Trigger 3 στον πίνακα job

```
drop trigger if exists jobtrig3;

delimiter $

create trigger jobtrig3 before delete on job

for each row

begin

    IF (EXISTS(SELECT count(*) FROM job inner join requestsevaluation on id=job_id)) THEN

        insert into log values(OLD.evaluator,now(),'Failure','delete','job');

                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'DELETE on JOB failed due to EXISTENT REQUESTS EVALUTTION FOR THIS JOB';

        else

                insert into log values(old.evaluator,now(),'Success','delete','job');

        end if;

        IF (EXISTS(SELECT count(*) FROM job inner join evaluationresult on id=job_id)) THEN

        insert into log values(OLD.evaluator,now(),'Failure','delete','job');

                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'DELETE on JOB failed due to EXISTENT EVALUTTION RESULTS FOR THIS JOB';

        else

                insert into log values(old.evaluator,now(),'Success','delete','job');

        end if;

end$

delimiter ;
```

<u>Αποτελέσματα 1 Δκτέλεσης 3ου Trigger του πίνακα Job</u>

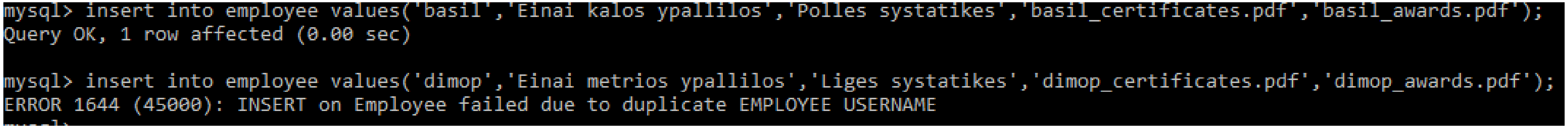


```
mysql> delete from job where id=4;
ERROR 1644 (45000): DELETE on JOB failed due to EXISTENT REQUESTS EVALUTTION FOR THIS JOB
```

2.4.4 Trigger 1 στον πίνακα Employee

```
drop trigger if exists employeetrig1;

delimiter $

create trigger employeetrig1 before insert on employee

for each row

begin

        DECLARE jeuname varchar(12);

        SET jeuname = (SELECT employee.username FROM employee WHERE employee.username   = NEW.username);

    IF jeuname IS NOT NULL THEN

        insert into log values(NEW.username,now(),'Failure','insert','employee');

                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'INSERT on Employee failed due to duplicate EMPLOYEE USERNAME';

        ELSE

                insert into log values(NEW.username,now(),'failure','insert','Empoyee');

    END IF;

end$

delimiter ;
```

<u>Αποτελέσματα 2 Εκτελέσεων 1ου Trigger του πίνακα Employee</u>

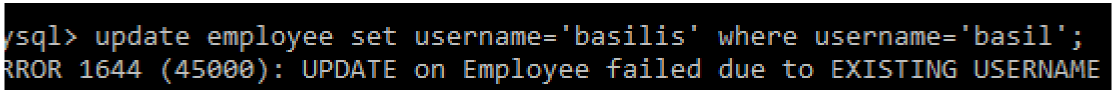


```
mysql> insert into employee values('basil','Einai kalos ypallilos','Polles systatikes','basil_certificates.pdf','basil_awards.pdf');
Query OK, 1 row affected (0.00 sec)

mysql> insert into employee values('dimop','Einai metrios ypallilos','Liges systatikes','dimop_certificates.pdf','dimop_awards.pdf');
ERROR 1644 (45000): INSERT on Employee failed due to duplicate EMPLOYEE USERNAME
mysql>
```

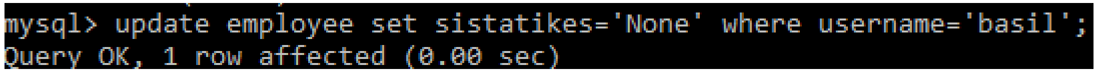
2.4.5 Trigger 2 στον πίνακα Employee

```
drop trigger if exists employeetrig2;

delimiter $

create trigger employeetrig2 before update on employee

for each row

begin

    IF new.username=old.username and new.bio=old.bio and new.sistatikes=old.sistatikes and new.certificates=old.certificates and new.awards=old.awards then

        insert into log values(NEW.username,now(),'Failure','update','Employee');

                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on Empoyee failed due to duplicate Elements';

        ELSE

                insert into log values(NEW.username,now(),'Success','update','Employee');

    END IF;

        IF (old.username <> new.username) THEN

                insert into log values(NEW.username,now(),'Failure','update','Employee');

                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on Employee failed due to EXISTING USERNAME';

        end if;

end$

delimiter ;
```

Αποτελέσματα 2 Εκτελέσεων 2ου Trigger του πίνακα Employee



```
ysql> update employee set username='basilis' where username='basil';
RROR 1644 (45000): UPDATE on Employee failed due to EXISTING USERNAME
```

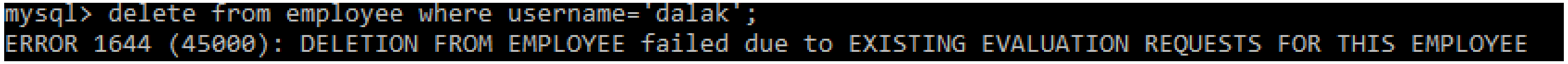


```
mysql> update employee set sistatikes='None' where username='basil';
Query OK, 1 row affected (0.00 sec)
```

2.4.6 Trigger 3 στον πίνακα Employee

```
drop trigger if exists employeetrig3;

delimiter $

create trigger employeetrig3 before delete on employee

for each row

begin

    IF (EXISTS(SELECT count(*) FROM employee inner join requestsevaluation on username=empl_username)) THEN

        insert into log values(OLD.username,now(),'Failure','delete','employee');

                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'DELETION FROM EMPLOYEE failed due to EXISTING EVALUATION REQUESTS FOR THIS EMPLOYEE';

        else

                insert into log values(old.username,now(),'Success','delete','employee');

        end if;

end$

delimiter ;
```

Αποτελέσματα 1 Εκτέλεσης 3ου Trigger του πίνακα Employee

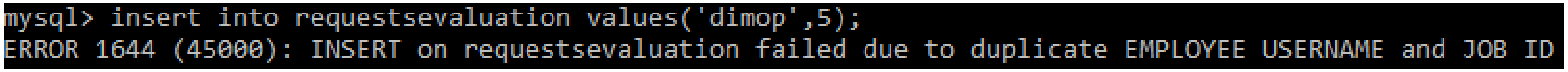


```
mysql> delete from employee where username='dalak';
ERROR 1644 (45000): DELETION FROM EMPLOYEE failed due to EXISTING EVALUATION REQUESTS FOR THIS EMPLOYEE
```

2.4.7 Trigger 1 στον πίνακα Requestsevaluation

```
drop trigger if exists requestsevaluationrig1;

delimiter $

create trigger requestsevaluationrig1 before insert on requestsevaluation

for each row

begin

        DECLARE jeuname varchar(12);

        SET jeuname = (SELECT requestsevaluation.empl_username FROM requestsevaluation WHERE requestsevaluation.empl_username=NEW.empl_username and requestsevaluation.job_id=new.job_id);

    IF jeuname IS NOT NULL THEN

        insert into log values(NEW.empl_username,now(),'Failure','insert','requestsevaluation');

                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'INSERT on requestsevaluation failed due to duplicate EMPLOYEE USERNAME and JOB ID';

        ELSE

                insert into log values(NEW.empl_username,now(),'failure','insert','requestsevaluation');

    END IF;

end$

delimiter ;

-- test

insert into requestsevaluation values('dimop',5);
```

Αποτελέσματα 1 Εκτέλεσης 1ου Trigger του πίνακα requestsevaluation

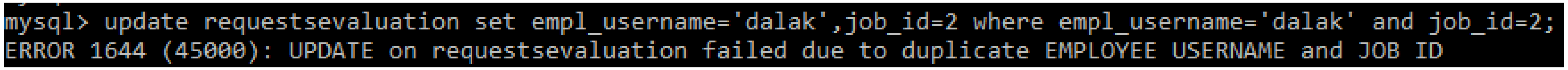


```
mysql> insert into requestsevaluation values('dimop',5);
ERROR 1644 (45000): INSERT on requestsevaluation failed due to duplicate EMPLOYEE USERNAME and JOB ID
```

2.4.8 Trigger 2 στον πίνακα Requestsevaluation

```
drop trigger if exists requestsevaluationrig2;

delimiter $

create trigger requestsevaluationrig2 before update on requestsevaluation

for each row

begin

        IF (old.empl_username = new.empl_username and old.job_id=new.job_id) THEN

                insert into log values(NEW.empl_username,now(),'Failure','Update','requestsevaluation');

                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on requestsevaluation failed due to duplicate EMPLOYEE USERNAME and JOB ID';

        ELSE

                insert into log values(NEW.empl_username,now(),'failure','update','requestsevaluation');

    END IF;

end$

delimiter ;
```

<u>Αποτελέσματα 1 Εκτέλεσης 2ου Trigger του πίνακα requestsevaluation</u>

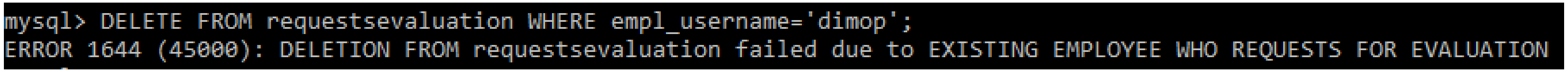


```
mysql> update requestsevaluation set empl_username='dalak',job_id=2 where empl_username='dalak' and job_id=2;
ERROR 1644 (45000): UPDATE on requestsevaluation failed due to duplicate EMPLOYEE USERNAME and JOB ID
```

2.4.9 Trigger 3 στον πίνακα Requestsevaluation

```
drop trigger if exists requestsevaluationrig3;
delimiter $
create trigger requestsevaluationrig3 before delete on requestsevaluation
for each row
begin
    IF (EXISTS(SELECT count(*) FROM requestsevaluation inner join employee on username=empl_username group by empl_username)) THEN
        insert into log values(OLD.empl_username,now(),'Failure','delete','requestsevaluation');
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'DELETION FROM requestsevaluation failed due to EXISTING EMPLOYEE WHO REQUESTS FOR EVALUATION';
        else
                insert into log values(old.empl_username,now(),'Success','delete','requestsevaluation');
        end if;
end$
delimiter ;
```

Αποτελέσματα 1 Εκτέλεσης 3ου Trigger του πίνακα requestsevaluation

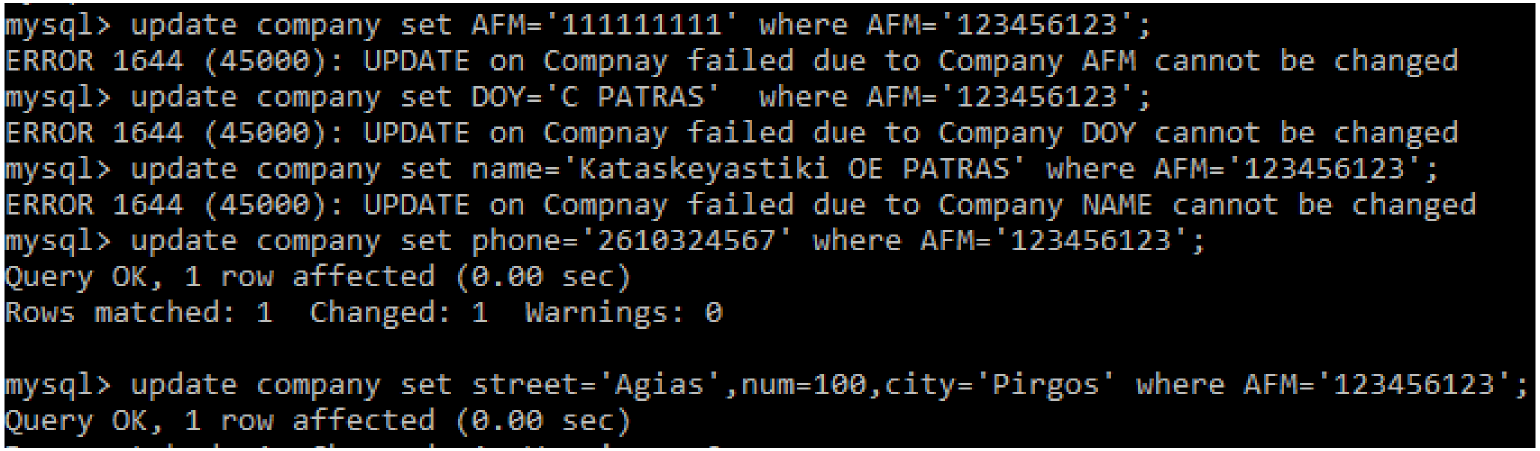


```
mysql> DELETE FROM requestsevaluation WHERE empl_username='dimop';
ERROR 1644 (45000): DELETION FROM requestsevaluation failed due to EXISTING EMPLOYEE WHO REQUESTS FOR EVALUATION
```

2.4.10 Trigger στον πίνακα company

```
drop trigger if exists companytrig;
delimiter $
create trigger companytrig before update on company
for each row
begin
        IF (old.AFM <> new.AFM) THEN
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on Compnay failed due to Company AFM cannot be changed';
    END IF;
        IF (old.DOY <> new.DOY) THEN
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on Compnay failed due to Company DOY cannot be changed';
    END IF;
        IF (old.name <> new.name) THEN
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on Compnay failed due to Company NAME cannot be changed';
    END IF;
end$
delimiter ;
```

<u>Αποτελέσματα 5 Εκτελέσεων Trigger του πίνακα company</u>

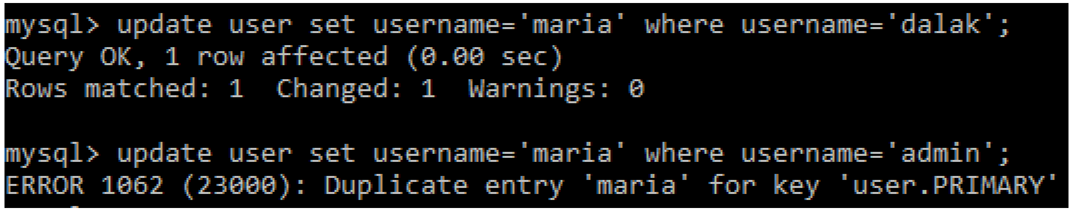


```
mysql> update company set AFM='111111111' where AFM='123456123';
ERROR 1644 (45000): UPDATE on Compnay failed due to Company AFM cannot be changed
mysql> update company set DOY='C PATRAS'  where AFM='123456123';
ERROR 1644 (45000): UPDATE on Compnay failed due to Company DOY cannot be changed
mysql> update company set name='Kataskeyastiki OE PATRAS' where AFM='123456123';
ERROR 1644 (45000): UPDATE on Compnay failed due to Company NAME cannot be changed
mysql> update company set phone='2610324567' where AFM='123456123';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> update company set street='Agias',num=100,city='Pirgos' where AFM='123456123';
Query OK, 1 row affected (0.00 sec)
```

2.4.11 Trigger στον πίνακα user

```
drop trigger if exists useradmintrig;

delimiter $

create trigger useradmintrig before update on user

for each row

begin

        IF (old.username <> new.username and old.username<>'admin') THEN

                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on User failed due to User Username cannot be changed';

    END IF;

end$

delimiter ;
```

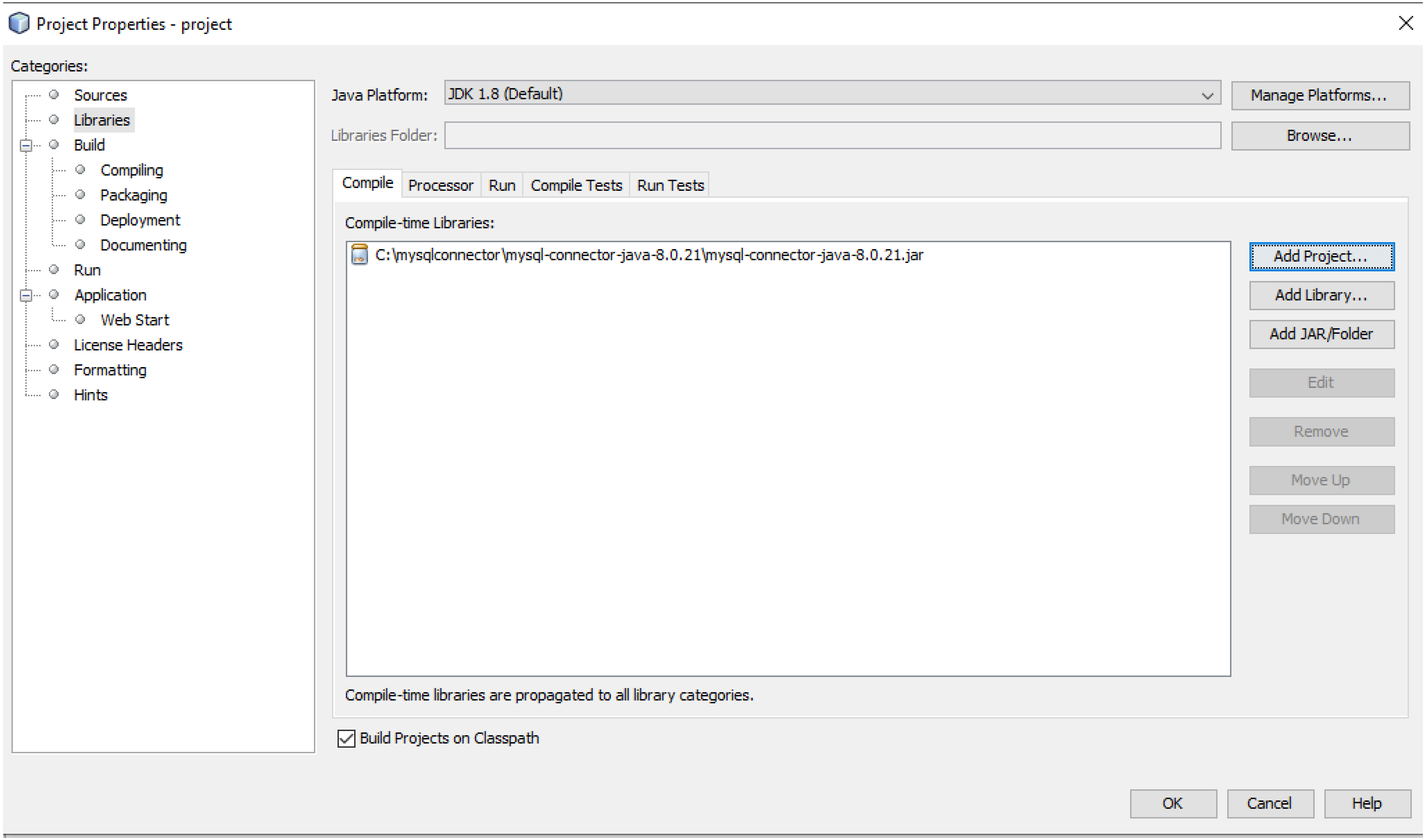
<u>Αποτελέσματα 2 Εκτελέσεων Trigger του πίνακα user</u>



```
mysql> update user set username='maria' where username='dalak';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> update user set username='maria' where username='admin';
ERROR 1062 (23000): Duplicate entry 'maria' for key 'user.PRIMARY'
```

3 Μέρος Β: GUIs

3.1 Σύνδεση Java με Mysql μέσω του πακέτου NetBeans



Project Properties - project

Java Platform: JDK 1.8 (Default)

Libraries Folder:

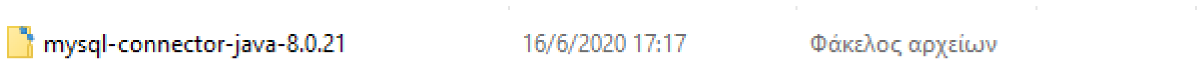
Compile-time Libraries:

C:\mysqlconnector\mysql-connector-java-8.0.21\mysql-connector-java-8.0.21.jar

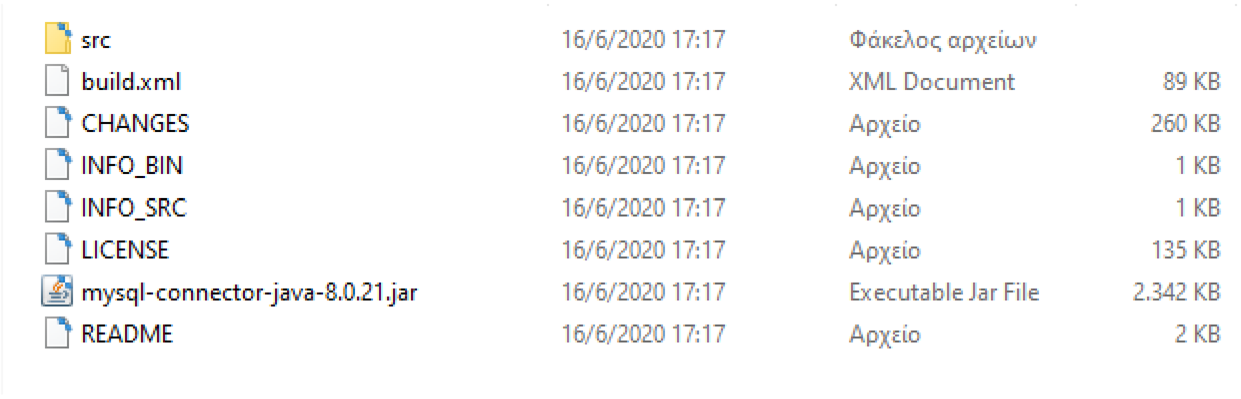
Compile-time libraries are propagated to all library categories.

☑ Build Projects on Classpath

Το .jar βρίσκεται στον ακόλουθο φάκελο



mysql-connector-java-8.0.21	16/6/2020 17:17	Φάκελος αρχείων	

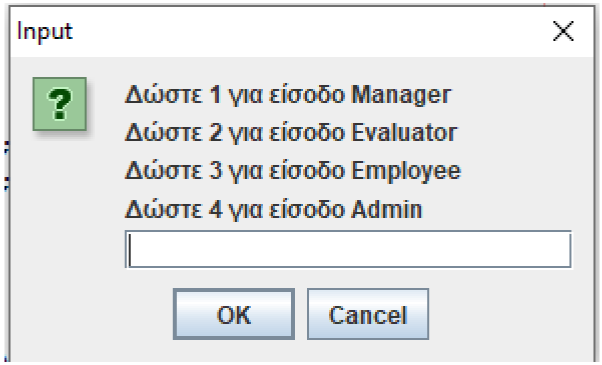


src	16/6/2020 17:17	Φάκελος αρχείων	
build.xml	16/6/2020 17:17	XML Document	89 KB
CHANGES	16/6/2020 17:17	Αρχείο	260 KB
INFO_BIN	16/6/2020 17:17	Αρχείο	1 KB
INFO_SRC	16/6/2020 17:17	Αρχείο	1 KB
LICENSE	16/6/2020 17:17	Αρχείο	135 KB
mysql-connector-java-8.0.21.jar	16/6/2020 17:17	Executable Jar File	2.342 KB
README	16/6/2020 17:17	Αρχείο	2 KB

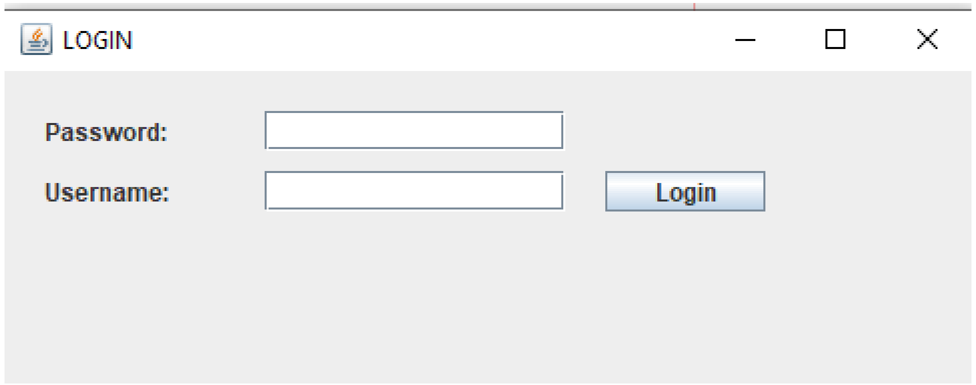
3.2 Λειτουργίες Διευθυντή (Manager)

Στα παραδείγματα που παραθέτουμε χρησιμοποιούμε ως manager το χρήστη με username papan ο οποίος ταυτόχρονα είναι και evaluator.

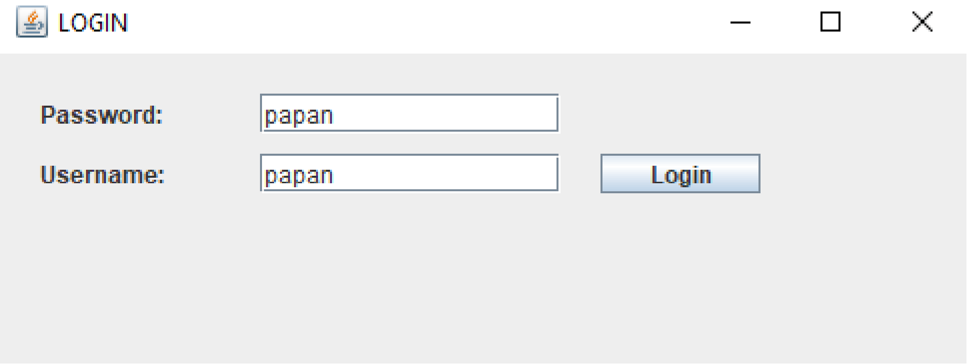
Αρχικά γίνεται είσοδος του Manager εκτελώντας το αρχείο login.java. Εμφανίζεται το ακόλουθο παράθυρο:


Δώστε 1 για είσοδο Manager
Δώστε 2 για είσοδο Evaluator
Δώστε 3 για είσοδο Employee
Δώστε 4 για είσοδο Admin

Ακολούθως εμφανίζεται το παράθυρο LOGIN για να εισάγει ο Manager τα στοιχεία του:


Password:
Username:

Συμπληρώνει τα στοιχεία του:

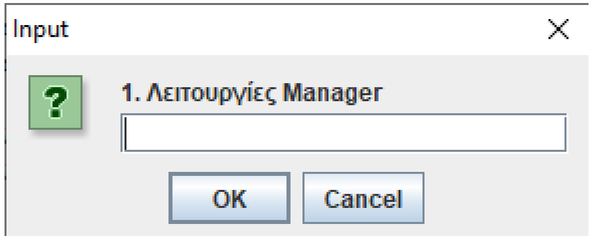


LOGIN

Password: papan

Username: papan

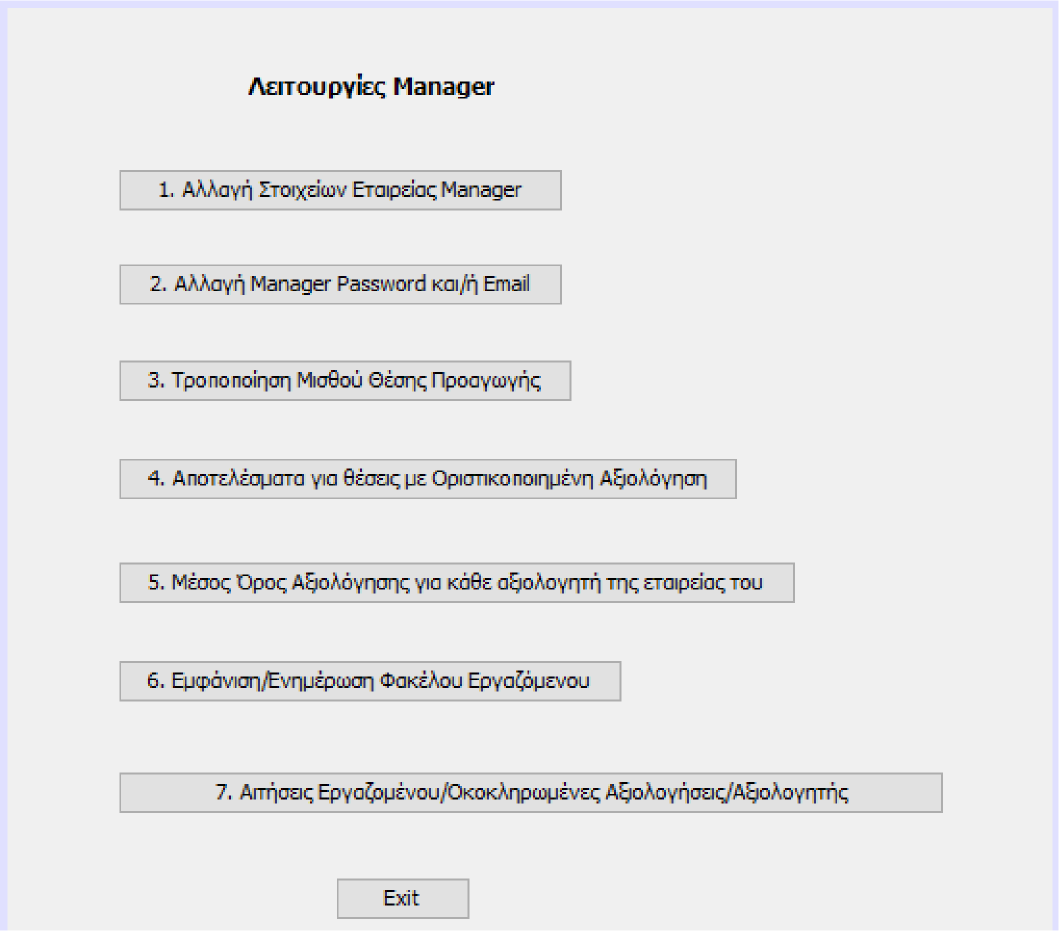
Εμφανίζεται το επόμενο παράθυρο στο οποίο εισάγει το 1:



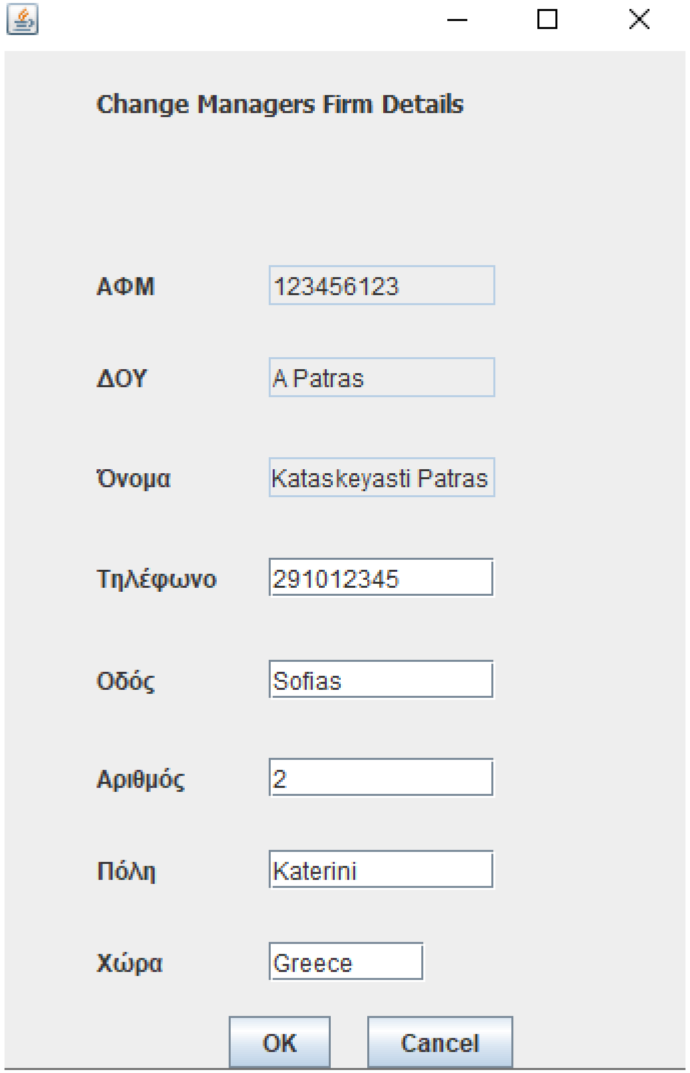
Input

1. Λειτουργίες Manager

Μετά εμφανίζεται το μενού με τις λειτουργίες του:


Λειτουργίες Manager
1. Αλλαγή Στοιχείων Εταιρείας Manager
2. Αλλαγή Manager Password και/ή Email
3. Τροποποίηση Μισθού Θέσης Προαγωγής
4. Αποτελέσματα για θέσεις με Οριστικοποιημένη Αξιολόγηση
5. Μέσος Όρος Αξιολόγησης για κάθε αξιολογητή της εταιρείας του
6. Εμφάνιση/Ενημέρωση Φακέλου Εργαζόμενου
7. Αιτήσεις Εργαζομένου/Οκοκληρωμένες Αξιολογήσεις/Αξιολογητής
Exit

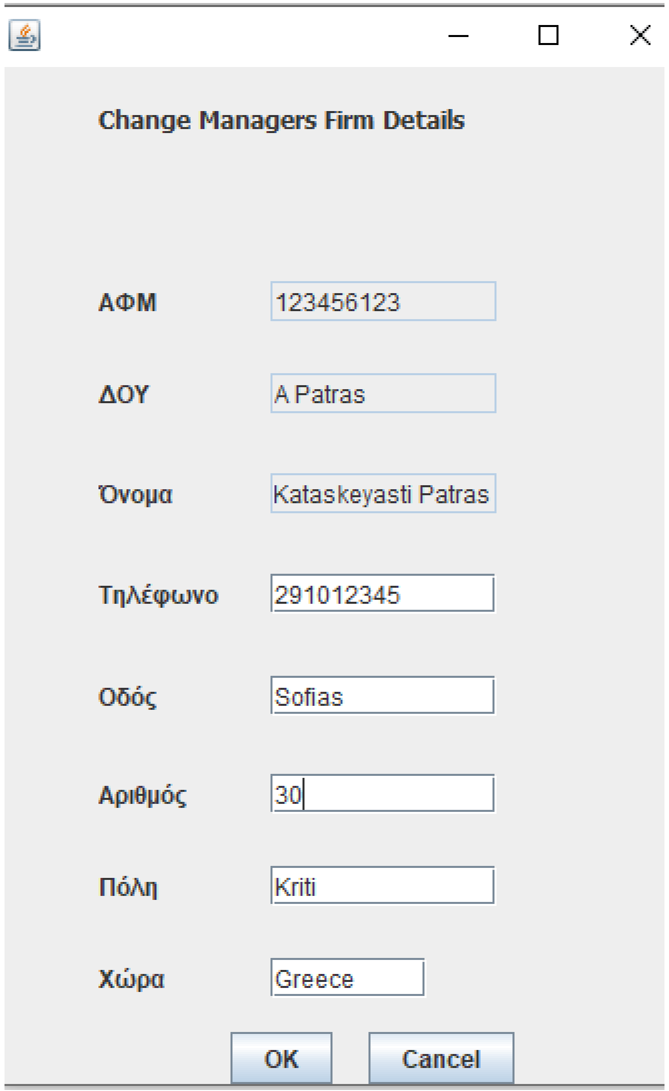
3.2.1 Λειτουργία 1 - Αλλαγή στοιχείων εταιρείας εκτός από το ΑΦΜ, τη ΔΟΥ και την επωνυμία (name)



Change Managers Firm Details

ΑΦΜ	123456123
ΔΟΥ	A Patras
Όνομα	Kataskeyasti Patras
Τηλέφωνο	291012345
Οδός	Sofias
Αριθμός	2
Πόλη	Katerini
Χώρα	Greece

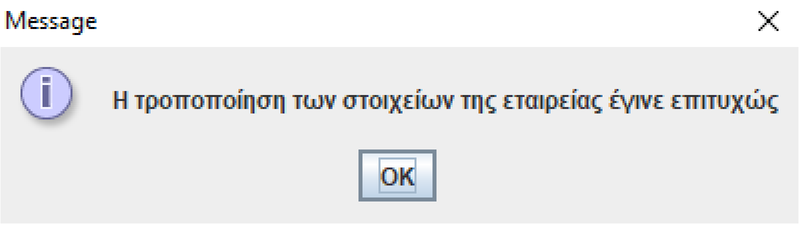
Παρατηρούμε ότι τα πεδία ΑΦΜ, ΔΟΥ και η επωνυμία της εταιρείας του είναι κλειδωμένα ώστε να μην μπορούν να τροποποιηθούν. Μπορεί να αλλάξει τα υπόλοιπα στοιχεία και πατώντας OK αυτά καταχωρούνται στη βάση δεδομένων staff evaluation και ενημερώνουν τον πίνακα company. Στην επόμενη εικόνα παραθέτουμε κάποιες ενδεικτικές αλλαγές:



Change Managers Firm Details

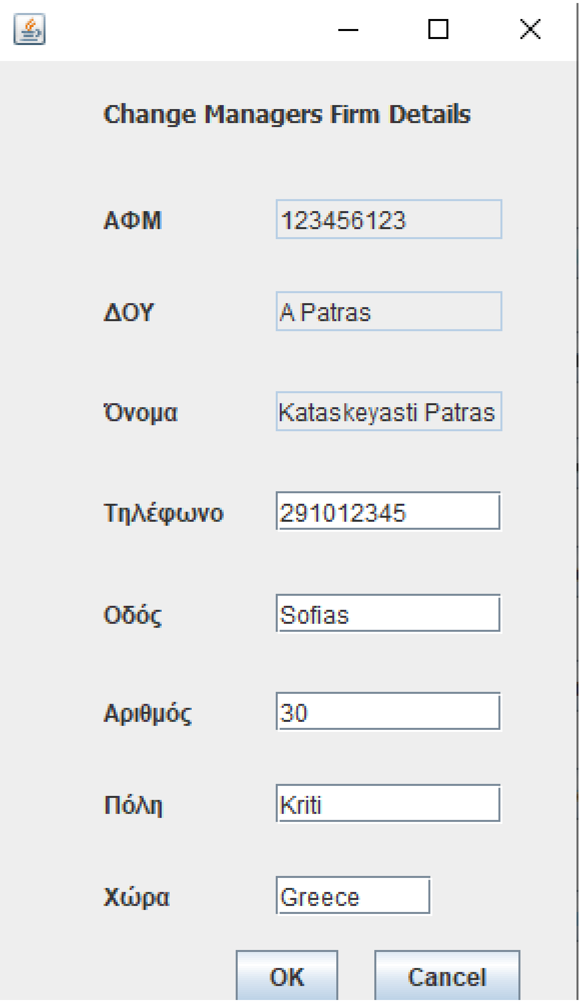
ΑΦΜ	123456123
ΔΟΥ	A Patras
Όνομα	Kataskeyasti Patras
Τηλέφωνο	291012345
Οδός	Sofias
Αριθμός	30
Πόλη	Kriti
Χώρα	Greece

Πατώντας OK και μετά επιλέγοντας πάλι την ίδια λειτουργία εμφανίζεται αρχικά η εικόνα:



Message

Η τροποποίηση των στοιχείων της εταιρείας έγινε επιτυχώς και μετά η εικόνα:



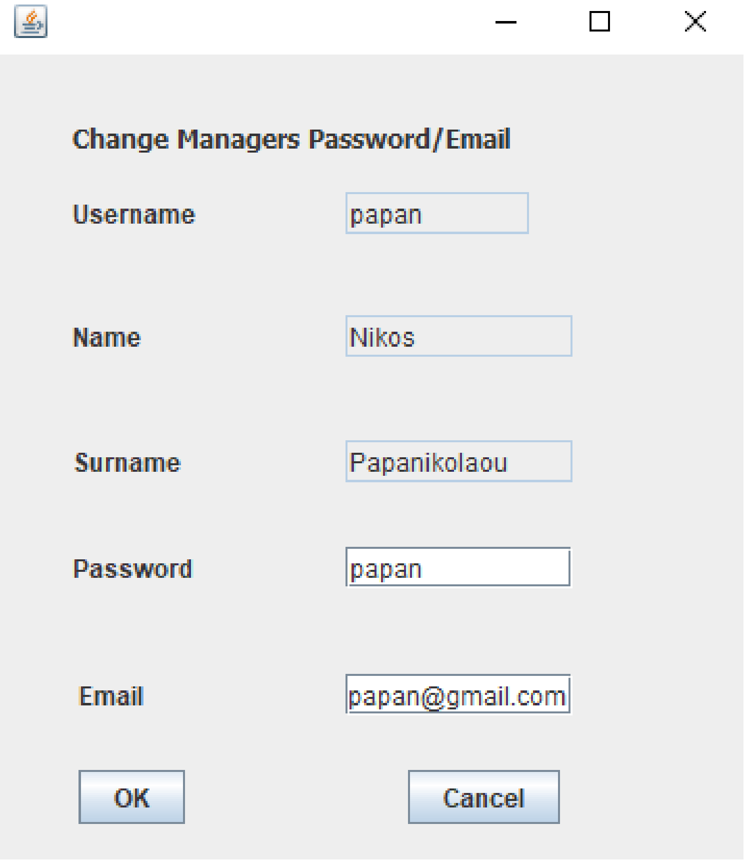
Change Managers Firm Details

ΑΦΜ	123456123
ΔΟΥ	A Patras
Όνομα	Kataskeyasti Patras
Τηλέφωνο	291012345
Οδός	Sofias
Αριθμός	30
Πόλη	Kriti
Χώρα	Greece

που δείχνει επιτυχή τροποποίηση

Ο κώδικας αυτής της λειτουργίας υλοποιείται στο αρχείο:

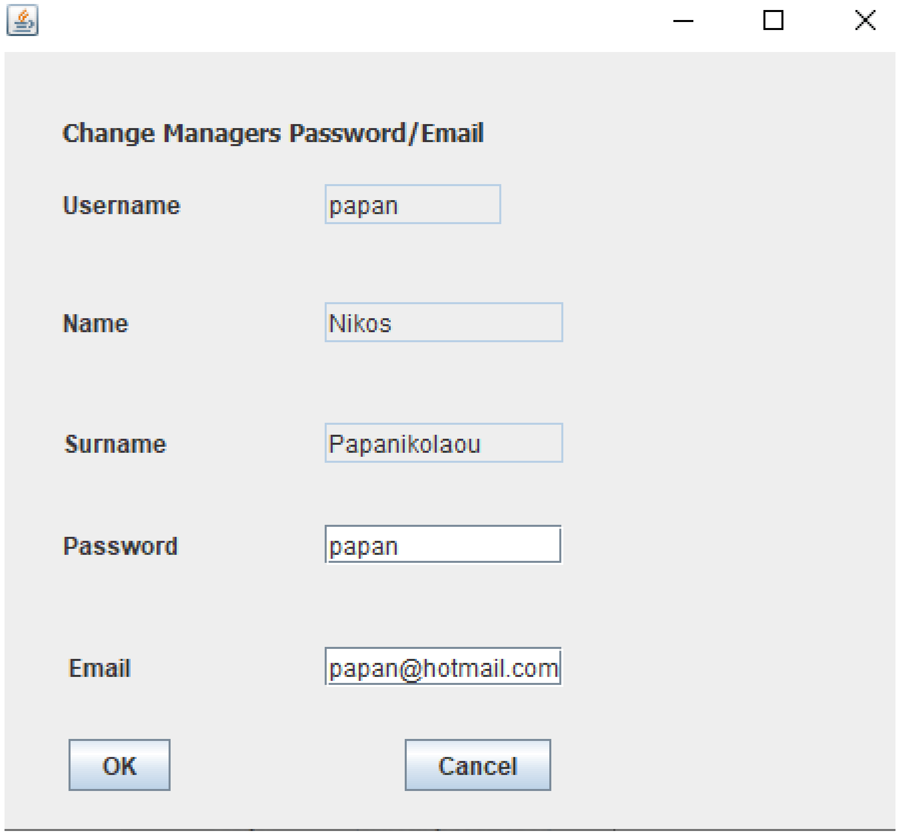
3.2.2 Λειτουργία 2 - Αλλαγή στοιχείων εταιρείας εκτός από το ΑΦΜ, τη ΔΟΥ και την επωνυμία (name)



Change Managers Password/Email

Username	papan
Name	Nikos
Surname	Papanikolaou
Password	papan
Email	papan@gmail.com

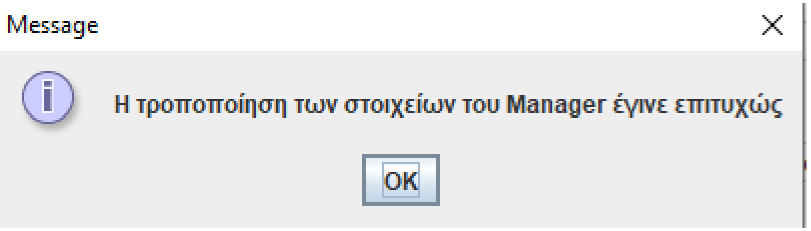
Παρατηρούμε ότι τα πεδία Username, Name, Surname και Password είναι κλειδωμένα ώστε να μην μπορούν να τροποποιηθούν. Μπορεί να αλλάξει τα υπόλοιπα στοιχεία και πατώντας OK αυτά καταχωρούνται στη βάση δεδομένων staff evaluation και ενημερώνουν τον πίνακα user. Στην επόμενη εικόνα παραθέτουμε κάποιες ενδεικτικές αλλαγές:



Change Managers Password/Email

Username	papan
Name	Nikos
Surname	Papanikolaou
Password	papan
Email	papan@hotmail.com

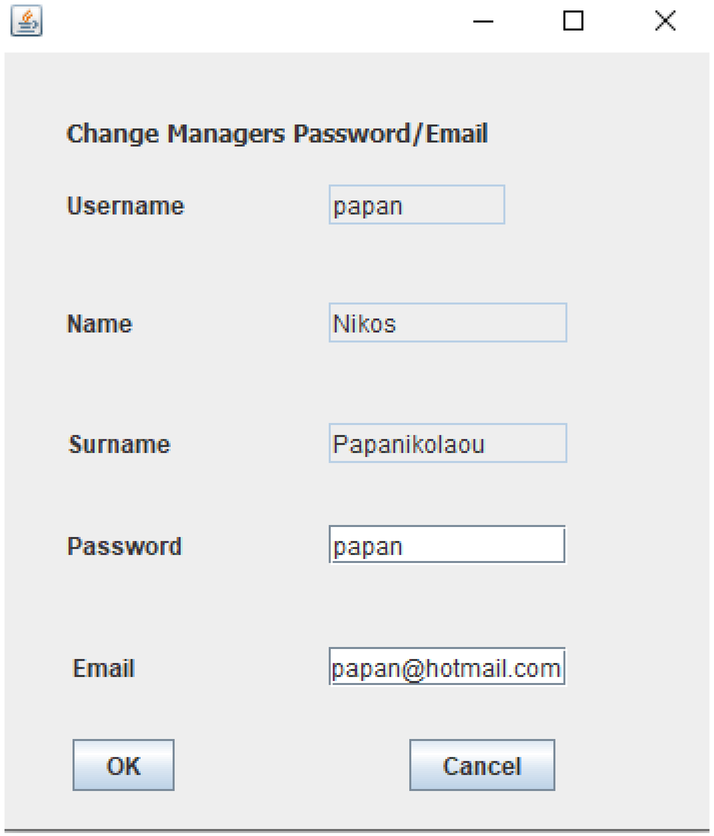
Πατώντας OK εμφανίζεται το παράθυρο:



Message

Η τροποποίηση των στοιχείων του Manager έγινε επιτυχώς

Επιλέγοντας πάλι την ίδια λειτουργία παρατηρούμε ότ ο αλλαγές έχουν γίνει:



Change Managers Password/Email

Username: papan

Name: Nikos

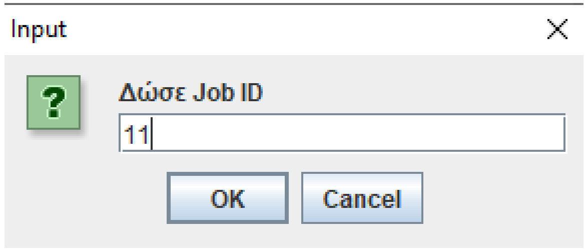
Surname: Papanikolaou

Password: papan

Email: papan@hotmail.com

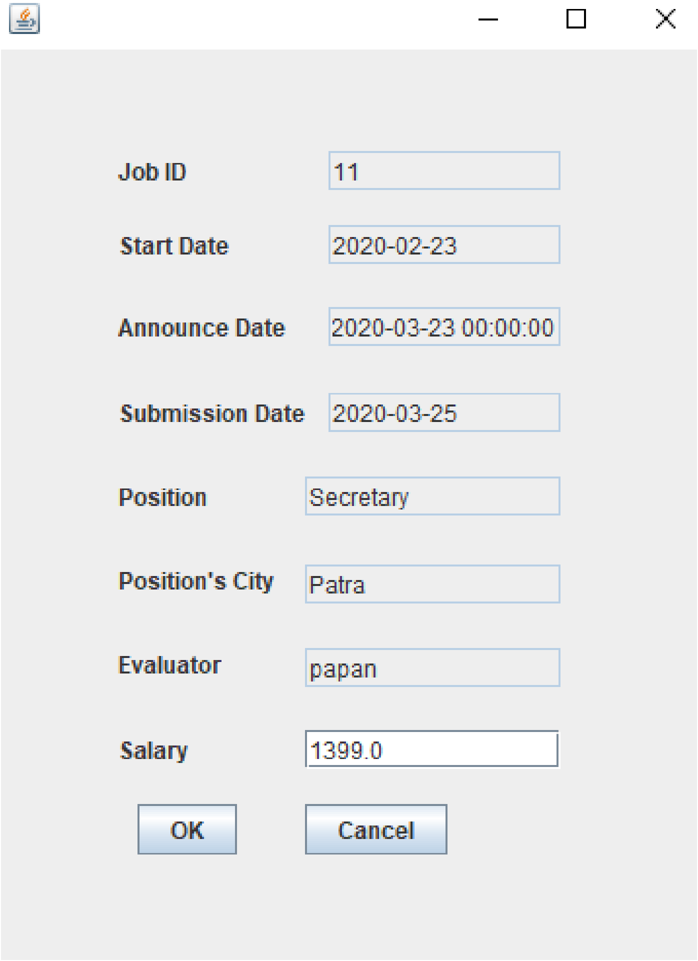
3.2.3 Λειτουργία 3 - Αλλαγή Μισθού Θέσης Προαγωγής

Δίνουμε αρχικά ένα JOB Id όπως στην επόμενη εικόνα:



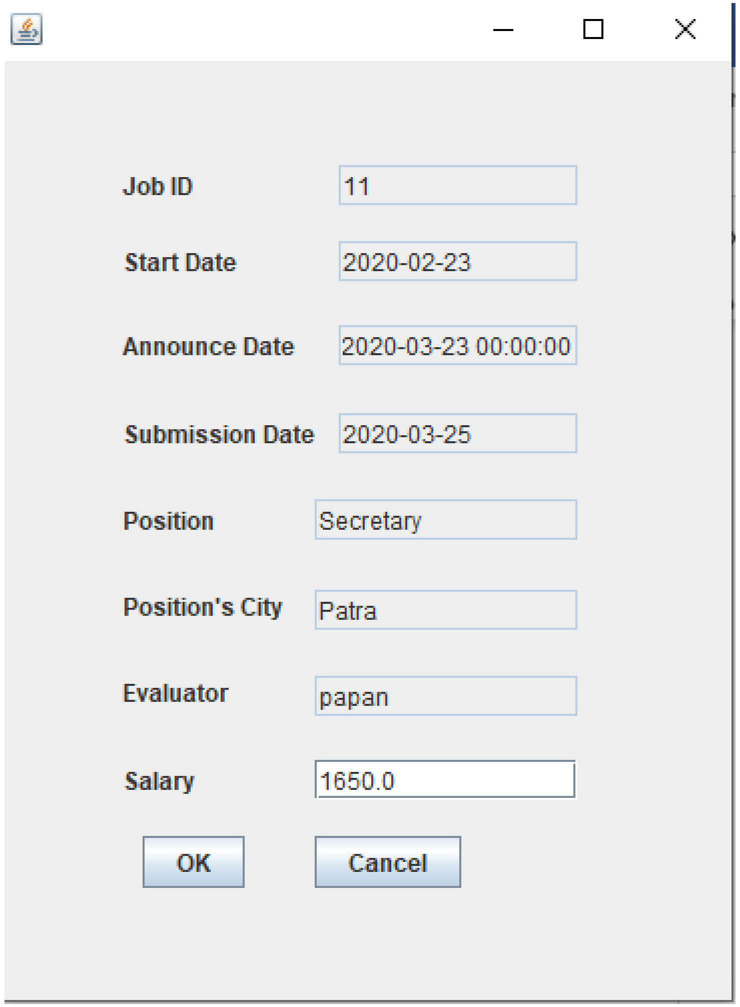
Δώσε Job ID: 11

και εμφανίζονται τα στοιχεία της Job όπως στην επόμενη εικόνα:



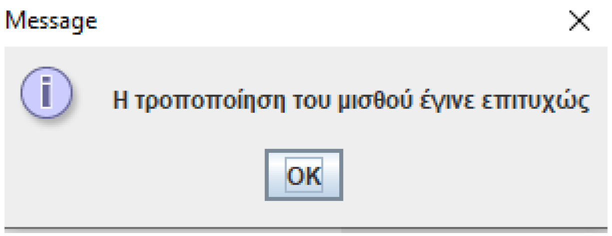
Job ID	11
Start Date	2020-02-23
Announce Date	2020-03-23 00:00:00
Submission Date	2020-03-25
Position	Secretary
Position's City	Patra
Evaluator	papan
Salary	1399.0

Παρατηρούμε ότι όλα τα πεδία είναι κλειδωμένα εκτός από το salary. Τροποποιούμε το salary όπως στην επόμενη εικόνα και πατώντας OK αυτά καταχωρούνται στη βάση δεδομένων staff evaluation και ενημερώνουν τον πίνακα job. Στην επόμενη εικόνα παραθέτουμε κάποιες ενδεικτικές αλλαγές:



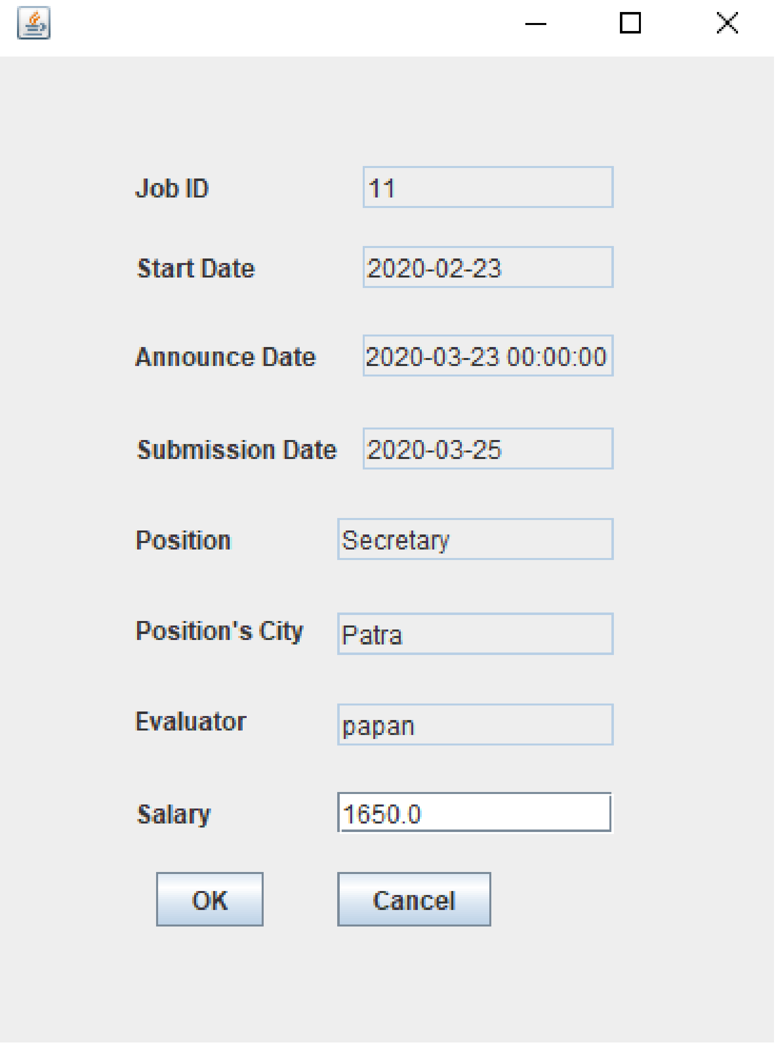
Job ID	11
Start Date	2020-02-23
Announce Date	2020-03-23 00:00:00
Submission Date	2020-03-25
Position	Secretary
Position's City	Patra
Evaluator	papan
Salary	1650.0

Πατώντας OK εμφανίζεται η εικόνα:



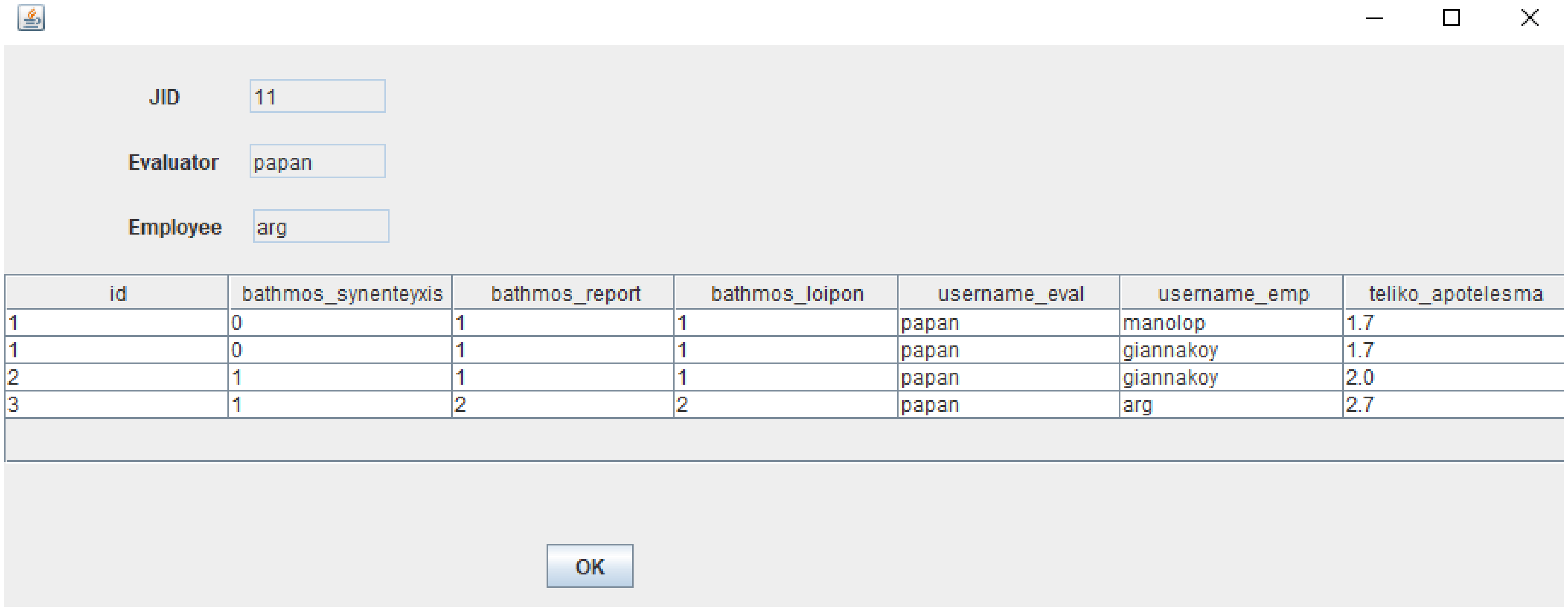
Η τροποποίηση του μισθού έγινε επιτυχώς

και μετά η εικόνα:



Job ID	11
Start Date	2020-02-23
Announce Date	2020-03-23 00:00:00
Submission Date	2020-03-25
Position	Secretary
Position's City	Patra
Evaluator	papan
Salary	1650.0

3.2.4 Λειτουργία 4 – Εμφάνιση Αποτελεσμάτων Αξιολόγησης για θέσεις της εταιρείας του που έχουν οριστικοποιηθεί οι αξιολογήσεις



JID: 11

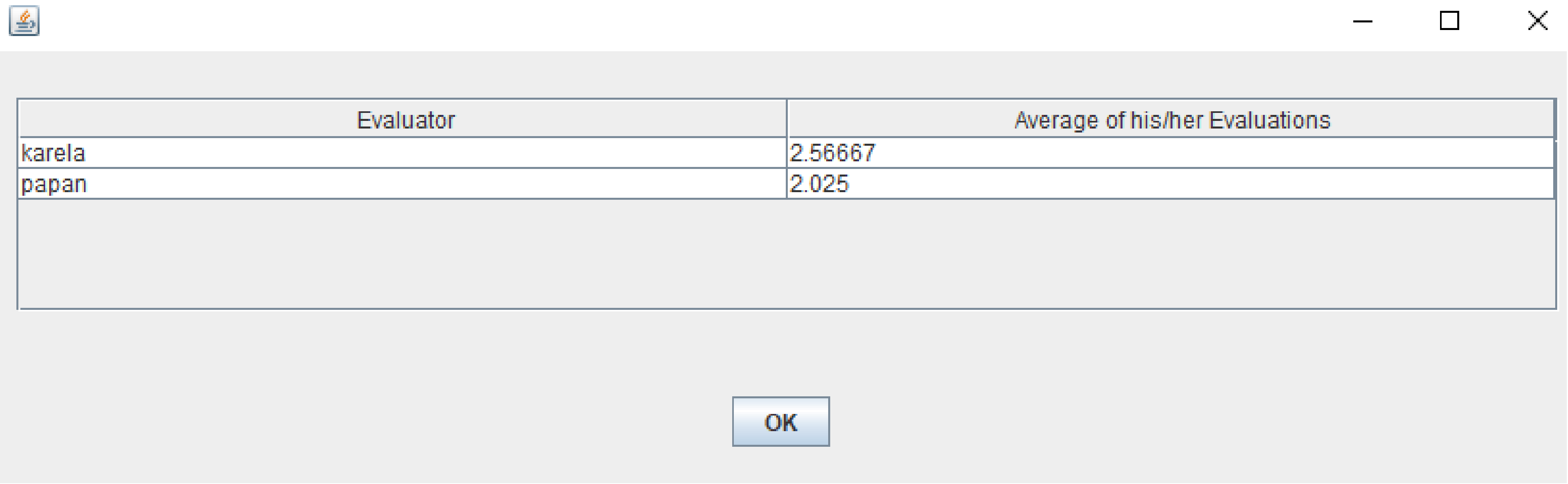
Evaluator: papan

Employee: arg

id	bathmos_synenteyxis	bathmos_report	bathmos_loipon	username_eval	username_emp	teliko_apotelesma
1	0	1	1	papan	manolop	1.7
1	0	1	1	papan	giannakoy	1.7
2	1	1	1	papan	giannakoy	2.0
3	1	2	2	papan	arg	2.7

Παρατηρούμε ότι απλά παρατίθενται όλα τα αποτελέσματα Αξιολόγησης για θέσεις με οριστικοποιημένο βαθμό δηλ. με περιεχόμενο στο πεδίο Teliko_apotelesma για την εταιρεία που είναι manager ο manager papan.

3.2.5 Λειτουργία 5 – Εμφάνιση Μέσου Όρου Βαθμού Αξιολόγησης για όλους τους Αξιολογητές της εταιρείας του

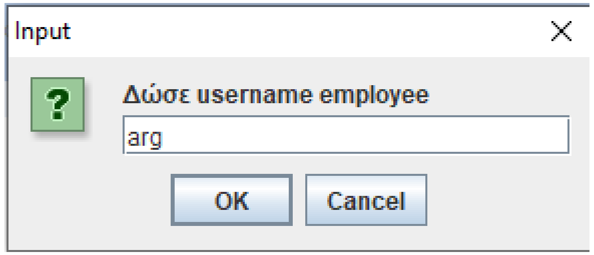


Evaluator	Average of his/her Evaluations
karela	2.56667
papan	2.025

Παρατηρούμε ότι για όλους τους evaluators της εταιρείας του manager papan (ο οποίος είναι ταυτόχρονα και evaluator) εμφανίζονται οι μέσοι όροι των αξιολογήσεων του καθενός

3.2.6 Λειτουργία 6 – Εμφάνιση και Ενημέρωση Φακέλου Εργαζόμενου

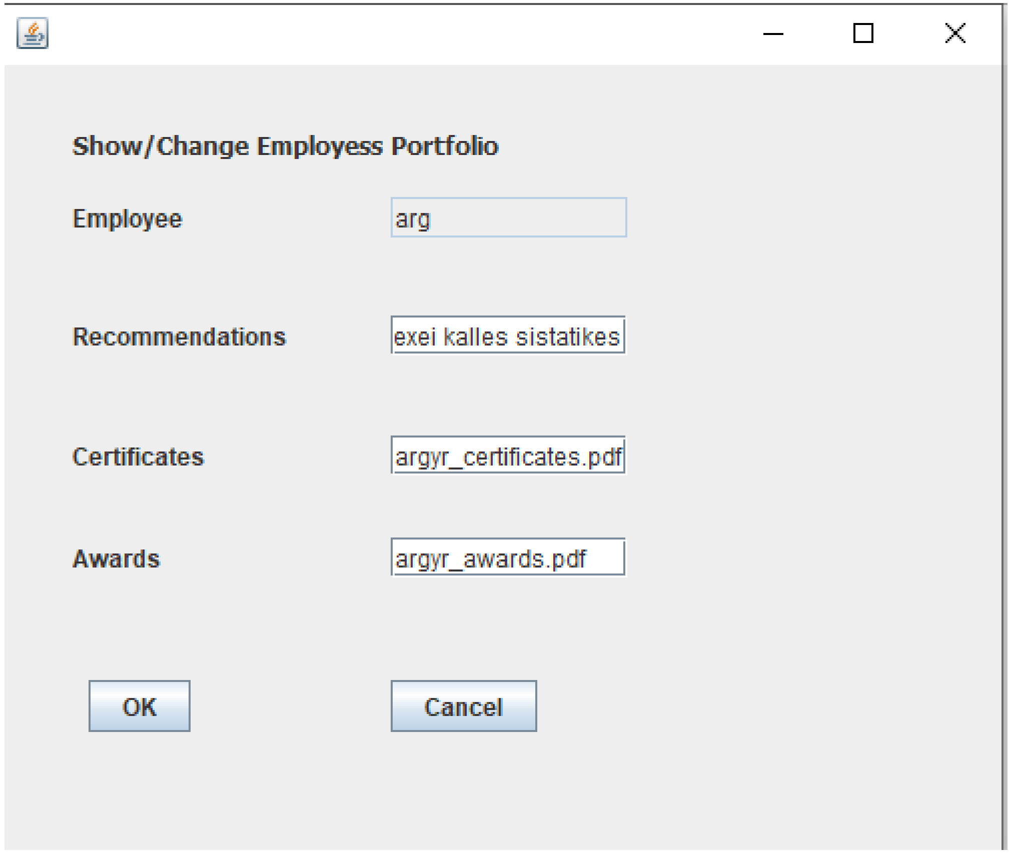
Πατώντας το πλήκτρο αυτής της λειτουργίας ανοίγει το επόμενο παράθυρο όπου συμπληρώνουμε το username ενός employee



Δώσε username employee

arg

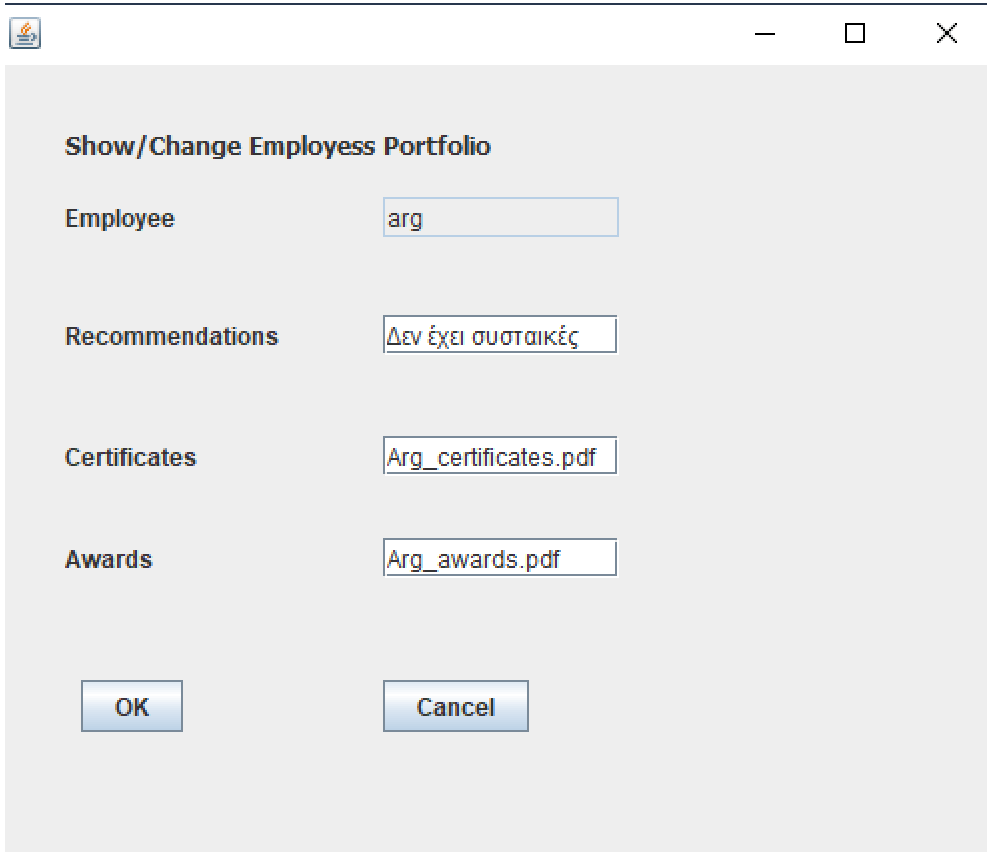
και εμφανίζεται το επόμενο παράθυρο:



Show/Change Employess Portfolio

Employee	arg
Recommendations	exei kalles sistatikes
Certificates	argyr_certificates.pdf
Awards	argyr_awards.pdf

Τροποποιούμε κάποια στοιχεία του φακέλου του συγκεκριμένου employee όπως στην επόμενη εικόνα:



Show/Change Employess Portfolio

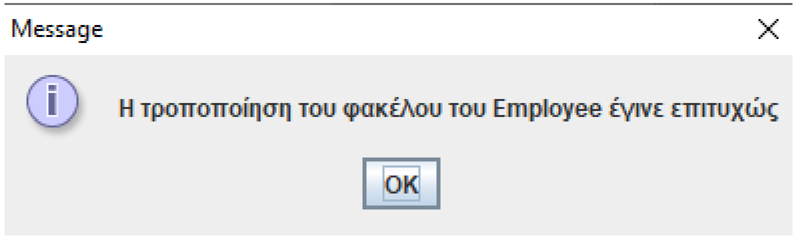
Employee: arg

Recommendations: Δεν έχει συσταικές

Certificates: Arg_certificates.pdf

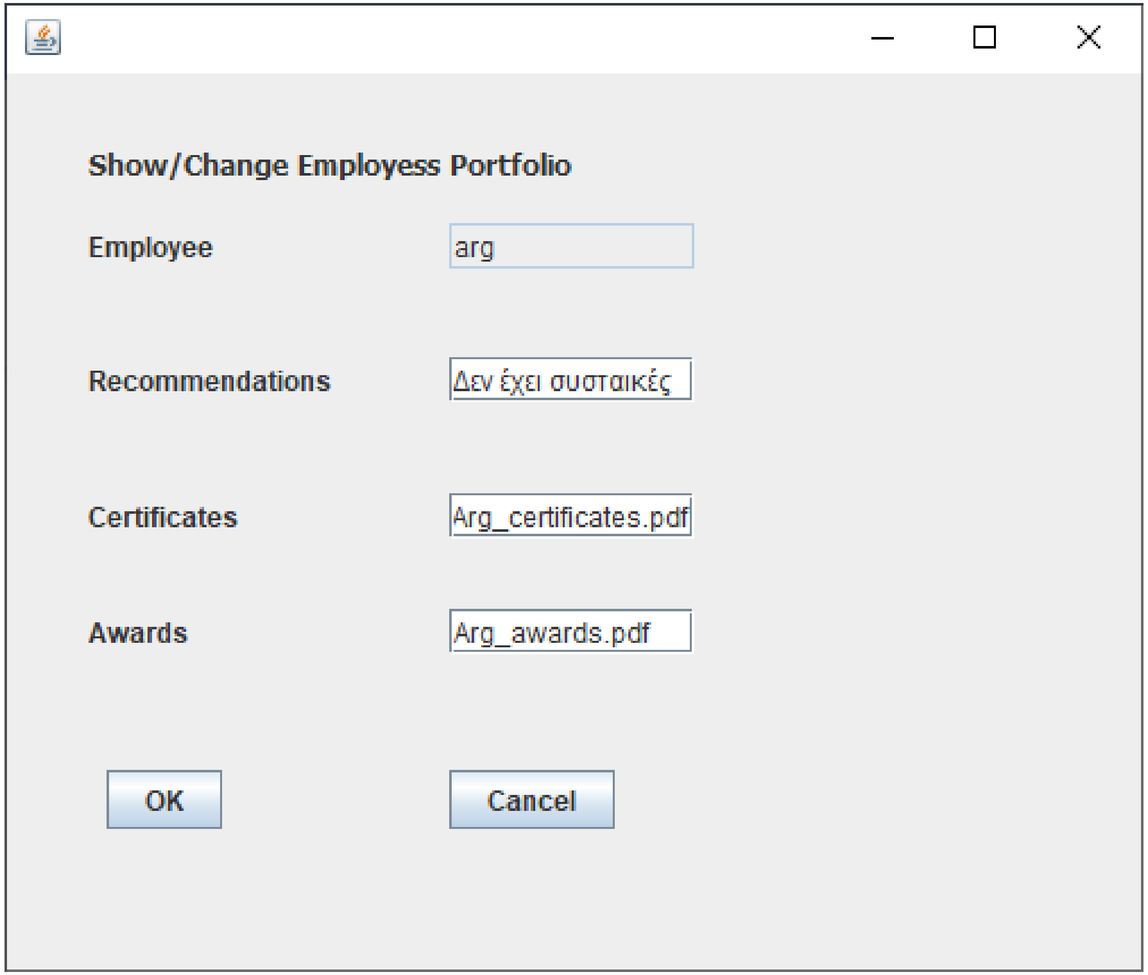
Awards: Arg_awards.pdf

και προκύπτει ότι



Message

Η τροποποίηση του φακέλου του Employee έγινε επιτυχώς και εμφανίζοντας πάλι τα στοιχεία του employee προκύπτει ότι οι αλλαγές καταχωρήθηκαν:

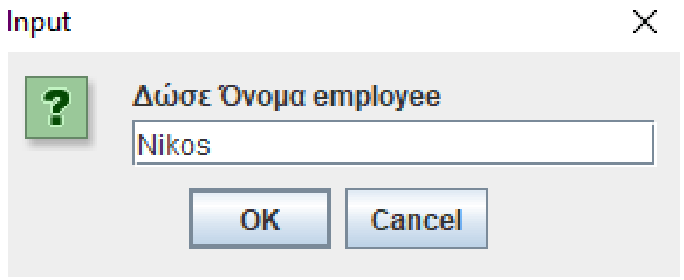


Show/Change Employess Portfolio

Employee	arg
Recommendations	Δεν έχει συσταικές
Certificates	Arg_certificates.pdf
Awards	Arg_awards.pdf

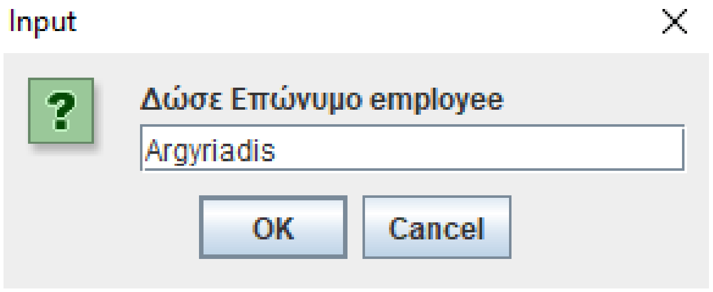
3.2.7 Λειτουργία 7– Εμφάνιση (α) όλων των αιτήσεων συγκεκριμένου υπαλλήλου (β) οι τελικές ολοκληρωμένες αξιολογήσεις του και (γ) το όνομα και επώνυμο του αντίστοιχου αξιολογητή

Πατώντας το πλήκτρο αυτής της λειτουργίας ανοίγει το επόμενο παράθυρο όπου συμπληρώνουμε το ονοματεπώνυμο ενός employee


Input

Δώσε Όνομα employee

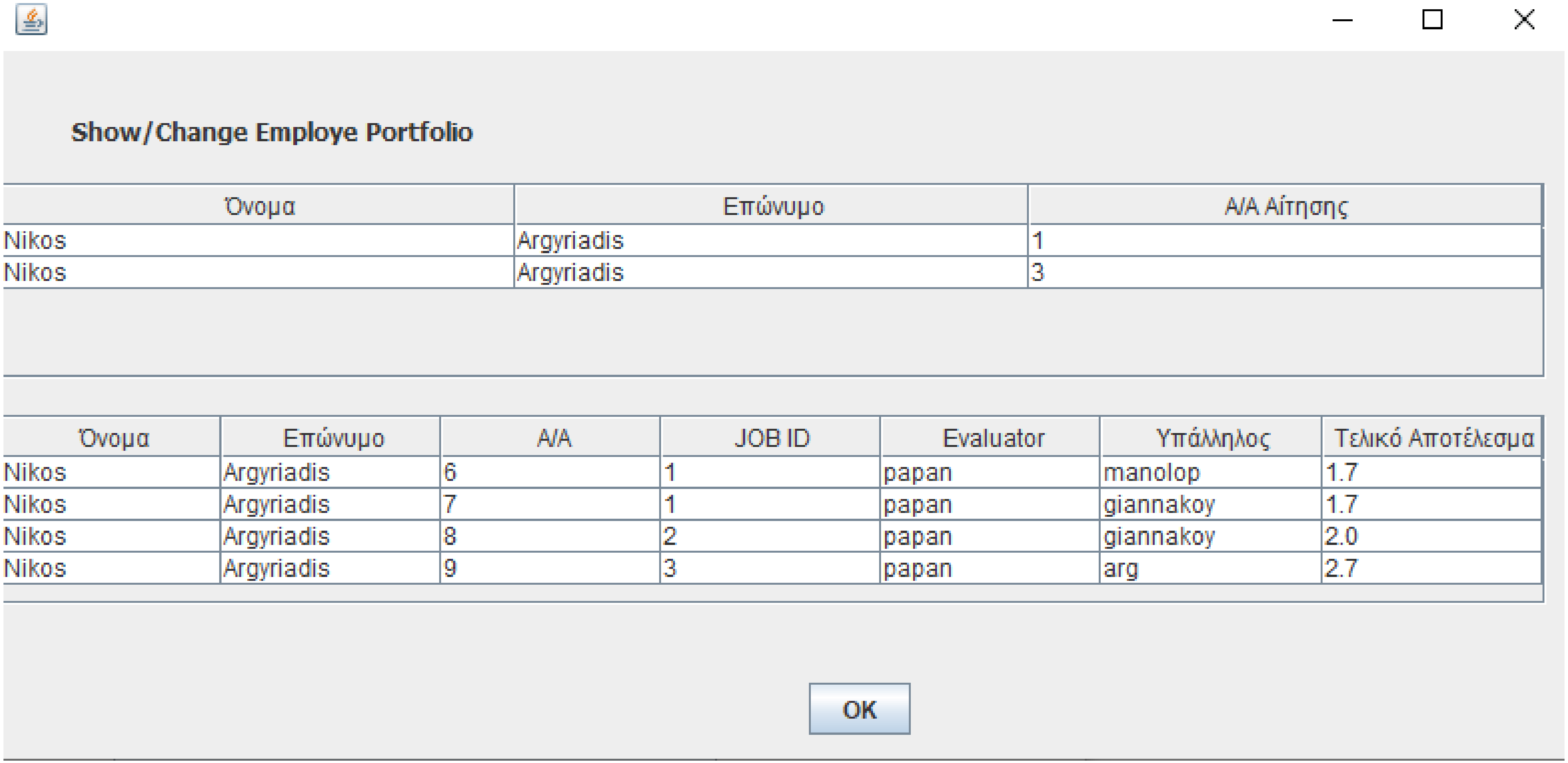
Nikos


Input

Δώσε Επώνυμο employee

Argyriadis

και μετά εμφανίζονται όλες οι αιτήσεις του, οι ολοκληρωμένες αξιολογήσεις του και το username του αντίστοιχου αξιολογητή (evaluator)



Show/Change Employe Portfolio

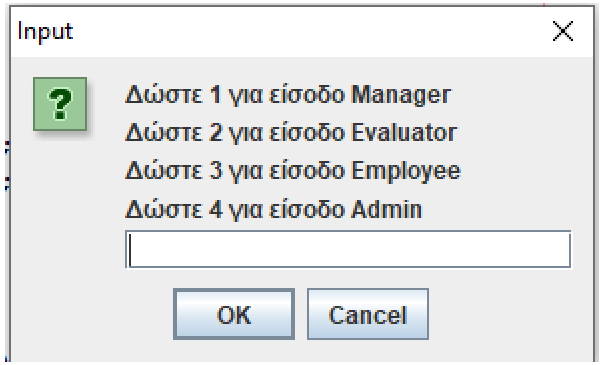
Όνομα	Επώνυμο	Α/Α Αίτησης
Nikos	Argyriadis	1
Nikos	Argyriadis	3

Όνομα	Επώνυμο	Α/Α	JOB ID	Evaluator	Υπάλληλος	Τελικό Αποτέλεσμα
Nikos	Argyriadis	6	1	papan	manolop	1.7
Nikos	Argyriadis	7	1	papan	giannakoy	1.7
Nikos	Argyriadis	8	2	papan	giannakoy	2.0
Nikos	Argyriadis	9	3	papan	arg	2.7

3.3 Λειτουργίες Αξιολογητή (Evaluator)

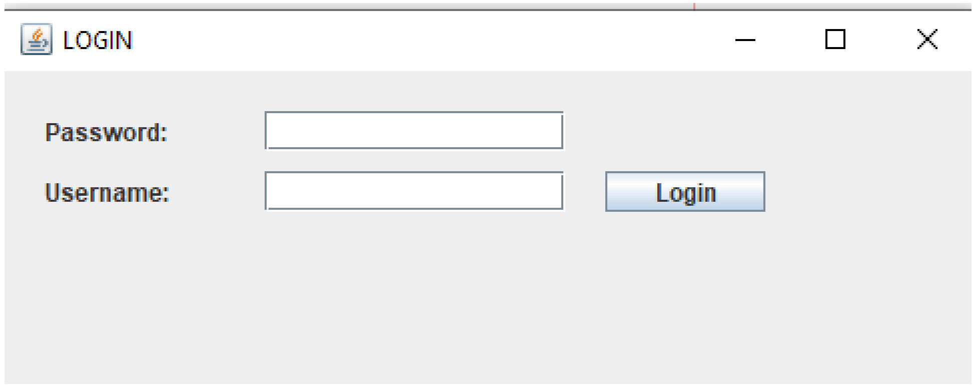
Στα παραδείγματα που παραθέτουμε χρησιμοποιούμε ως Evaluator το χρήστη με username papan ο οποίος ταυτόχρονα είναι και evaluator.

Αρχικά γίνεται είσοδος του Evaluator εκτελώντας το αρχείο login.java. Εμφανίζεται το ακόλουθο παράθυρο:

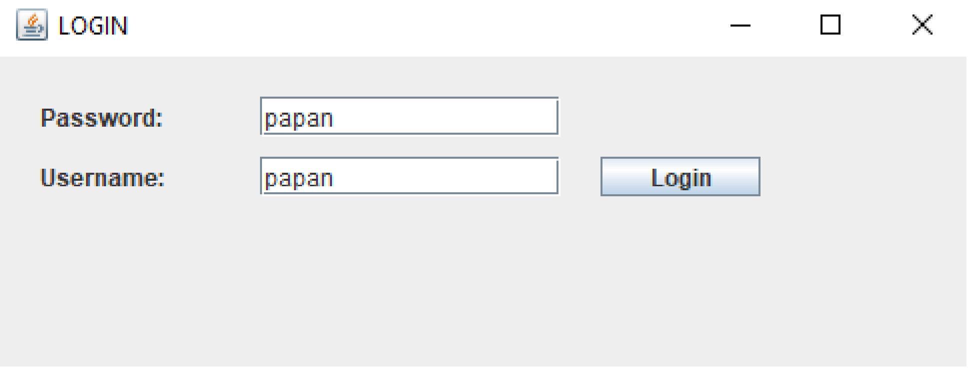


Δώστε 1 για είσοδο Manager
Δώστε 2 για είσοδο Evaluator
Δώστε 3 για είσοδο Employee
Δώστε 4 για είσοδο Admin

Ακολούθως εμφανίζεται το παράθυρο LOGIN για να εισάγει ο Evaluator τα στοιχεία του:



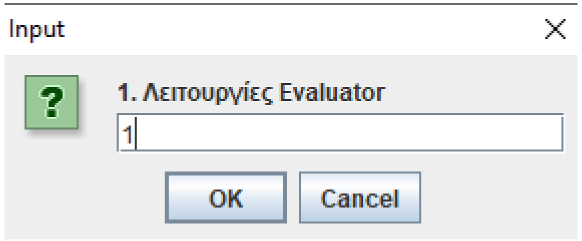
Συμπληρώνει τα στοιχεία του:



Password: papan

Username: papan

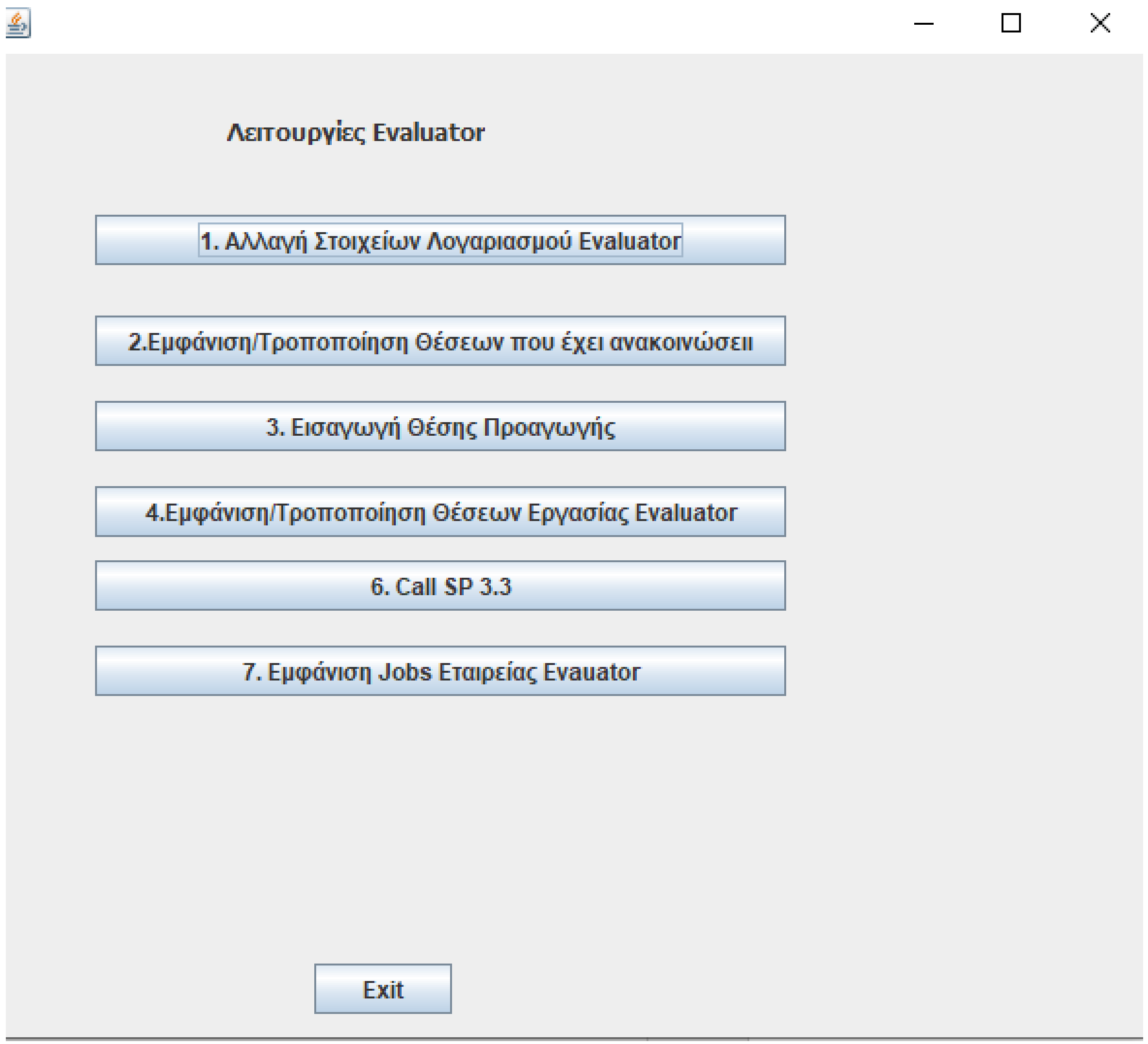
Εμφανίζεται το επόμενο παράθυρο στο οποίο εισάγει το 1:



1. Λειτουργίες Evaluator

1

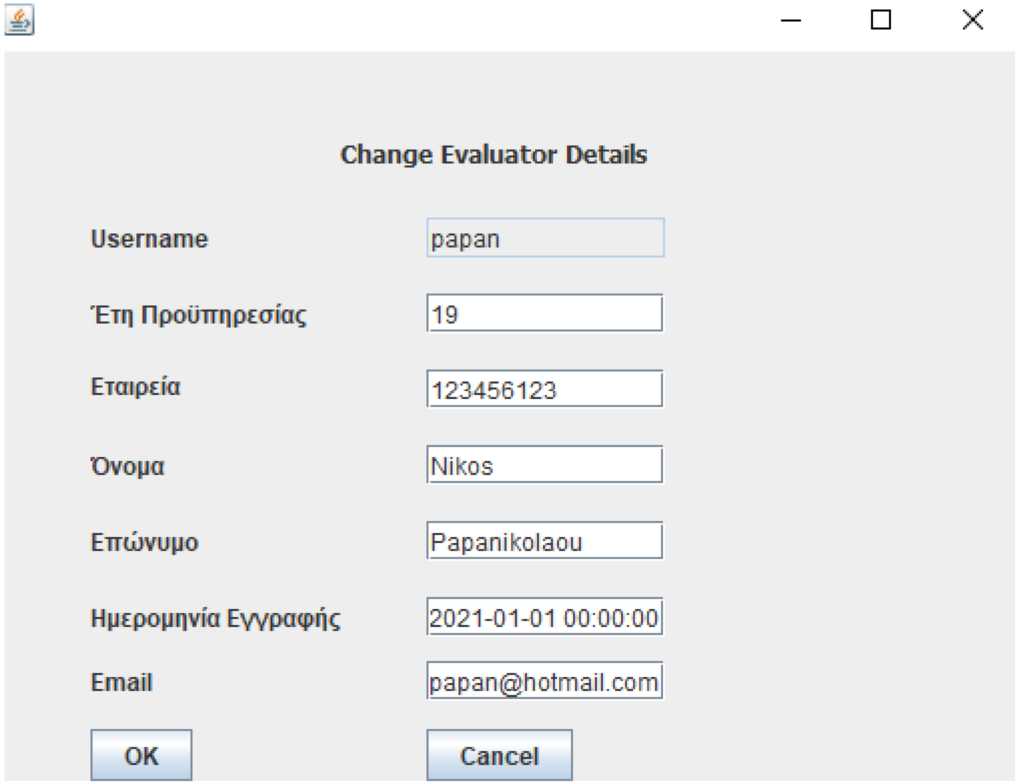
Μετά εμφανίζεται το μενού με τις λειτουργίες του:



Λειτουργίες Evaluator
1. Αλλαγή Στοιχείων Λογαριασμού Evaluator
2.Εμφάνιση/Τροποποίηση Θέσεων που έχει ανακοινώσειι
3. Εισαγωγή Θέσης Προαγωγής
4.Εμφάνιση/Τροποποίηση Θέσεων Εργασίας Evaluator
6. Call SP 3.3
7. Εμφάνιση Jobs Εταιρείας Evauator
Exit

3.3.1 Λειτουργία 1 – Εμφάνιση και Επεξεργασία των Στοιχείων του Λογαριασμού του

Πατώντας το πλήκτρο Αλλαγή Στοιχείων Λογαριασμού Evaluator εμφανίζεται το ακόλουθο παράθυρο:

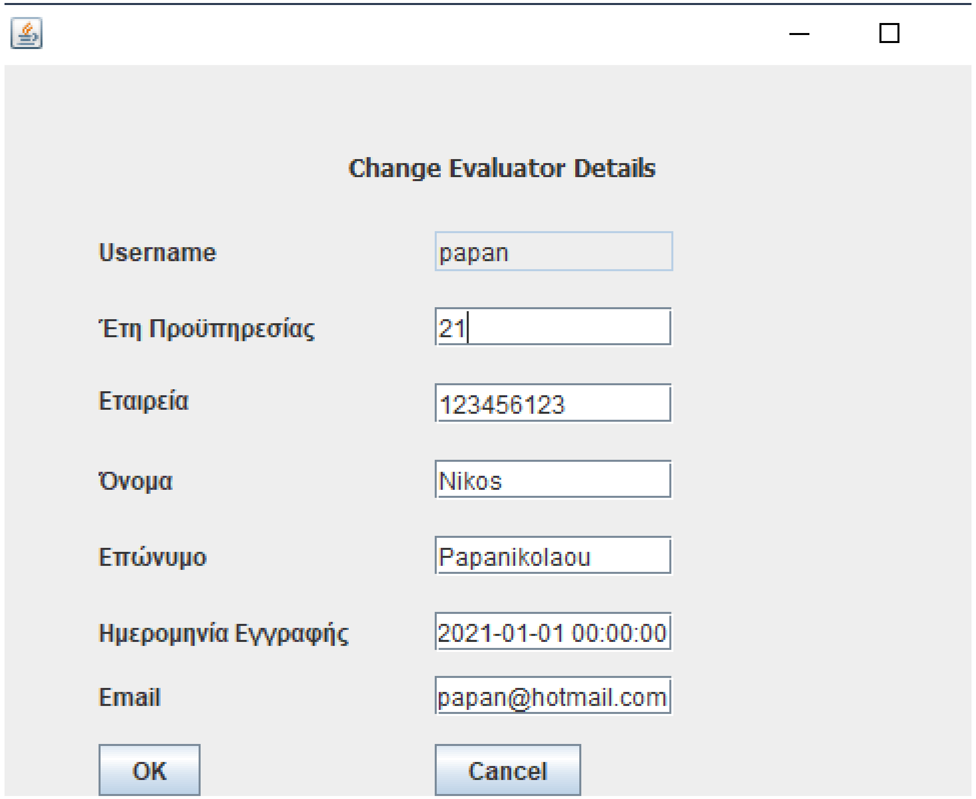


Change Evaluator Details

Username	papan
Έτη Προϋπηρεσίας	19
Εταιρεία	123456123
Όνομα	Nikos
Επώνυμο	Papanikolaou
Ημερομηνία Εγγραφής	2021-01-01 00:00:00
Email	papan@hotmail.com

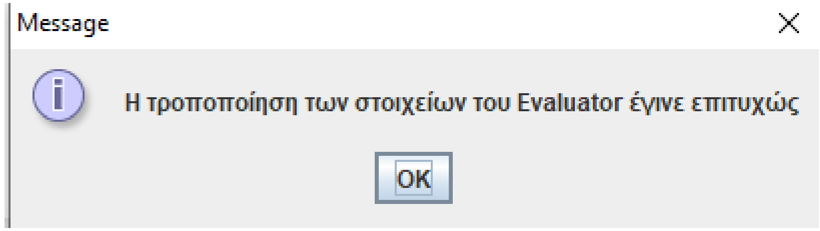
Αλλάζουμε την ηλικία από 19 σε 21 και πατάμε OK

Η επόμενη εικόνα δείχνει ότι η αλλαγή αυτή καταχωρήθηκε επιτυχώς.



Change Evaluator Details

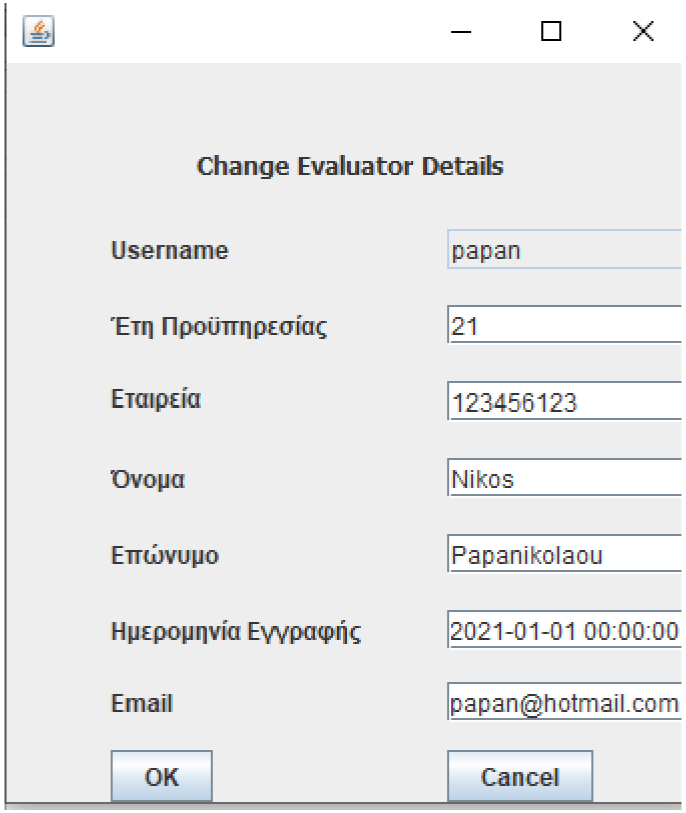
Username	papan
Έτη Προϋπηρεσίας	21
Εταιρεία	123456123
Όνομα	Nikos
Επώνυμο	Papanikolaou
Ημερομηνία Εγγραφής	2021-01-01 00:00:00
Email	papan@hotmail.com



Message

Η τροποποίηση των στοιχείων του Evaluator έγινε επιτυχώς

Όταν μετά ο evaluator επιλέξει πάλι την ίδια λειτουργία φαίνεται ότι η τροποποίηση έχει γίνει επιτυχώς από την επόμενη εικόνα

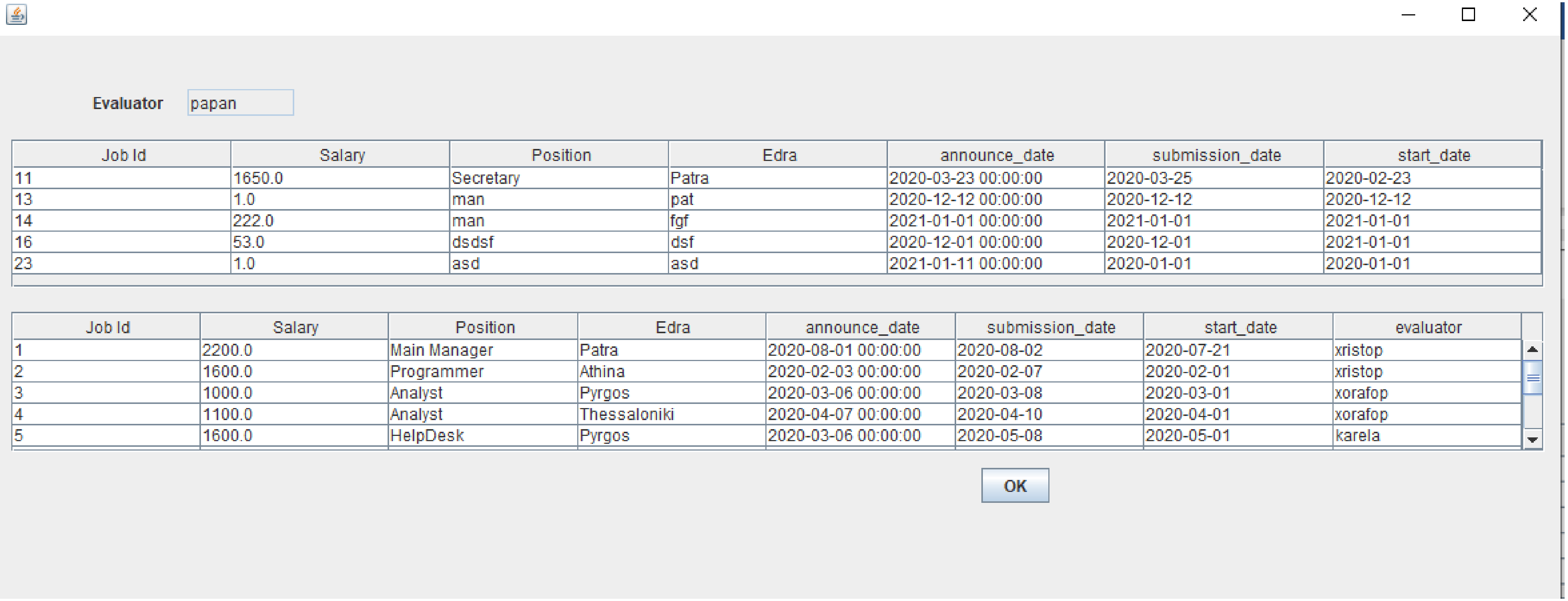


Change Evaluator Details

Username	papan
Έτη Προϋπηρεσίας	21
Εταιρεία	123456123
Όνομα	Nikos
Επώνυμο	Papanikolaou
Ημερομηνία Εγγραφής	2021-01-01 00:00:00
Email	papan@hotmail.com

3.3.2 Λειτουργία 2 – Εμφάνιση Θέσεων Προαγωγής Εταιρείας/Επεξεργασία των Δικών του

Αν στο βασικό μενού με τις Λειτουργίες του Evaluator επιλέξουμε τη 2η λειτουργία με τίτλο ‘Εμφάνιση/Τροποποίηση Θέσεων που έχει ανακοινώσει τότε εμφανίζεται η ακόλουθη εικόνα για τον evaluator που έχει κάνει login



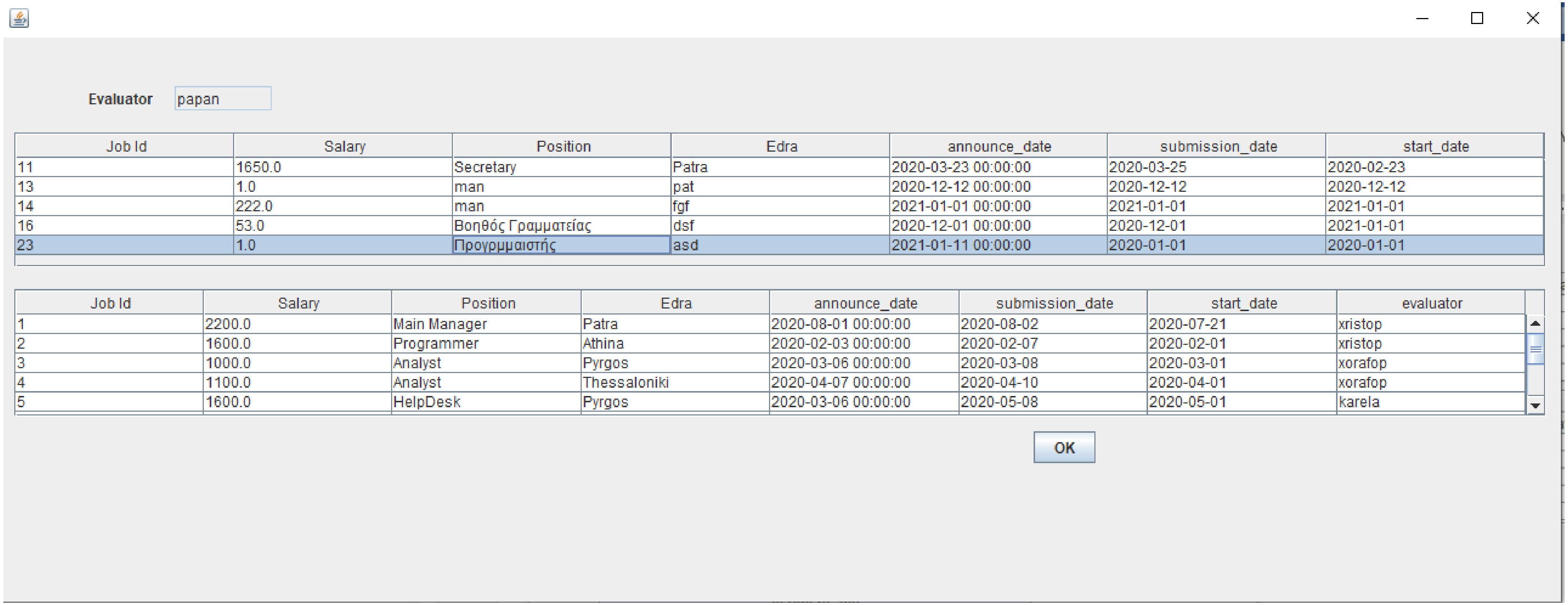
Evaluator papan

Job Id	Salary	Position	Edra	announce_date	submission_date	start_date
11	1650.0	Secretary	Patra	2020-03-23 00:00:00	2020-03-25	2020-02-23
13	1.0	man	pat	2020-12-12 00:00:00	2020-12-12	2020-12-12
14	222.0	man	fgf	2021-01-01 00:00:00	2021-01-01	2021-01-01
16	53.0	dsdsf	dsf	2020-12-01 00:00:00	2020-12-01	2021-01-01
23	1.0	asd	asd	2021-01-11 00:00:00	2020-01-01	2020-01-01

Job Id	Salary	Position	Edra	announce_date	submission_date	start_date	evaluator
1	2200.0	Main Manager	Patra	2020-08-01 00:00:00	2020-08-02	2020-07-21	xristop
2	1600.0	Programmer	Athina	2020-02-03 00:00:00	2020-02-07	2020-02-01	xristop
3	1000.0	Analyst	Pyrgos	2020-03-06 00:00:00	2020-03-08	2020-03-01	xorafop
4	1100.0	Analyst	Thessaloniki	2020-04-07 00:00:00	2020-04-10	2020-04-01	xorafop
5	1600.0	HelpDesk	Pyrgos	2020-03-06 00:00:00	2020-05-08	2020-05-01	karela

Στην 1η λίστα εμφανίζεται η λίστα των θέσεων που έχει ανακοινώσει ο ίδιος και οι οποίες είναι επεξεργάσιμες/τροποποιήσιμε από τον evaluator papan που έχει κάνει login.
Στην 2η λίστα εμφανίζεται η λίστα των θέσεων που έχουν ανακοινώσει άλλοι evaluator της εταιρείας του και οι οποίες δεν είναι επεξεργάσιμες/τροποποιήσιμε από τον evaluator papan που έχει κάνει login παρά μόνο ορατές σε μια λίστα.

Στην επόμενη εικόνα φαίνονται μερικές αλλαγές που κάνει ο evaluator στις δικές του θέσεις προαγωγής



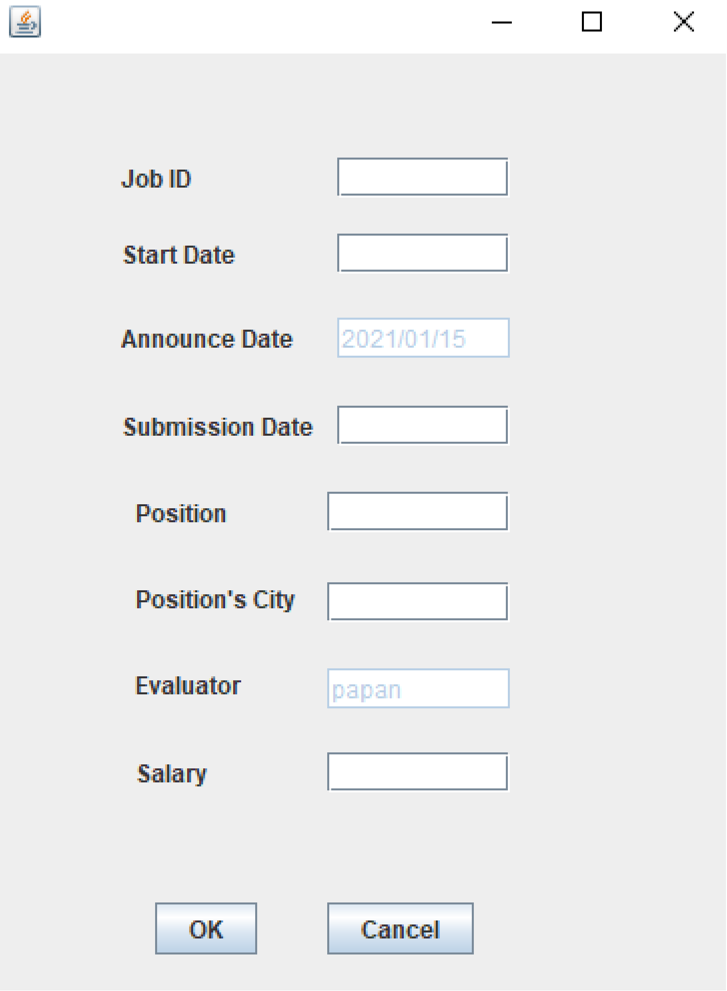
Evaluator papan

Job Id	Salary	Position	Edra	announce_date	submission_date	start_date
11	1650.0	Secretary	Patra	2020-03-23 00:00:00	2020-03-25	2020-02-23
13	1.0	man	pat	2020-12-12 00:00:00	2020-12-12	2020-12-12
14	222.0	man	fgf	2021-01-01 00:00:00	2021-01-01	2021-01-01
16	53.0	Βοηθός Γραμματείας	dsf	2020-12-01 00:00:00	2020-12-01	2021-01-01
23	1.0	Προγρμμαιστής	asd	2021-01-11 00:00:00	2020-01-01	2020-01-01

Job Id	Salary	Position	Edra	announce_date	submission_date	start_date	evaluator
1	2200.0	Main Manager	Patra	2020-08-01 00:00:00	2020-08-02	2020-07-21	xristop
2	1600.0	Programmer	Athina	2020-02-03 00:00:00	2020-02-07	2020-02-01	xristop
3	1000.0	Analyst	Pyrgos	2020-03-06 00:00:00	2020-03-08	2020-03-01	xorafop
4	1100.0	Analyst	Thessaloniki	2020-04-07 00:00:00	2020-04-10	2020-04-01	xorafop
5	1600.0	HelpDesk	Pyrgos	2020-03-06 00:00:00	2020-05-08	2020-05-01	karela

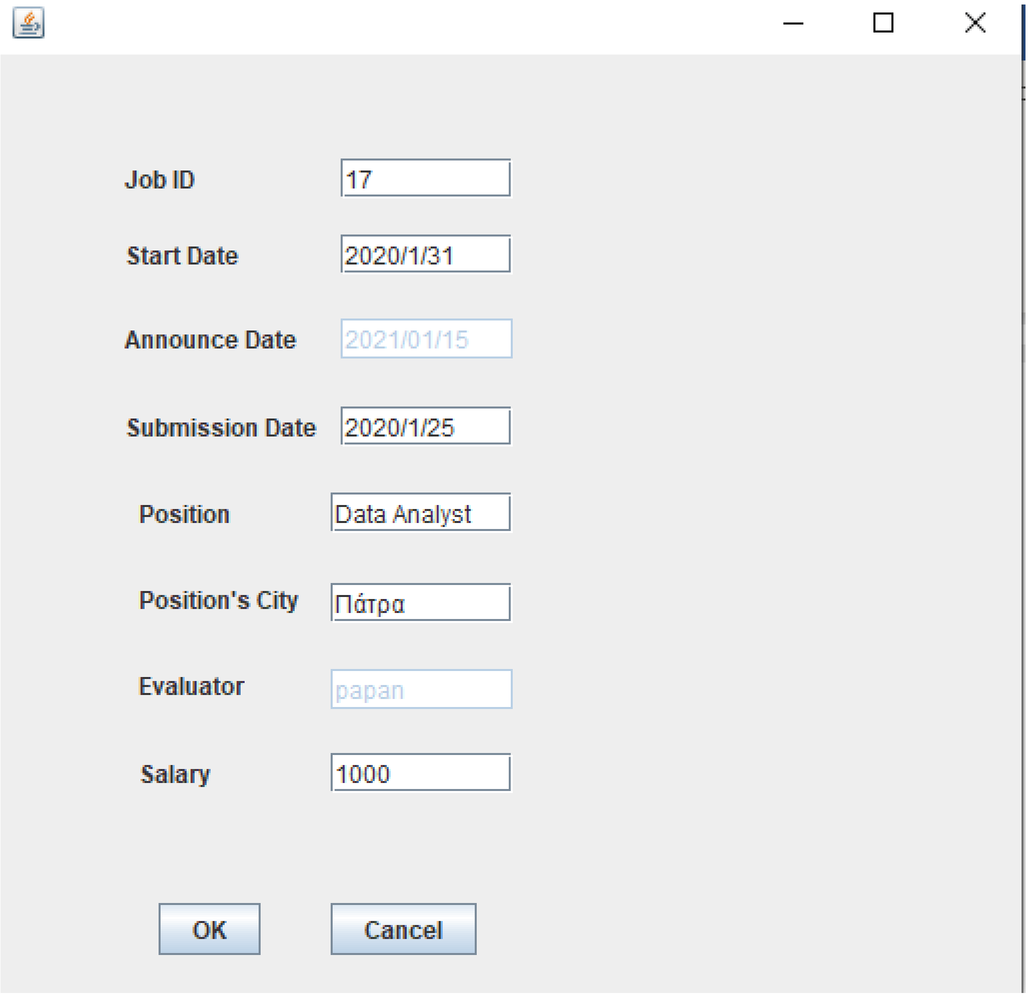
3.3.3 Λειτουργία 3 – Προσθήκη Νέας Θέσης Προαγωγής από Evaluator

Αν στο βασικό μενού με τις Λειτουργίες του Evaluator επιλέξουμε τη 3η λειτουργία με τίτλο ‘Εισαγωγή Θέσης Προαγωγής’ τότε εμφανίζεται η ακόλουθη εικόνα:



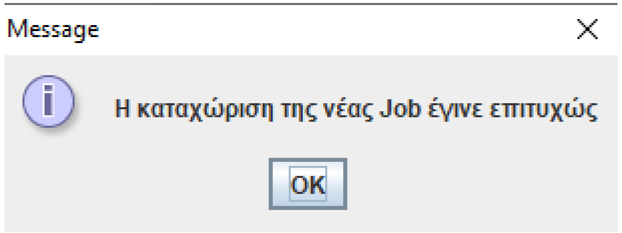
Job ID	
Start Date	
Announce Date	2021/01/15
Submission Date	
Position	
Position's City	
Evaluator	papan
Salary	

O evaluator συμπληρώνει τα στοιχεία της νέας θέσης όπως στην επόμενη εικόνα:

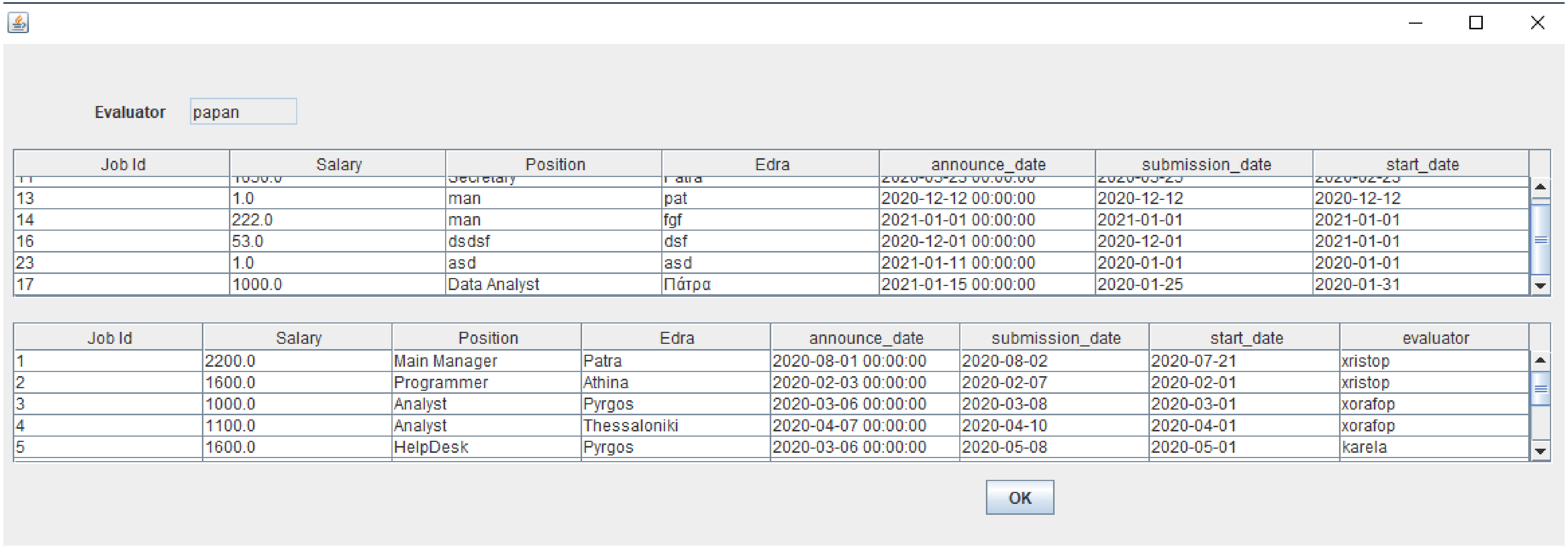


Job ID	17
Start Date	2020/1/31
Announce Date	2021/01/15
Submission Date	2020/1/25
Position	Data Analyst
Position's City	Πάτρα
Evaluator	papan
Salary	1000

Όταν πατήσει OK η θέση καταχωρείται εμφανίζεται η ακόλουθη εικόνα



Η καταχώριση της νέας Job έγινε επιτυχώς και η επιβεβαίωση γιαυτό είναι ότι αν επιλέξει και πάλι την 2η επιλογή του μενού του ‘Εμφάνιση/Τροποποίηση των θέσεων που έχει ανακοινώσει εμφανίζεται η επόμενη εικόνα



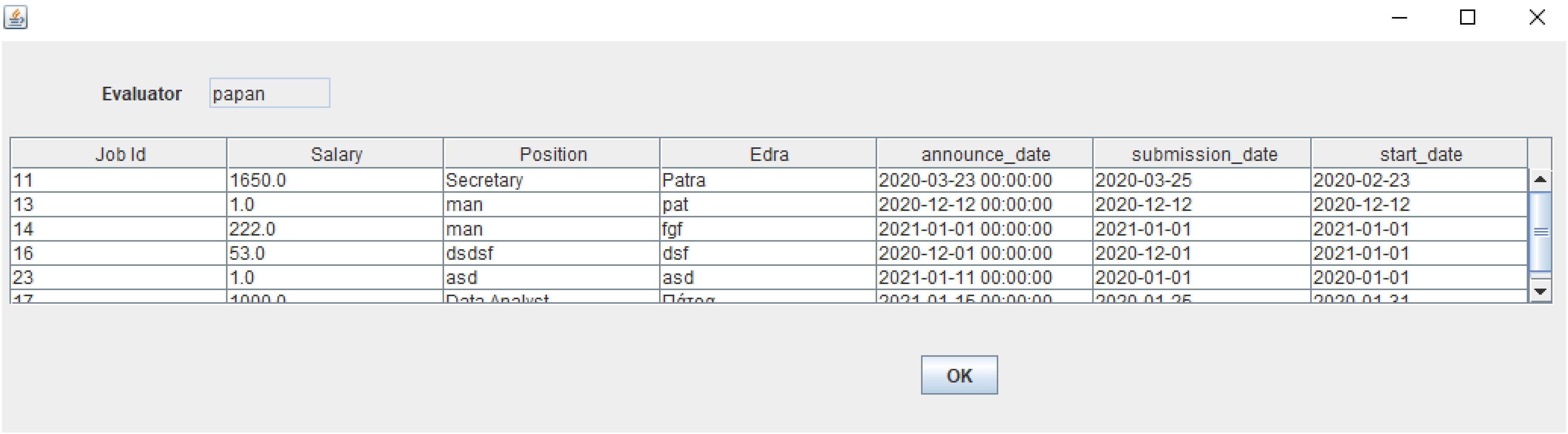
Evaluator papan

Job Id	Salary	Position	Edra	announce_date	submission_date	start_date
11	1050.0	Secretary	Patra	2020-03-25 00:00:00	2020-03-25	2020-02-25
13	1.0	man	pat	2020-12-12 00:00:00	2020-12-12	2020-12-12
14	222.0	man	fgf	2021-01-01 00:00:00	2021-01-01	2021-01-01
16	53.0	dsdsf	dsf	2020-12-01 00:00:00	2020-12-01	2021-01-01
23	1.0	asd	asd	2021-01-11 00:00:00	2020-01-01	2020-01-01
17	1000.0	Data Analyst	Πάτρα	2021-01-15 00:00:00	2020-01-25	2020-01-31

Job Id	Salary	Position	Edra	announce_date	submission_date	start_date	evaluator
1	2200.0	Main Manager	Patra	2020-08-01 00:00:00	2020-08-02	2020-07-21	xristop
2	1600.0	Programmer	Athina	2020-02-03 00:00:00	2020-02-07	2020-02-01	xristop
3	1000.0	Analyst	Pyrgos	2020-03-06 00:00:00	2020-03-08	2020-03-01	xorafop
4	1100.0	Analyst	Thessaloniki	2020-04-07 00:00:00	2020-04-10	2020-04-01	xorafop
5	1600.0	HelpDesk	Pyrgos	2020-03-06 00:00:00	2020-05-08	2020-05-01	karela

3.3.4 Λειτουργία 4 – Εμφάνιση Θέσεων/Εμφάνιση Εργασίας από Evaluator

Αν στο βασικό μενού με τις Λειτουργίες του Evaluator επιλέξουμε τη 4η λειτουργία με τίτλο ‘Εμφάνιση/Τροποποίηση Θέσεων Εργασίας Evaluator’ τότε εμφανίζεται η ακόλουθη εικόνα:



Evaluator papan

Job Id	Salary	Position	Edra	announce_date	submission_date	start_date
11	1650.0	Secretary	Patra	2020-03-23 00:00:00	2020-03-25	2020-02-23
13	1.0	man	pat	2020-12-12 00:00:00	2020-12-12	2020-12-12
14	222.0	man	fgf	2021-01-01 00:00:00	2021-01-01	2021-01-01
16	53.0	dsdsf	dsf	2020-12-01 00:00:00	2020-12-01	2021-01-01
23	1.0	asd	asd	2021-01-11 00:00:00	2020-01-01	2020-01-01
17	1000.0	Data Analyst	Πάτρα	2021-01-15 00:00:00	2020-01-25	2020-01-31

Κάνοντας διπλό κλικ σε μια θέση μπορεί να τροποποιήσει τα στοιχεία της

3.3.5 Λειτουργία 6 – Επιλογή Θέσης Εργασίας και κλήση stored procedure 3.3.

Για την εκτέλεση αυτής της λειτουργίας έχουμε ξαναγράψει την **stored procedure 3.3 ελαφρώς τροποποιημένη ως ακολούθως:**

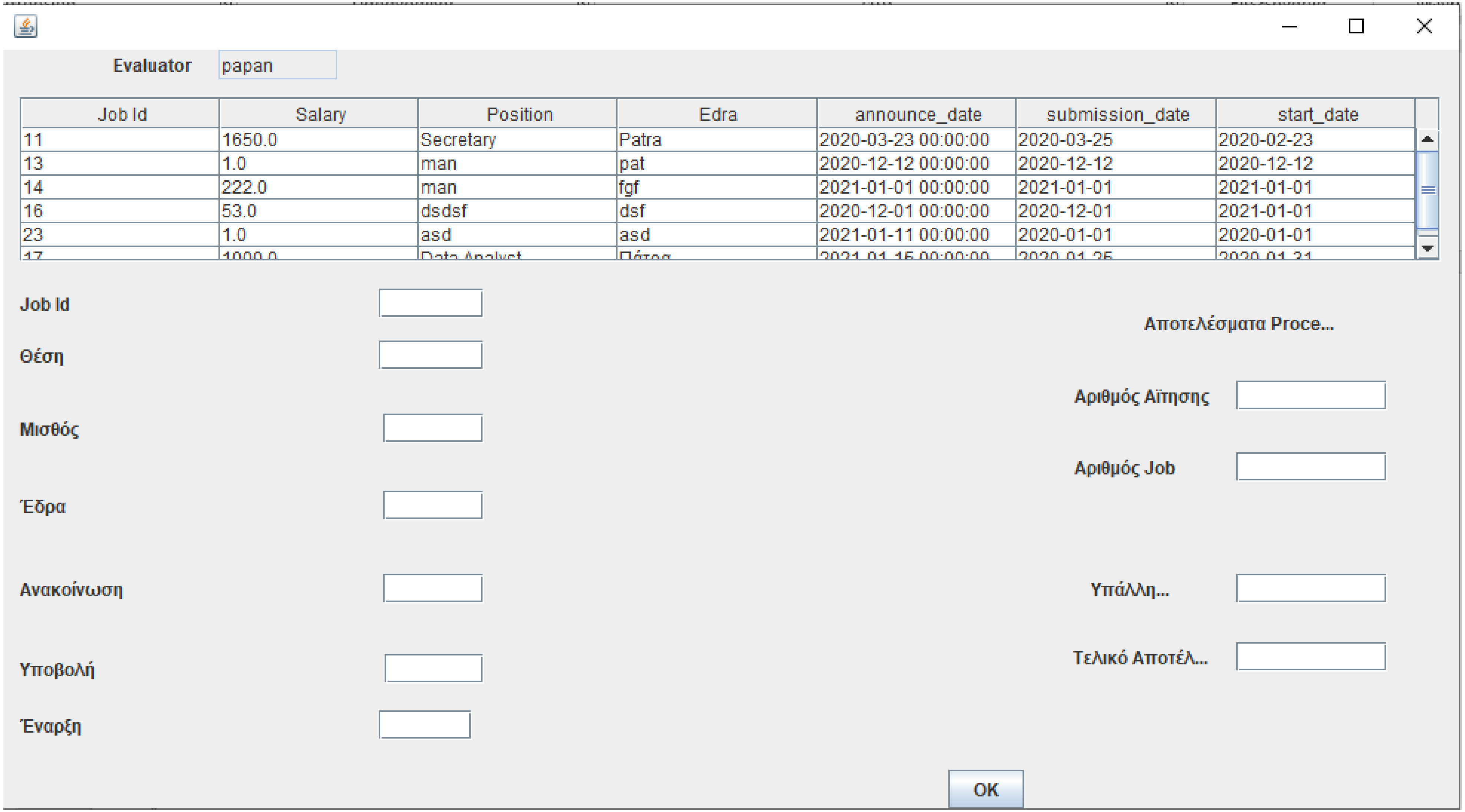
```
drop procedure if exists proc3b;
delimiter $
create procedure proc3b(IN jbdid int,OUT ar_aitisis int,OUT jid int,OUT empl varchar(12),OUT eval varchar(12),OUT ta float(3,1))
begin
  declare nf int;

  -- declare ar_aitisis int;
  -- declare jid int;
  -- declare empl varchar(12);
  -- declare eval varchar(12);
```

```
   declare bs int;
   declare br int;
   declare bl int;
   declare uname varchar(25);
   declare usname varchar(35);
   declare final_grade float(3,1);
   -- declare ta float(3,1);
   declare counter int;

   declare bcursor cursor for select distinct
evaluation_process.aa,evaluation_process.job_id,evaluation_process.username_emp,evaluation_process.username_eval,evaluation_process.teliko_apotelesma
   from user,employee,evaluation_process,evaluator,requestsevaluation
   where user.username=employee.username and employee.username=requestsevaluation.empl_username and
    employee.username=evaluation_process.username_emp and evaluation_process.job_id=jbdid;

    declare bcursor2 cursor for select distinct
evaluation_process.aa,evaluation_process.job_id,evaluation_process.username_emp,evaluation_process.username_eval,evaluation_process.teliko_apotelesma
                                   from user,employee,evaluation_process,evaluator,requestsevaluation
                                   where user.username=employee.username and employee.username=requestsevaluation.empl_username and
                                   employee.username=evaluation_process.username_emp and
                                   evaluation_process.job_id=jbdid
                                   order by evaluation_process.teliko_apotelesma desc;



           declare bcursor3 cursor for select distinct
evaluation_process.aa,evaluation_process.job_id,evaluation_process.username_emp,evaluation_process.username_eval,evaluation_process.teliko_apotelesma
                                   from user,employee,evaluation_process,evaluator,requestsevaluation
                                   where user.username=employee.username and employee.username=requestsevaluation.empl_username and
```

```
                                        employee.username=evaluation_process.username_emp and   evaluation_process.job_id=jbdid;



 declare continue handler for not found set nf=1;



set nf=0;
set counter=0;

open bcursor;

repeat
      fetch bcursor into ar_aitisis,jid,empl,eval,ta;

      if (nf=0) then
              if (ta is null) then
                  set counter=counter+1;
              end if;
      end if;
  until (nf=1)
end repeat;

      select 'Counter =',counter;

      if (counter=0) then
              select 'ORISTIKOPOIHMENOI PINAKES';

              set nf=0;
              open bcursor2;

              repeat
                  fetch bcursor2 into ar_aitisis,jid,empl,eval,ta;

                  if (nf=0) then
                      if (ta is not null) then
                          select ar_aitisis,jid,empl,eval,ta;
                      end if;
                  end if;

              until (nf=1)
              end repeat;
      end if;



  if (counter<>0) then
              select 'AXIOLOGHSH SE EXELIXI. EKKREMOUN AKOMA: ',counter;

              set nf=0;
              open bcursor3;

              repeat
                  fetch bcursor3 into ar_aitisis,jid,empl,eval,ta;
                  select ar_aitisis,jid,empl,eval,ta;
                  select '---------------------------------';
                  if (nf=0) then
                      if (ta is not null) then
```

```
                                        select ar_aitisis,jid,empl,eval,ta;
                                end if;
                        end if;

                until (nf=1)
                end repeat;
        end if;

end$
delimiter ;
```

Στη συνέχεια πατώντας το πλήκτρο Calls SP 3.3 και εμφανίζεται η ακόλουθη εικόνα:



Evaluator papan

Job Id	Salary	Position	Edra	announce_date	submission_date	start_date
11	1650.0	Secretary	Patra	2020-03-23 00:00:00	2020-03-25	2020-02-23
13	1.0	man	pat	2020-12-12 00:00:00	2020-12-12	2020-12-12
14	222.0	man	fgf	2021-01-01 00:00:00	2021-01-01	2021-01-01
16	53.0	dsdsf	dsf	2020-12-01 00:00:00	2020-12-01	2021-01-01
23	1.0	asd	asd	2021-01-11 00:00:00	2020-01-01	2020-01-01
17	1000.0	Data Analyst	Πάτρα	2021-01-15 00:00:00	2020-01-25	2020-01-31

Job Id

Θέση

Μισθός

Έδρα

Ανακοίνωση

Υποβολή

Έναρξη

Αποτελέσματα Proce...

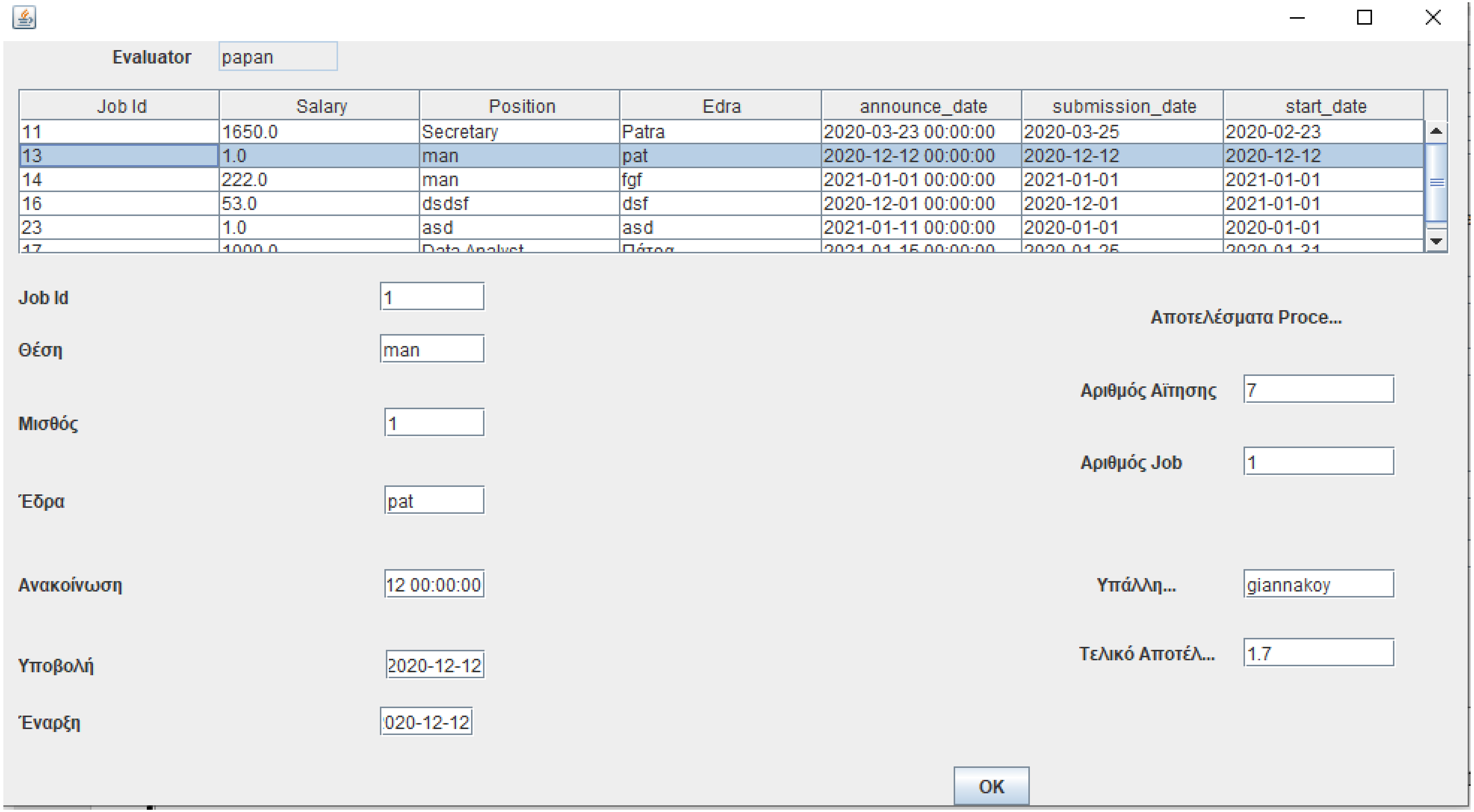
Αριθμός Αίτησης

Αριθμός Job

Υπάλλη...

Τελικό Αποτέλ...

Επιλέγοντας μια εργασία από τη λίστα εμφανίζονται αριστερά τα στοιχεία της και δεξιά τα αποτελέσματα της εκτέλεσης από την εκτέλεση της stored procedure 3.3



Evaluator papan

Job Id	Salary	Position	Edra	announce_date	submission_date	start_date
11	1650.0	Secretary	Patra	2020-03-23 00:00:00	2020-03-25	2020-02-23
13	1.0	man	pat	2020-12-12 00:00:00	2020-12-12	2020-12-12
14	222.0	man	fgf	2021-01-01 00:00:00	2021-01-01	2021-01-01
16	53.0	dsdsf	dsf	2020-12-01 00:00:00	2020-12-01	2021-01-01
23	1.0	asd	asd	2021-01-11 00:00:00	2020-01-01	2020-01-01
17	1000.0	Data Analyst	Πάτρα	2021-01-15 00:00:00	2020-01-25	2020-01-31

Job Id 1

Θέση man

Μισθός 1

Έδρα pat

Ανακοίνωση 12 00:00:00

Υποβολή 2020-12-12

Έναρξη 020-12-12

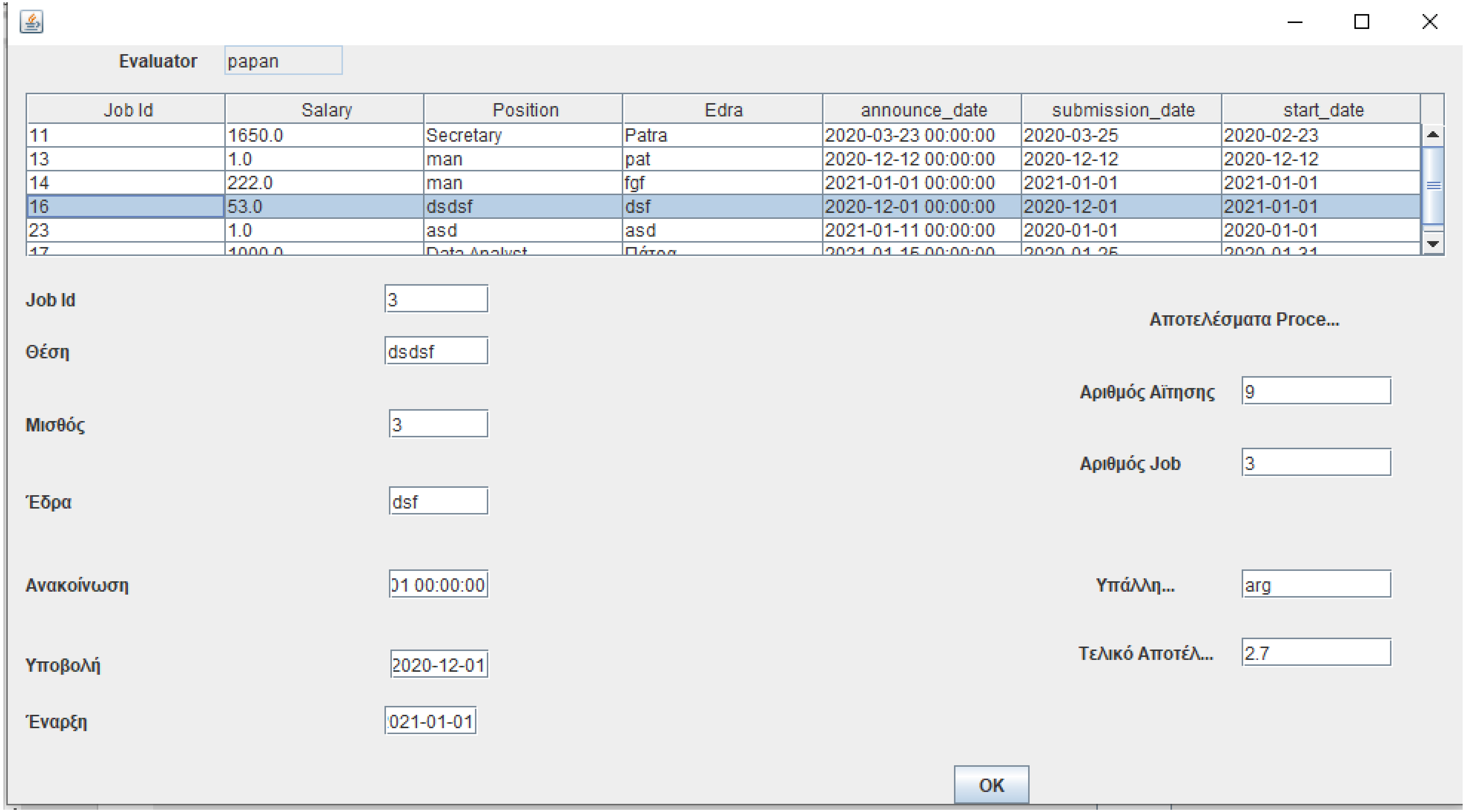
Αποτελέσματα Proce...

Αριθμός Αίτησης 7

Αριθμός Job 1

Υπάλλη... giannakoy

Τελικό Αποτέλ... 1.7



Evaluator papan

Job Id	Salary	Position	Edra	announce_date	submission_date	start_date
11	1650.0	Secretary	Patra	2020-03-23 00:00:00	2020-03-25	2020-02-23
13	1.0	man	pat	2020-12-12 00:00:00	2020-12-12	2020-12-12
14	222.0	man	fgf	2021-01-01 00:00:00	2021-01-01	2021-01-01
16	53.0	dsdsf	dsf	2020-12-01 00:00:00	2020-12-01	2021-01-01
23	1.0	asd	asd	2021-01-11 00:00:00	2020-01-01	2020-01-01
17	1000.0	Data Analyst	Πάτρα	2021-01-15 00:00:00	2020-01-25	2020-01-31

Job Id 3

Θέση dsdsf

Μισθός 3

Έδρα dsf

Ανακοίνωση)1 00:00:00

Υποβολή 2020-12-01

Έναρξη 021-01-01

Αποτελέσματα Proce...

Αριθμός Αίτησης 9

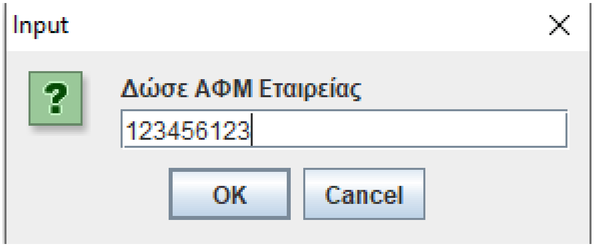
Αριθμός Job 3

Υπάλλη... arg

Τελικό Αποτέλ... 2.7

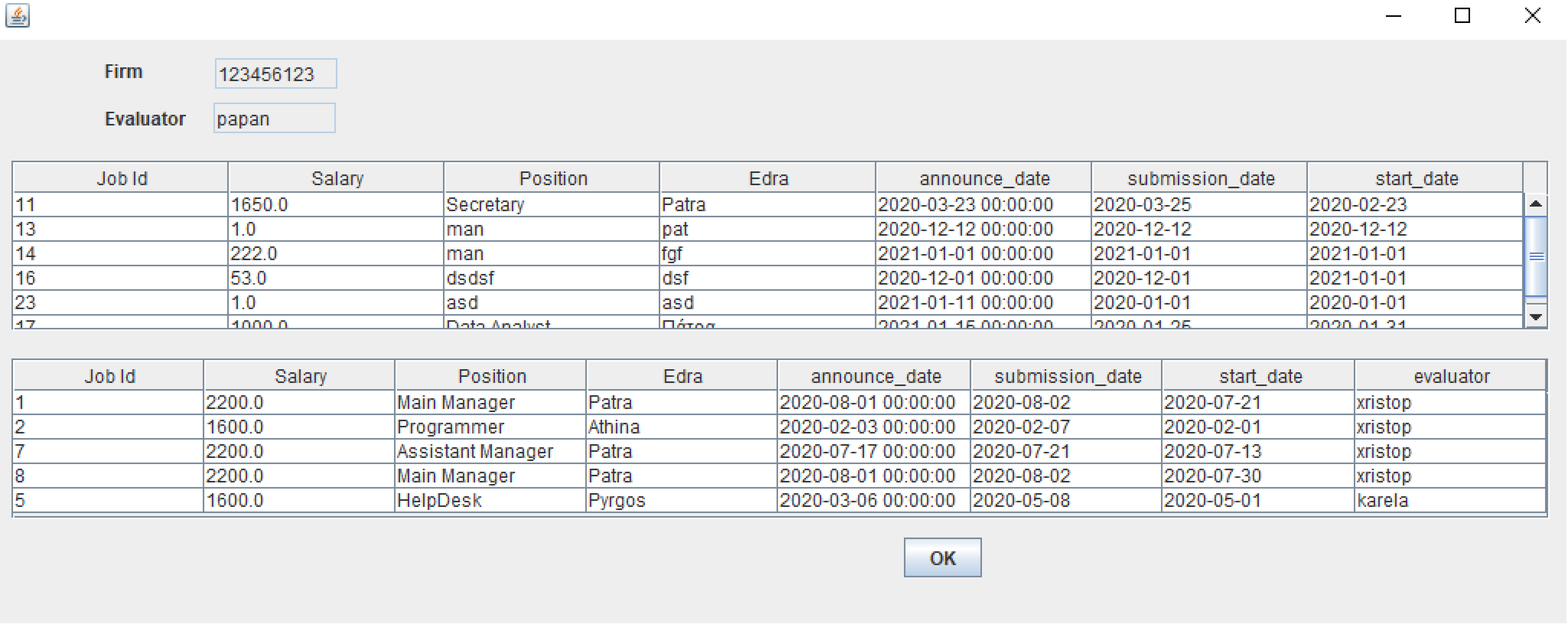
3.3.6 Λειτουργία 7 – Εμφάνιση Θέσεων που έχουν αναρτηθεί από την εταιρεία/Τροποποίηση μόνο των δικών του θέσεων

Πατώντας την επιλογή ‘Εμφάνιση Jobs Εταιρείας Evaluator’ εμφανίζεται η ακόλουθη εικόνα: στην οποία συμπληρώνουμε το ΑΦΜ Εταιρείας όπως φαίνεται στην επόμενη εικόνα:



Δώσε ΑΦΜ Εταιρείας

123456123



Firm 123456123

Evaluator papan

Job Id	Salary	Position	Edra	announce_date	submission_date	start_date
11	1650.0	Secretary	Patra	2020-03-23 00:00:00	2020-03-25	2020-02-23
13	1.0	man	pat	2020-12-12 00:00:00	2020-12-12	2020-12-12
14	222.0	man	fgf	2021-01-01 00:00:00	2021-01-01	2021-01-01
16	53.0	dsdsf	dsf	2020-12-01 00:00:00	2020-12-01	2021-01-01
23	1.0	asd	asd	2021-01-11 00:00:00	2020-01-01	2020-01-01
17	1000.0	Data Analyst	Πάτρα	2021-01-15 00:00:00	2020-01-25	2020-01-31

Job Id	Salary	Position	Edra	announce_date	submission_date	start_date	evaluator
1	2200.0	Main Manager	Patra	2020-08-01 00:00:00	2020-08-02	2020-07-21	xristop
2	1600.0	Programmer	Athina	2020-02-03 00:00:00	2020-02-07	2020-02-01	xristop
7	2200.0	Assistant Manager	Patra	2020-07-17 00:00:00	2020-07-21	2020-07-13	xristop
8	2200.0	Main Manager	Patra	2020-08-01 00:00:00	2020-08-02	2020-07-30	xristop
5	1600.0	HelpDesk	Pyrgos	2020-03-06 00:00:00	2020-05-08	2020-05-01	karela

Στην 1η λίστα εμφανίζεται των θέσεων που έχει αναρτήσει ο ίδιος και μπορεί να τις επεξεργαστεί

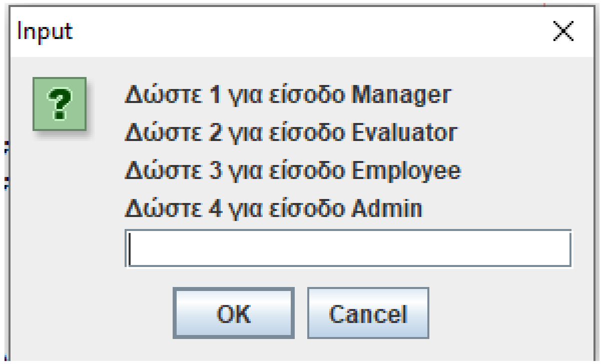
Στην 2η λίστα εμφανίζεται των θέσεων που έχει αναρτήσει ο ίδιος αλλά άλλοι evaluatorκαι της εταιρίας του δεν μπορεί να τις επεξεργαστεί

Πράγματι αν κάνει διπλό κλικ στην 1η λίστα ο evaluator μπορεί να επεξεργαστεί και να τροποποιήσει οποιαδήποτε θέση που έχει ανακοινώσει ο ίδιος. Αν επιχειρήσει να κάνει όμως διπλό κλικ όμως στη 2η λίστα αυτή είναι κλειδωμένη και τον εμποδίζει να κάνει οποιαδήποτε τροποποίηση,

3.4 Λειτουργίες Υπαλλήλου (Employee)

Στα παραδείγματα που παραθέτουμε χρησιμοποιούμε ως Employee το χρήστη με username arg.

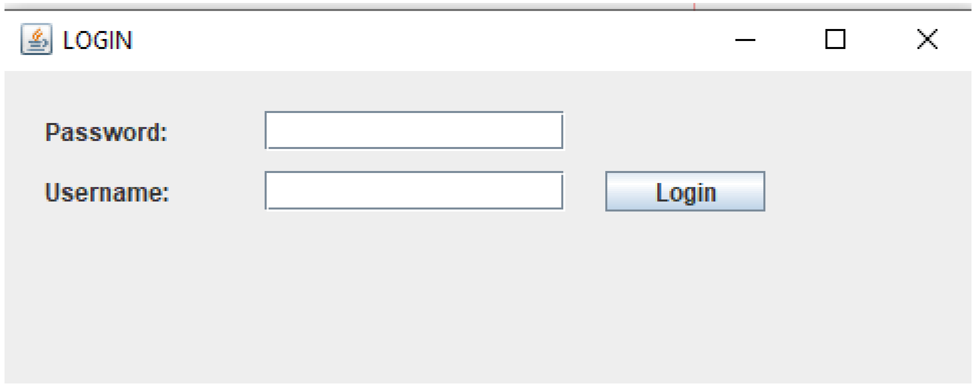
Αρχικά γίνεται είσοδος του Employee εκτελώντας το αρχείο login.java. Εμφανίζεται το ακόλουθο παράθυρο:



Input

Δώστε 1 για είσοδο Manager
Δώστε 2 για είσοδο Evaluator
Δώστε 3 για είσοδο Employee
Δώστε 4 για είσοδο Admin

Ακολούθως εμφανίζεται το παράθυρο LOGIN για να εισάγει ο Employee τα στοιχεία του:

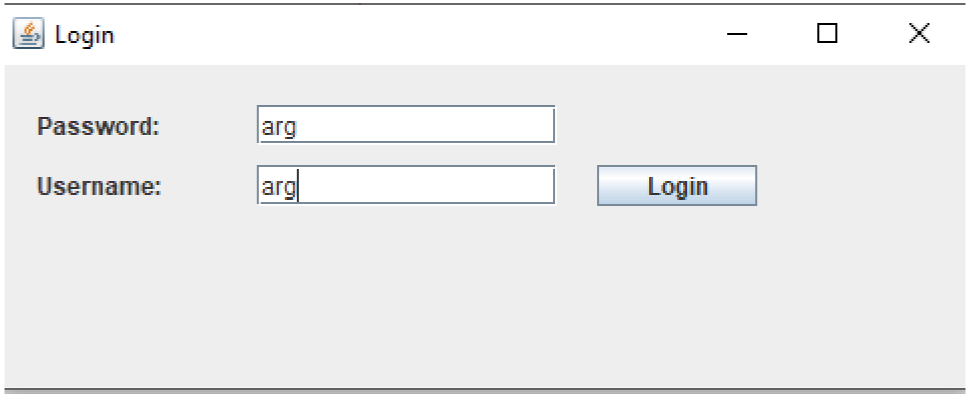


LOGIN

Password:

Username:

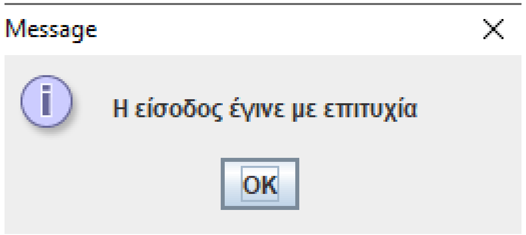
Συμπληρώνει τα στοιχεία του:



Password: arg

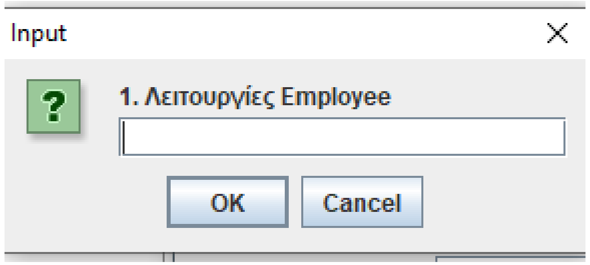
Username: arg

και εμφανίζεται το ακόλουθο μήνυμα:



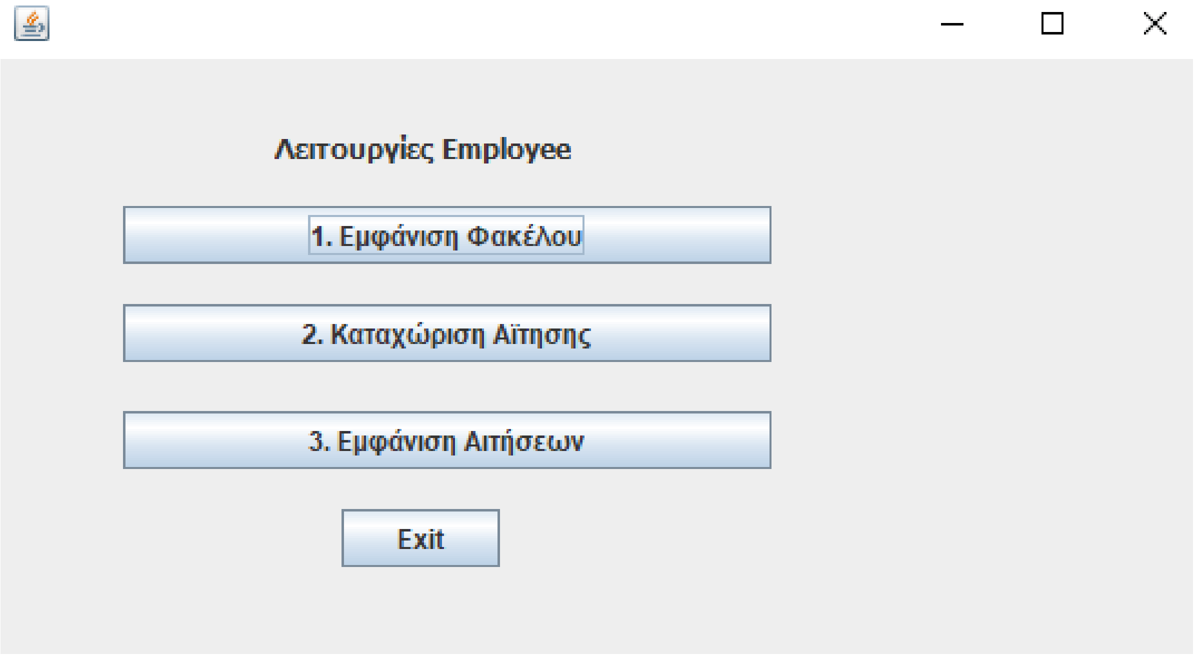
Η είσοδος έγινε με επιτυχία

Μετά εμφανίζεται η ακόλουθη εικόνα:



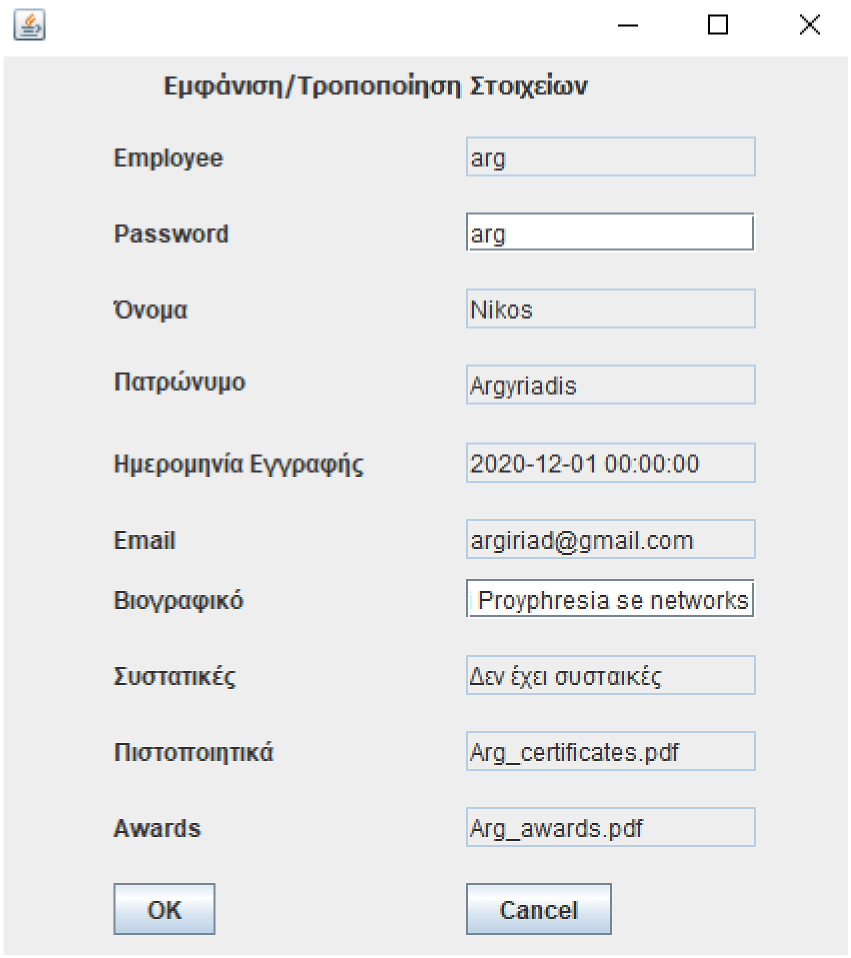
1. Λειτουργίες Employee

Εκεί δίνουμε την τιμή 1 και εμφανίζεται το επόμενο μενού με τις λειτουργίες του Employee



3.4.1 Λειτουργία 1 – Εμφάνιση Φακέλου και Επεξεργασία των Στοιχείων του

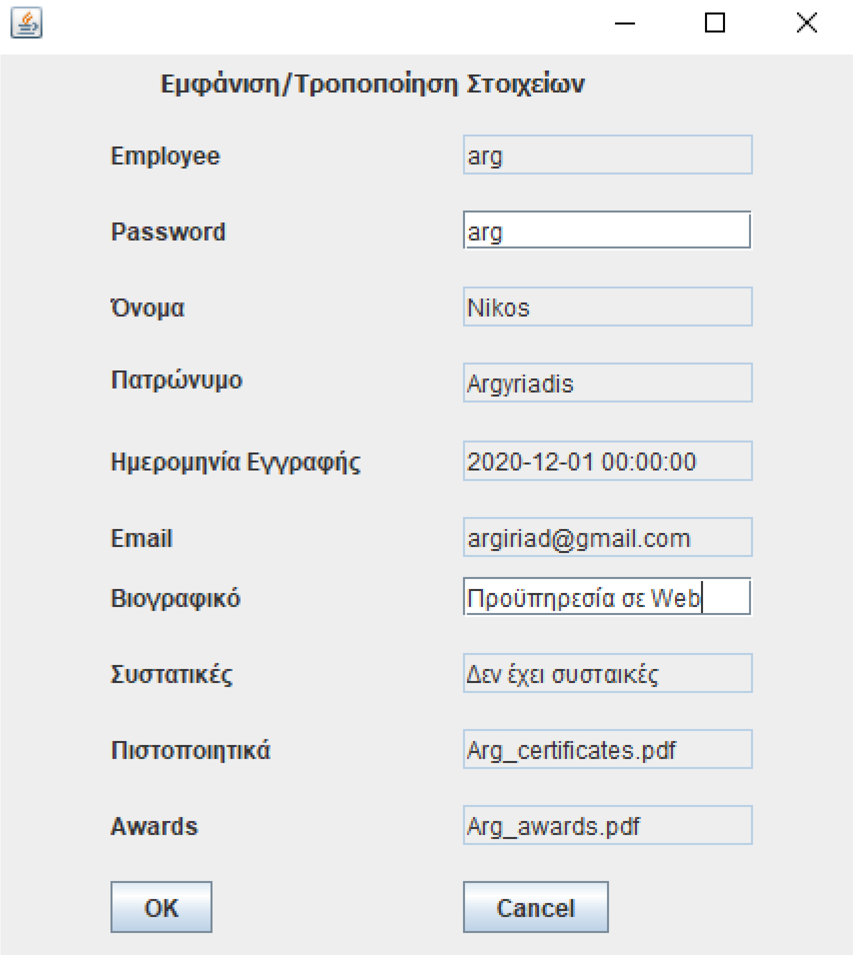
Πατώντας το πλήκτρο 'Εμφάνιση Φακέλου 'εμφανίζεται το ακόλουθο παράθυρο:



Εμφάνιση/Τροποποίηση Στοιχείων

Employee	arg
Password	arg
Όνομα	Nikos
Πατρώνυμο	Argyriadis
Ημερομηνία Εγγραφής	2020-12-01 00:00:00
Email	argiriad@gmail.com
Βιογραφικό	Proyphresia se networks
Συστατικές	Δεν έχει συσταικές
Πιστοποιητικά	Arg_certificates.pdf
Awards	Arg_awards.pdf

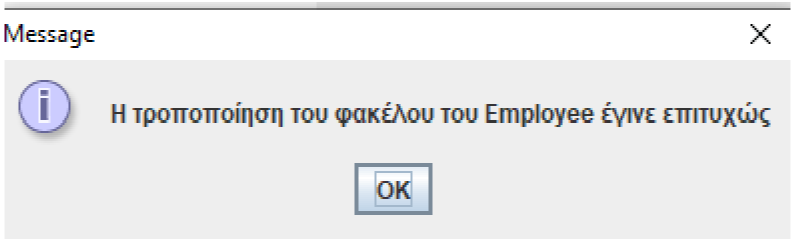
Ο employee βλέπει το φάκελο του και μπορεί να τροποποιήσει μόνο password και Βιογραφικό Για παράδειγμα έστω ότι ο υπάλληλος arg τροποποιεί το βιογραφικό του



Εμφάνιση/Τροποποίηση Στοιχείων

Employee	arg
Password	arg
Όνομα	Nikos
Πατρώνυμο	Argyriadis
Ημερομηνία Εγγραφής	2020-12-01 00:00:00
Email	argiriad@gmail.com
Βιογραφικό	Προϋπηρεσία σε Web
Συστατικές	Δεν έχει συσταικές
Πιστοποιητικά	Arg_certificates.pdf
Awards	Arg_awards.pdf

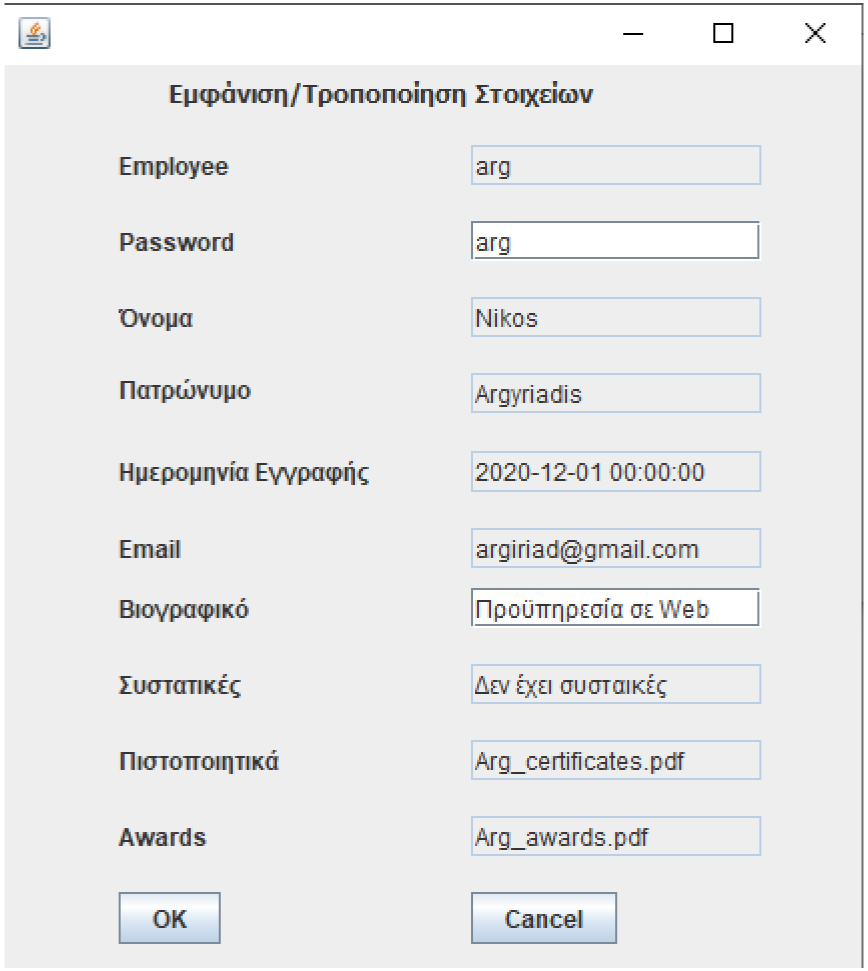
και πατώντας ΟΚ εμφανίζεται το ακόλουθο μήνυμα:



Message

Η τροποποίηση του φακέλου του Employee έγινε επιτυχώς

Αν ξαναπατήσουμε το ίδιο πλήκτρο παρατηρούμε ότι η αλλαγή έχει καταχωρηθεί

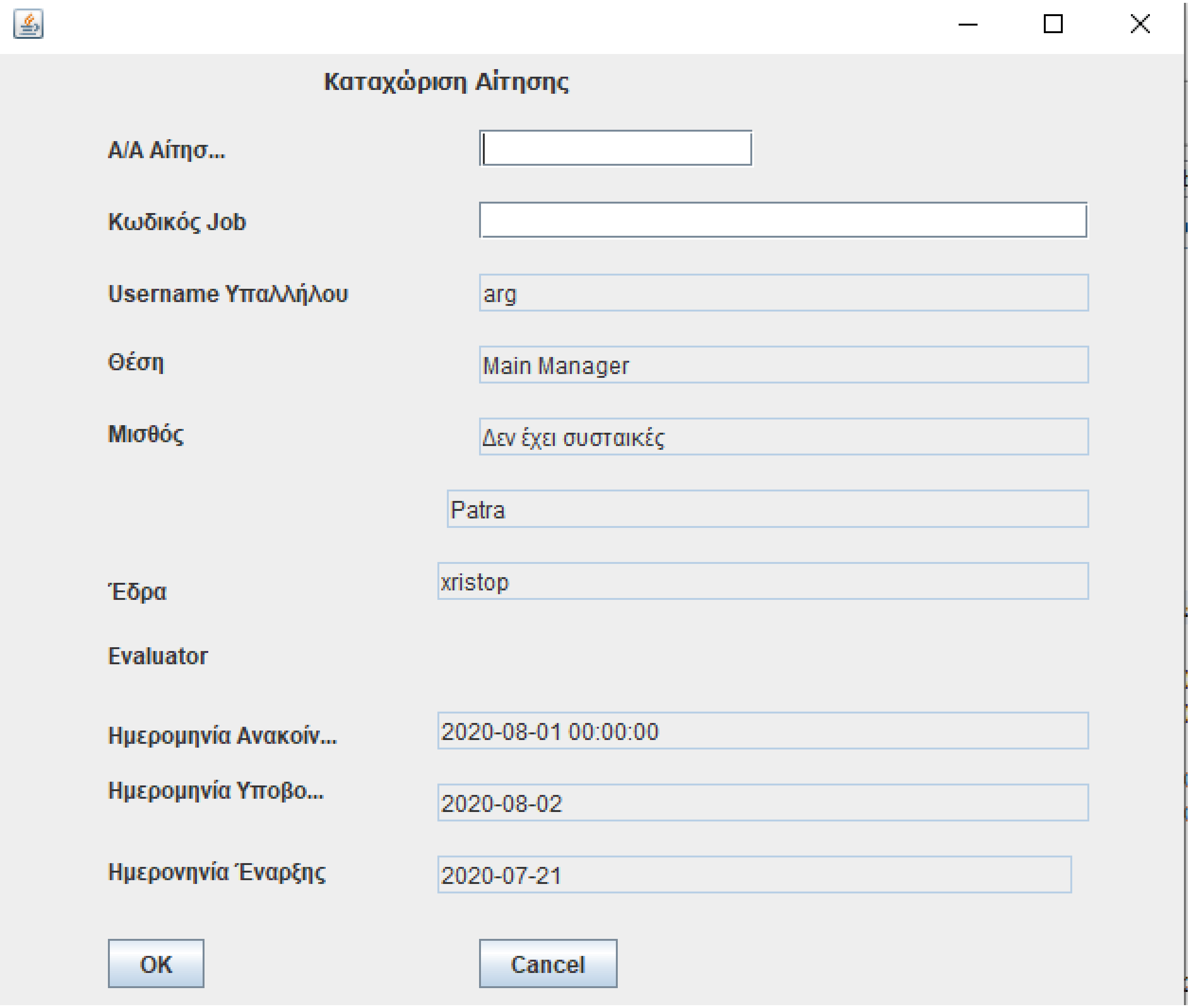


Εμφάνιση/Τροποποίηση Στοιχείων

Employee	arg
Password	arg
Όνομα	Nikos
Πατρώνυμο	Argyriadis
Ημερομηνία Εγγραφής	2020-12-01 00:00:00
Email	argiriad@gmail.com
Βιογραφικό	Προϋπηρεσία σε Web
Συστατικές	Δεν έχει συσταικές
Πιστοποιητικά	Arg_certificates.pdf
Awards	Arg_awards.pdf

3.4.2 Λειτουργία 2 – Καταχώριση Αίτησης σε ανοιχτές θέσεις

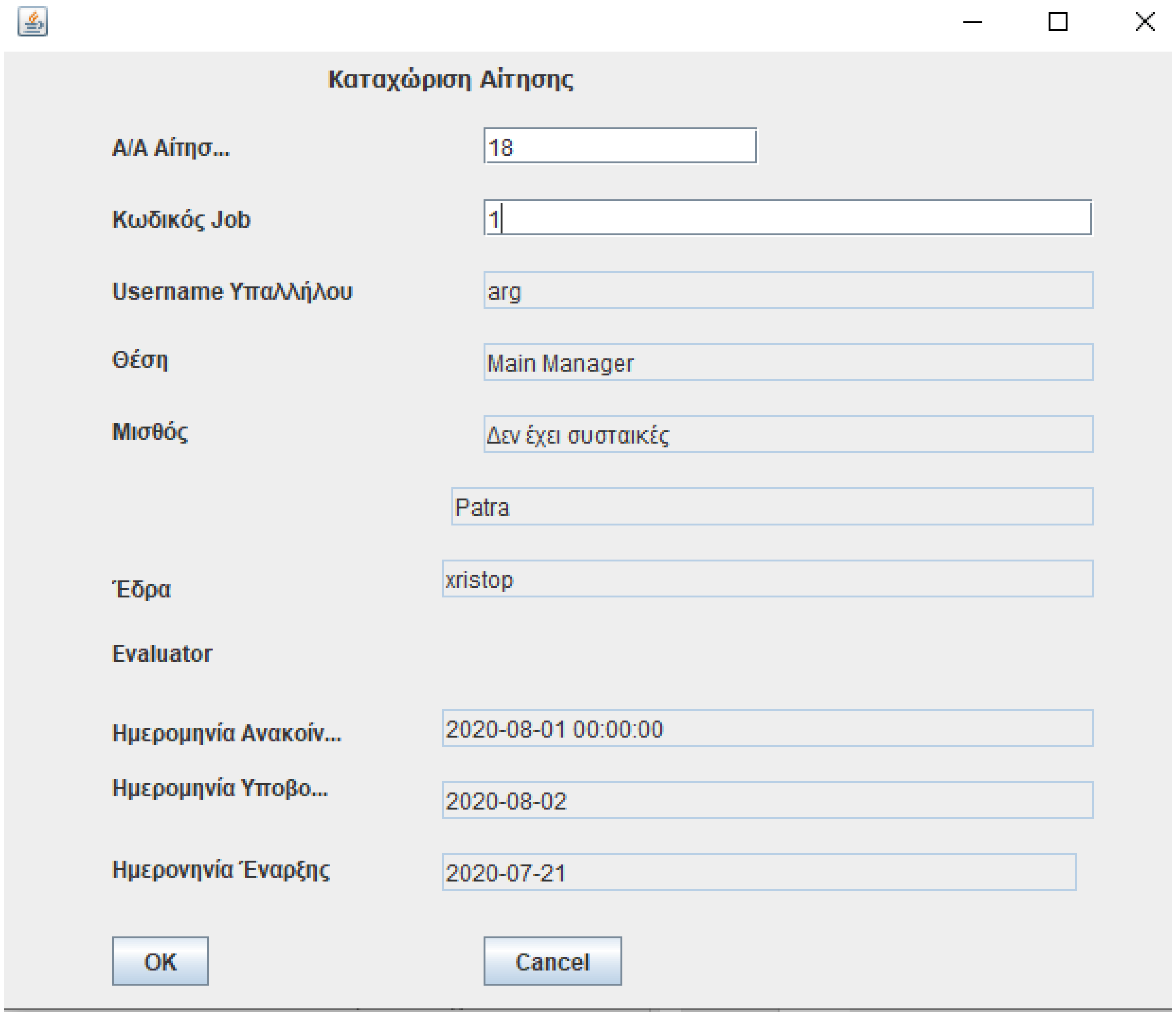
Πατώντας τη 2η επιλογή 'Καταχώριση Αίτησης' εμφανίζεται η ακόλουθη εικόνα:



Καταχώριση Αίτησης

A/A Αίτησ...	
Κωδικός Job	
Username Υπαλλήλου	arg
Θέση	Main Manager
Μισθός	Δεν έχει συσταικές
	Patra
Έδρα	xristop
Evaluator	
Ημερομηνία Ανακοίν...	2020-08-01 00:00:00
Ημερομηνία Υποβο...	2020-08-02
Ημερονηνία Έναρξης	2020-07-21

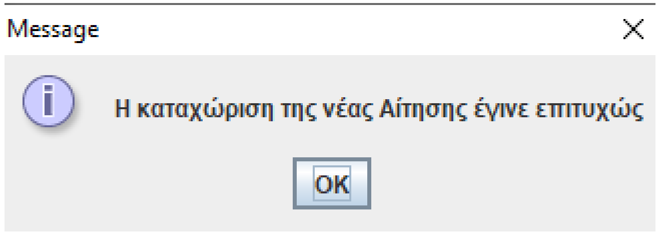
Ο Υπάλληλος συμπληρώνει τα στοιχείς μιας νέας αίτησης όπως στο επόμενο σχήμα:



Καταχώριση Αίτησης

Α/Α Αίτησ...	18
Κωδικός Job	1
Username Υπαλλήλου	arg
Θέση	Main Manager
Μισθός	Δεν έχει συσταικές
	Patra
Έδρα	xristop
Evaluator	
Ημερομηνία Ανακοίν...	2020-08-01 00:00:00
Ημερομηνία Υποβο...	2020-08-02
Ημερονηνία Έναρξης	2020-07-21

Πατώντας OK εμφανίζεται το επόμενο μήνυμα:

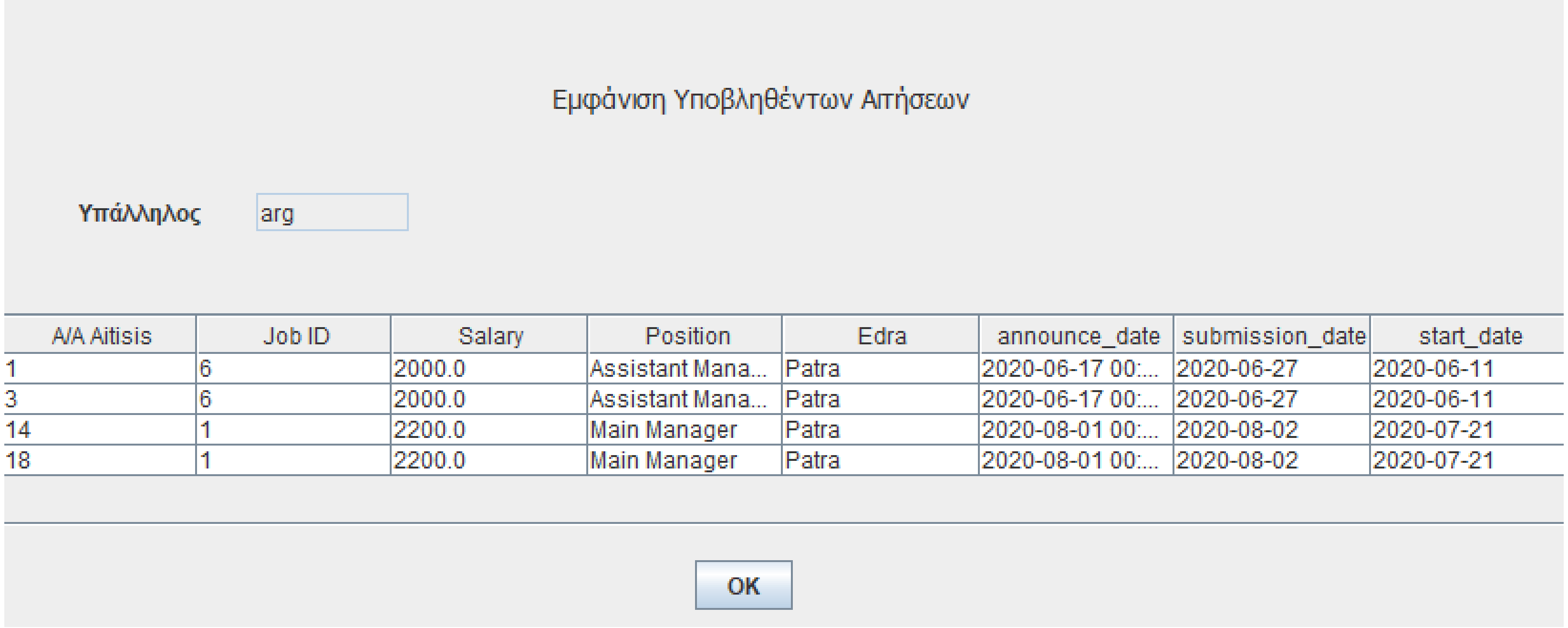


Message

Η καταχώριση της νέας Αίτησης έγινε επιτυχώς

3.4.3 Λειτουργία 3 – Εμφάνιση Υποβληθέντων Αιτήσεων

Πατώντας τη 2η επιλογή ′Εμφάνιση Αιτήσεων′ εμφανίζεται η ακόλουθη εικόνα:



Εμφάνιση Υποβληθέντων Αιτήσεων

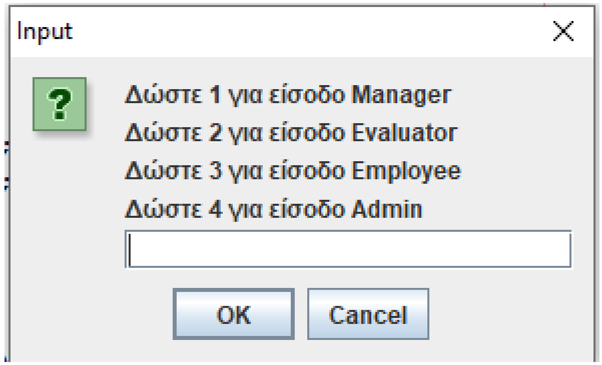
Υπάλληλος: arg

A/A Aitisis	Job ID	Salary	Position	Edra	announce_date	submission_date	start_date
1	6	2000.0	Assistant Mana...	Patra	2020-06-17 00:...	2020-06-27	2020-06-11
3	6	2000.0	Assistant Mana...	Patra	2020-06-17 00:...	2020-06-27	2020-06-11
14	1	2200.0	Main Manager	Patra	2020-08-01 00:...	2020-08-02	2020-07-21
18	1	2200.0	Main Manager	Patra	2020-08-01 00:...	2020-08-02	2020-07-21

Παρατηρούμε ότι ως επιβεβαίωση ότι η προηγούμενη αίτηση έχει καταχωρηθεί στο σύστημα

3.5 Λειτουργίες Διαχειριστή (Admin)

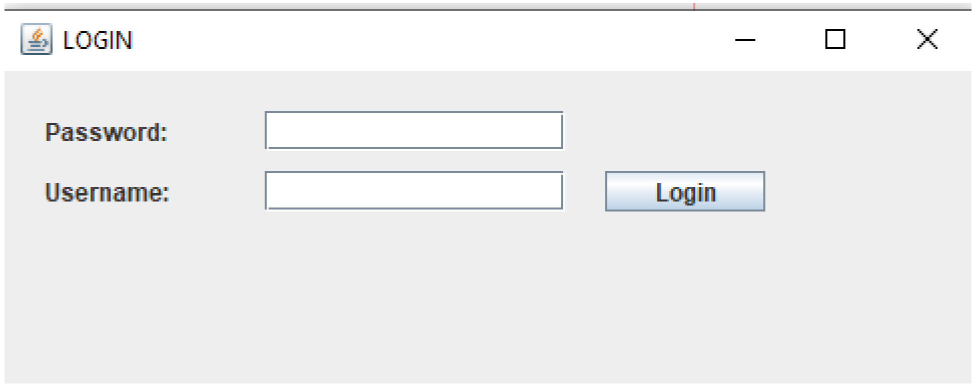
Αρχικά γίνεται είσοδος του Διαχειριστή εκτελώντας το αρχείο login.java. Εμφανίζεται το ακόλουθο παράθυρο:



Input

Δώστε 1 για είσοδο Manager
Δώστε 2 για είσοδο Evaluator
Δώστε 3 για είσοδο Employee
Δώστε 4 για είσοδο Admin

Ακολούθως εμφανίζεται το παράθυρο LOGIN για να εισάγει ο Admin τα στοιχεία του:

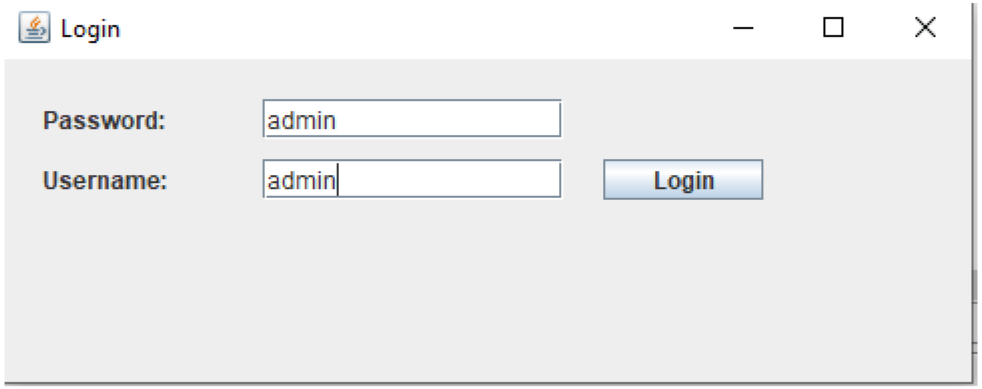


LOGIN

Password:

Username:

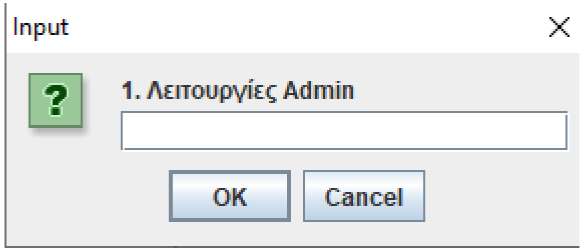
Συμπληρώνει τα στοιχεία του:



Password: admin

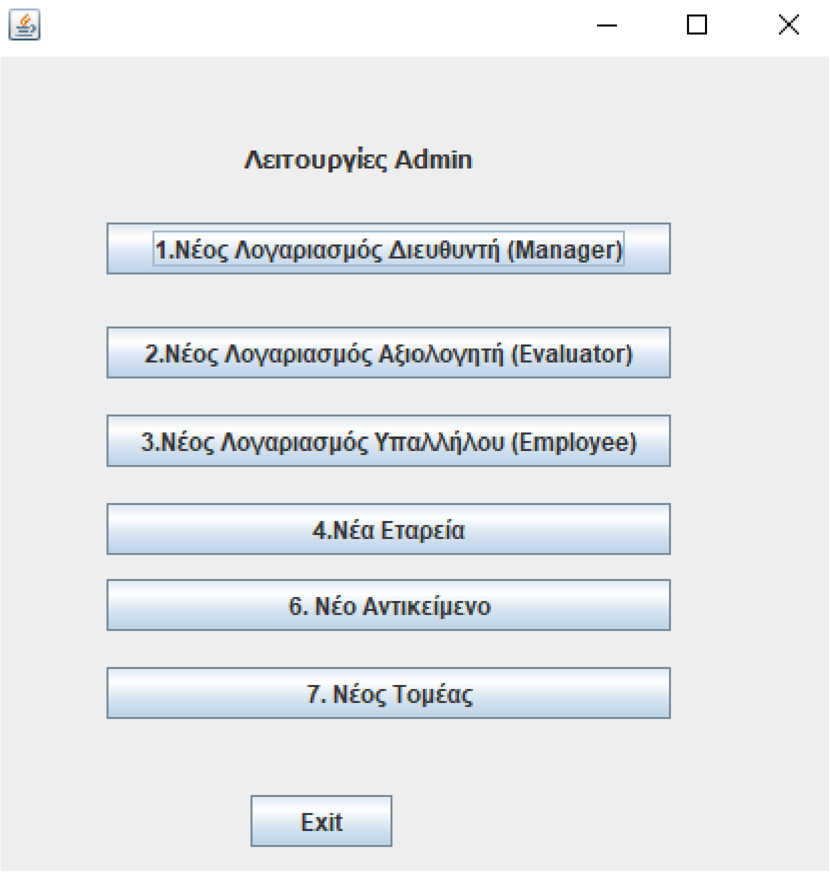
Username: admin

Εμφανίζεται το επόμενο παράθυρο στο οποίο εισάγει το 1:



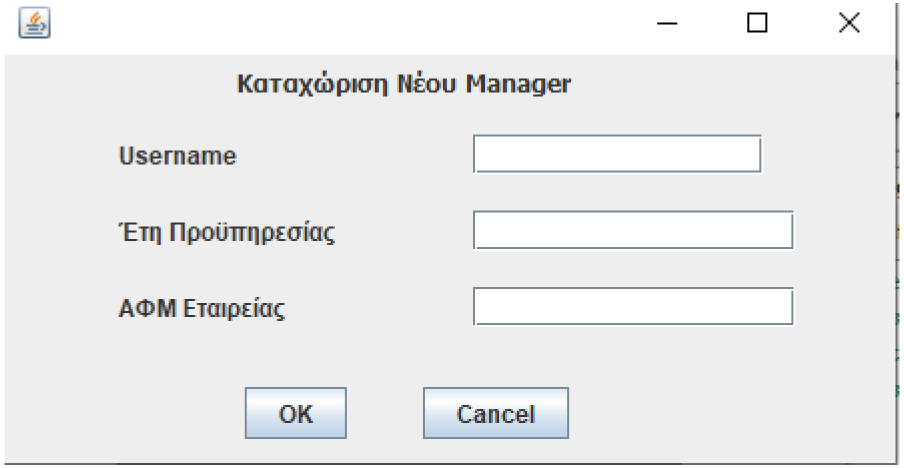
1. Λειτουργίες Admin

και εμφανίζεται το ακόλουθο μενού:


Λειτουργίες Admin
1.Νέος Λογαριασμός Διευθυντή (Manager)
2.Νέος Λογαριασμός Αξιολογητή (Evaluator)
3.Νέος Λογαριασμός Υπαλλήλου (Employee)
4.Νέα Εταρεία
6. Νέο Αντικείμενο
7. Νέος Τομέας
Exit

3.5.1 Λειτουργία 1 – Δημιουργία Λογαριασμού για νέο Διευθυντή

Πατώντας την 1η επιλογή ′1. Νέος Λογαριασμός Διευθυντή (Manager) ′ εμφανίζεται η ακόλουθη εικόνα όπου καταχωρούμε τα στοιχεία για ένα νέο Διευθυντή:



Καταχώριση Νέου Manager

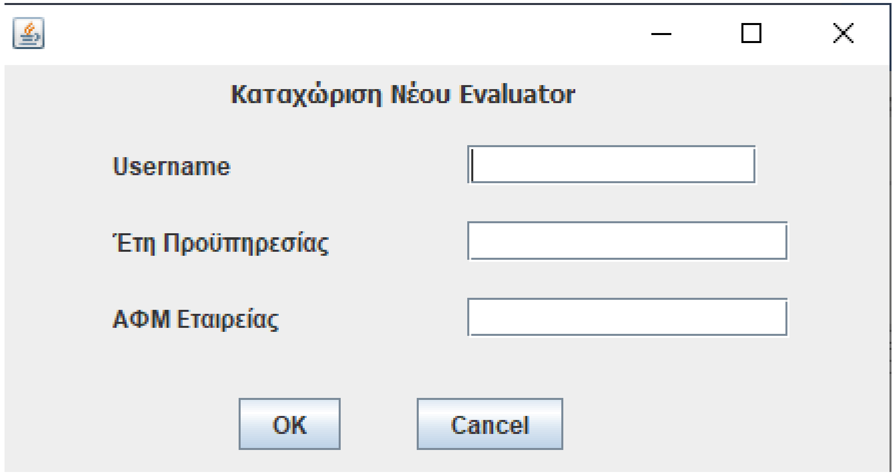
Username

Έτη Προϋπηρεσίας

ΑΦΜ Εταιρείας

3.5.2 Λειτουργία 2 – Δημιουργία Λογαριασμού για νέο Αξιολογητή

Πατώντας την 2η επιλογή ′2. Νέος Λογαριασμός Αξιολογητή (Evaluator) ′ εμφανίζεται η ακόλουθη εικόνα όπου καταχωρούμε τα στοιχεία για ένα νέο αξιολογητή



Καταχώριση Νέου Evaluator

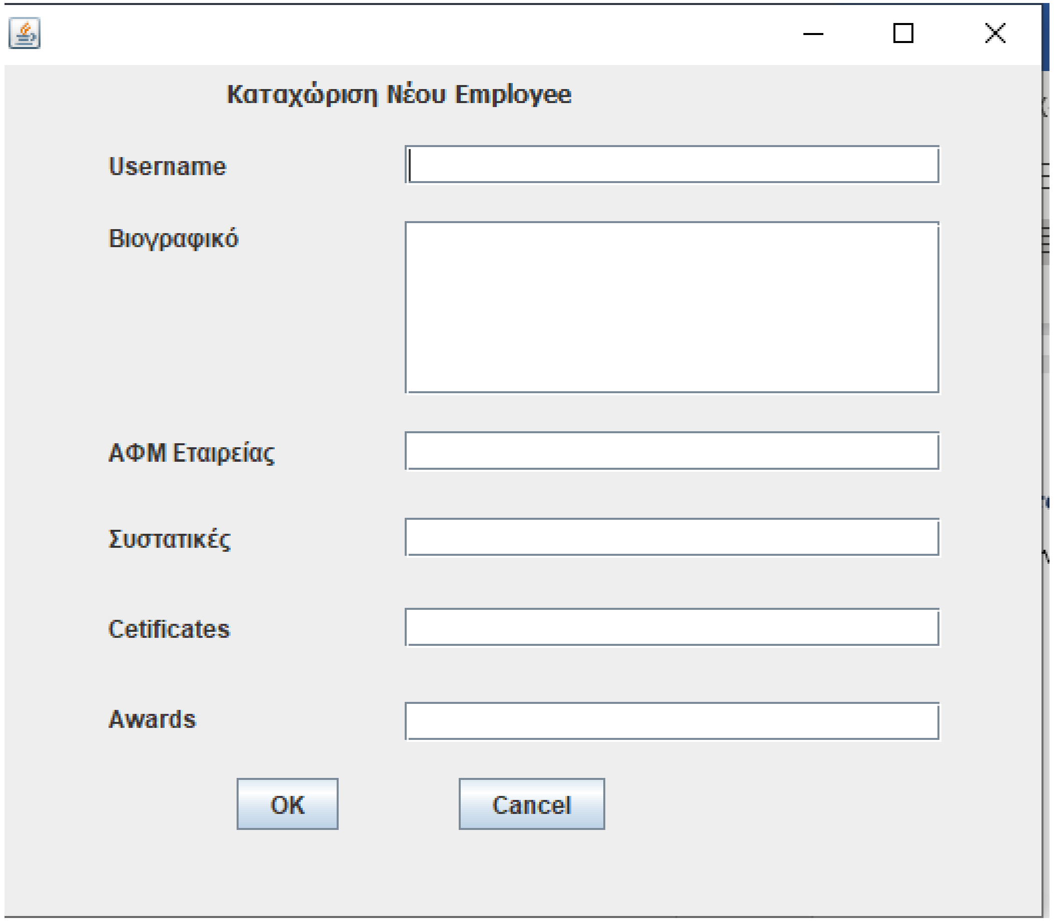
Username

Έτη Προϋπηρεσίας

ΑΦΜ Εταιρείας

3.5.3 Λειτουργία 3 – Δημιουργία Λογαριασμού για νέο Υπάλληλο

Πατώντας την 3η επιλογή '3. Νέος Λογαριασμός Υπαλλήλου (Employee) ' εμφανίζεται η ακόλουθη εικόνα όπου καταχωρούμε τα στοιχεία για ένα νέο υπάλληλο



Καταχώριση Νέου Employee

Username

Βιογραφικό

ΑΦΜ Εταιρείας

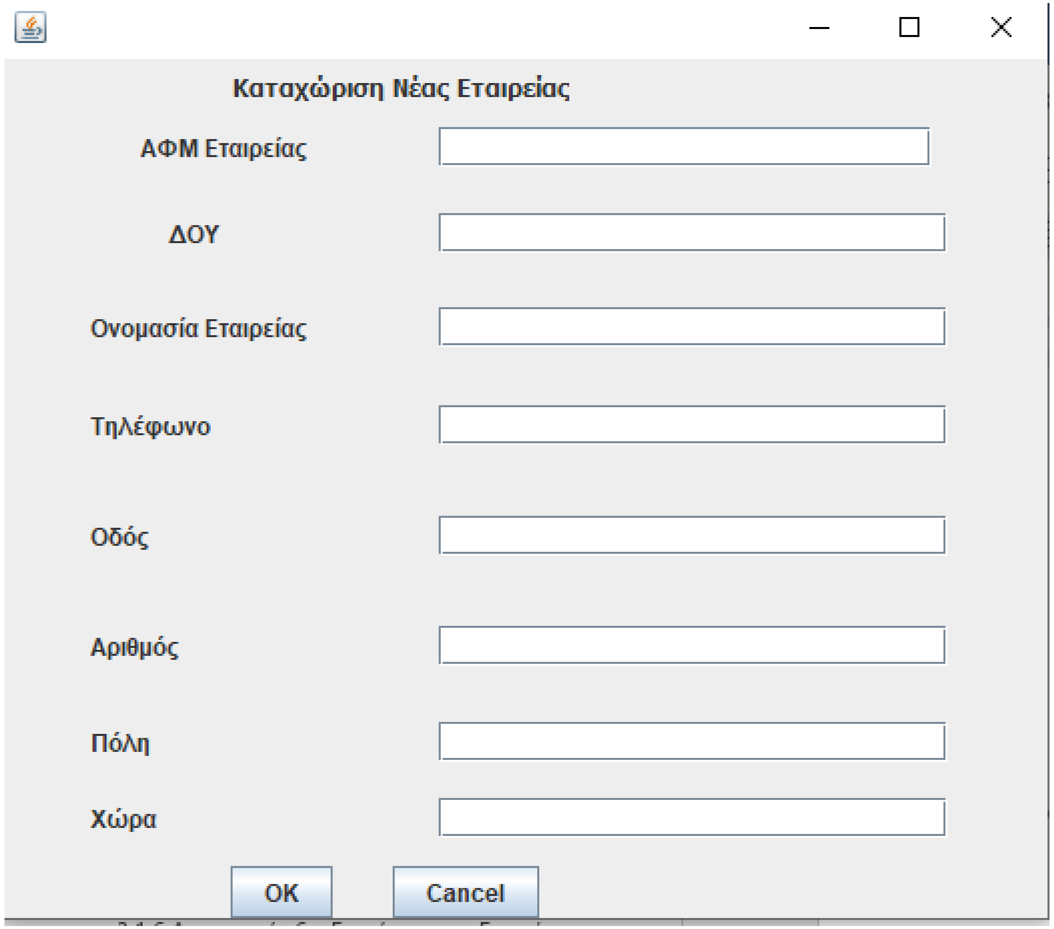
Συστατικές

Cetificates

Awards

3.5.4 Λειτουργία 4 – Δημιουργία νέα Εταιρείας

Πατώντας την 4η επιλογή '4.Νέα Εταιρεία ' εμφανίζεται η ακόλουθη εικόνα όπου καταχωρούμε τα στοιχεία για μια νέα εταιρεία



Καταχώριση Νέας Εταιρείας

ΑΦΜ Εταιρείας

ΔΟΥ

Ονομασία Εταιρείας

Τηλέφωνο

Οδός

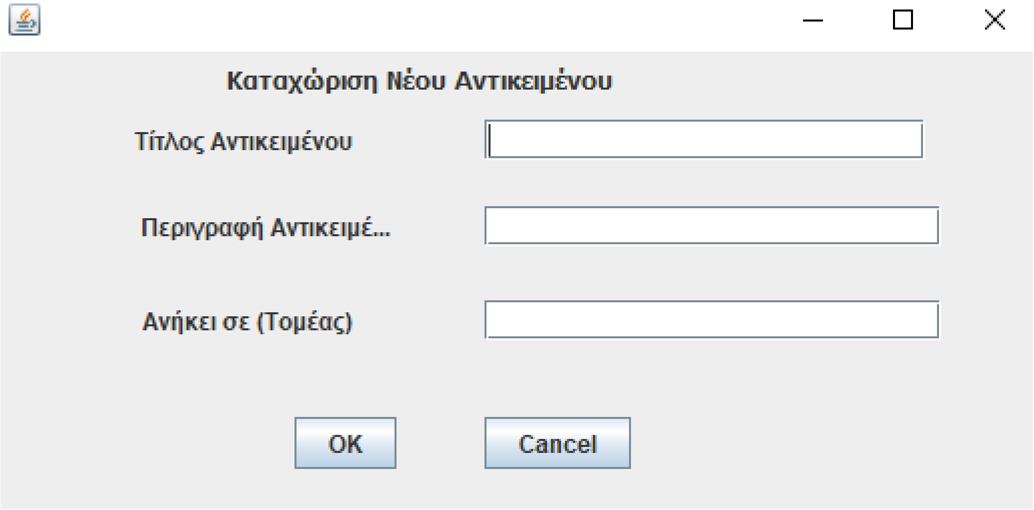
Αριθμός

Πόλη

Χώρα

3.5.5 Λειτουργία 5 – Δημιουργία νέου Αντικειμένου

Πατώντας την 5η επιλογή ′5. Νέο Αντικείμενο′ εμφανίζεται η ακόλουθη εικόνα όπου καταχωρούμε τα στοιχεία για ένα νέο αντικείμενο


Καταχώριση Νέου Αντικειμένου

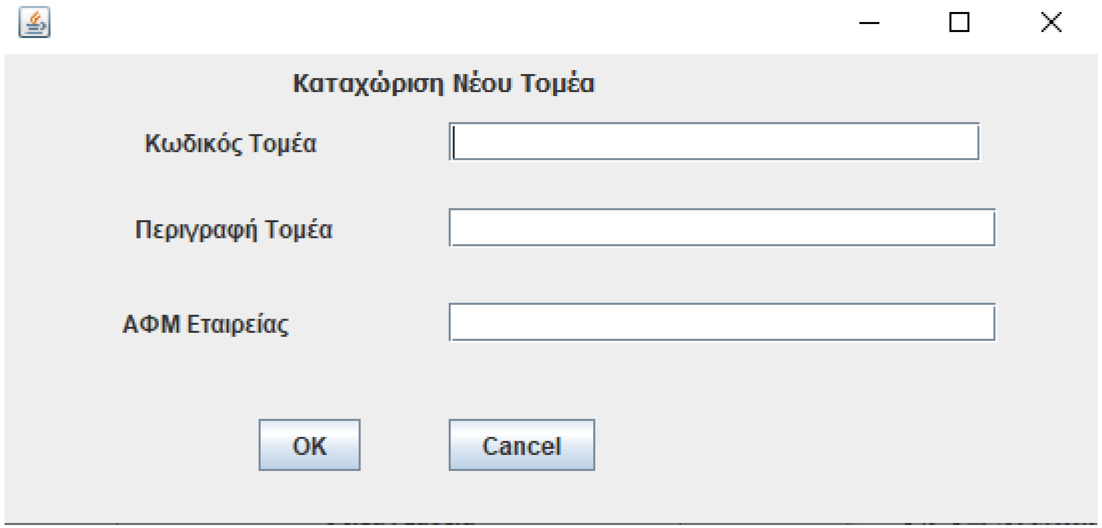
Τίτλος Αντικειμένου

Περιγραφή Αντικειμέ...

Ανήκει σε (Τομέας)

3.5.6 Λειτουργία 6 – Δημιουργία Λογαριασμού για νέο Αντικείμενο

Πατώντας την 6η επιλογή ′6. Νέο Αντικείμενο′ εμφανίζεται η ακόλουθη εικόνα όπου καταχωρούμε τα στοιχεία για ένα νέο τομέα της εταιρείας



Στην επιλογή αυτή δημιουργήσαμε ένα νέο πίνακα σύμφωνα με τις ακόλουθες εντολές:

```
create table tomeas(kt int not null,decription varchar(50),AFM char(9),primary key(kt), constraint wz foreign key(AFM) references company(AFM));
insert into tomeas values(1,'Kataskeyazei logistika kai emporika paketa','123456123');
insert into tomeas values(2,'Kataskeyazei paketa sxediasis gia mixanikous','123456123');
```

Παράρτημα με Κώδικα Βάσης

```
drop database if exists staffevaluation;

create database staffevaluation;
use staffevaluation;

create table company(AFM char(9) primary key not null,DOY varchar(15), name varchar(35), phone bigint(16),street varchar(15),num tinyint(4), city varchar(15),country varchar(15));
create table antikeim(title varchar(35) primary key, descr tinytext, belongs_to varchar(36),constraint b foreign key(belongs_to) references antikeim(title));


create table user(username varchar(12) primary key,password varchar(10),name varchar(25),surname varchar(35), reg_date datetime, email varchar(20),constraint c foreign key(username) references manager(username) on delete cascade on update cascade);

create table manager(managerusername varchar(12),exp_years tinyint(4),firm char(9),primary key(managerusername,firm), constraint a1 foreign key(firm) references company(AFM), constraint a2 foreign key(managerusername) references user(username));

create table employee(username varchar(12) primary key,bio text,sistatikes varchar(35), certificates varchar(35), awards varchar(35));

create table evaluator(username varchar(12) primary key, exp_years tinyint(4), firm char(9), constraint d foreign key(username) references manager(username) on delete cascade on update cascade, constraint e foreign key(firm) references company(AFM) on delete cascade on update cascade);


create table job(id int(4) primary key, start_date date, salary float(6,1),position varchar(40), edra varchar(45), evaluator varchar(12), announce_date datetime, submission_date date,constraint f foreign key(evaluator) references evaluator(username) on delete cascade on update cascade);

create table project(empl varchar(12),num tinyint(4),descr text, url varchar(60),primary key(empl,num), constraint f foreign key(num) references needs(job_id) on delete cascade on update cascade);
create table needs(job_id int(4),antikeim_title varchar(36),primary key(job_id,antikeim_title), constraint g foreign key(antikeim_title) references antikeim(title) on delete cascade on update cascade);

create table requestsevaluation(empl_username varchar(12),job_id int(4),primary key(empl_username,job_id),constraint h foreign key(empl_username) references employee(username) on delete cascade on update cascade, constraint j foreign key(job_id) references job(job_id) on delete cascade on update cascade);

create table languages(employee varchar(12) primary key, lang set('EN','FR','SP','GR'), constraint k foreign key(employee) references employee(username) on delete cascade on update cascade);

create table degree(titlos varchar(50), idryma varchar(40),primary key(titlos,idryma));

create table has_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),constraint m foreign key(degr_title) references degree(title),constraint n foreign key(empl_username) references employee(username));

create table evaluationresult(Evld int(4) not null auto_increment, empl_username varchar(12), job_id int(4), grade int(4), comments varchar(255),primary key(evld,empl_username), constraint p foreign key(empl_username) references employee(username), constraint q foreign key(job_id) references job(job_id));


create table evaluation_process(aa int primary key,job_id int,
bathmos_synenteyxis ENUM('0','1','2','3','4'),sxolia_synenetyxis text,xaraktirismos_synenteyxis ENUM('1','2','3'),
bathmos_report ENUM('0','1','2','3','4'), sxolia_report text,xaraktirismos_report ENUM('1','2','3'),
bathmos_loipon ENUM('0','1','2'),sxolia_loipon text,xaraktirismos_loipon ENUM('1','2','3'),
username_eval varchar(12),
username_emp varchar(12),
teliko_apotelesma float (3,1),
constraint z1 foreign key(username_eval) references evaluator(username),
```

```sql
constraint z2 foreign key(username_emp) references employee(username),
constraint z3 foreign key(job_id) references job(job_id));

create table ypoboli_aitiseon(aa_aitisis int primary key auto_increment,job_id int, eusername varchar(12), constraint t foreign key(job_id) references job(job_id),constraint z foreign key(eusername) references employee(username));

create table log(username varchar(12),hmer_ora datetime,apotelesma varchar(20),eidos_energeias varchar(20),onoma_pinaka varchar(20));


-- -------------------------------------------------------------------------------------------------------------------------------------------------------
insert into company values('123456123','A Patras','Kataskeyasti Patras','2610-394509','Korinthou',10,'Patras','Greece');
insert into company values('123456888','B Patras','Emporio Sidiroy Patras','2610-324317','Avlonos',20,'Patras','Greece');
insert into company values('100045608','Pyrgou','Emporio Domion Ylikon','23610-52455','Korinthou',87,'Patras','Greece');

insert into user values('admin','admin','Manos','Giannakopoulos','2020-1-1','manos@gmail.com');
insert into user values('arg','000','Nikos','Argyriadis','2020-12-1','argiriad@gmail.com');
insert into user values('manolop','657','Kostas','Manolopoulos','2020-11-1','manolop@gmail.com');
insert into user values('giannakoy','123','Maria','Giannakou','2020-11-1','gianakou@gmail.com');
insert into user values('dimop','233','Giorgos','Dimopoulos','2019-11-10','dimop$hotmail.com');
insert into user values('delig','999','Andreas','Deligiannis','2018-6-10','delig$hotmail.com');
insert into user values('dalak','551','Anna','Dalaka','2017-1-9','dalak$hotmail.com');
insert into user values('papan','112','Nikos','Papanikolaou','2020-1-1','argiriad@hotmail.com');
insert into user values('papag','555','Kostas','Papageorgiou','2020-9-1','papag@hotmail.com');
insert into user values('papad','563','Vasilis','Papdionysiou','2020-8-1','papad@gmail.com');
insert into user values('xristop','345','Eleni','Papanikolaou','2020-51-1','xristop@gmail.com');
insert into user values('xorafop','346','Anna','Papageorgiou','2020-7-1','xorafop@gmail.com');
insert into user values('karela','347','Sofia','Karela','2020-4-1','karela@gmail.com');
insert into manager values('papan',10,'123456123');
insert into manager values('papag',8,'123456888');
insert into manager values('papad',7,'100045608');



insert into employee values('arg','Exei megali Proyphresia se networks','exei polles sistatikes','arg_certificates.pdf','arg_awards.pdf');
insert into employee values('manolop','Exei mikri Proyphresia se networks','exei liges sistatikes','manolop_certificates.pdf','manolop_awards.pdf');
insert into employee values('giannakoy','Exei sxetika megali Proyphresia se Databases','exei arrketes sistatikes','giannakoy_certificates.pdf','giannakoy_awards.pdf');
insert into employee values('dimop','Exei Proyphresia se Databases','exei sistatikes apo proigoumenes douleis','dimop_certificates.pdf','dimop_awards.pdf');
insert into employee values('delig','Exei doulepsei poly se ierarxikes Databases','exei liges sistatikes','delig.pdf','delig.pdf');
insert into employee values('dalak','Exei empeiria se Databases kai Networks','Polles sistatikes','dalak_certificates.pdf','dalak_awards.pdf');




insert into employee values('karela','Enai i pio kali evaluator','Polles systatikes','karela_certificates.pdf','karela_awards.pdf');
insert into employee values('xristop','Enai i pio kali evaluator','Liges systatikes','xristop_certificates.pdf','xristop_awards.pdf');
insert into employee values('xorafop','Enai i pio kali evaluator','Polles systatikes','xorafop_certificates.pdf','xorafop_awards.pdf');



insert into evaluator values('papan',9,'123456123');
insert into evaluator values('papag',8,'123456888');
insert into evaluator values('papad',7,'100045608');

insert into evaluator values('xristop',10,'123456123');
insert into evaluator values('xorafop',20,'123456888');
insert into evaluator values('karela',15,'123456123');
insert into evaluator values('karela',15,'123456888');
insert into antikeim values('Networks','Root Element',NULL);
insert into antikeim values('Local Area Networks','Second Level Element','Networks');
insert into antikeim values('Wide Area Networks','Second Level Element','Networks');
insert into antikeim values('Metropolitan Area Networks','Second Level Element','Networks');
insert into antikeim values('Databases','Root Element',NULL);
insert into antikeim values('MySQL','Second Level Element','Databases');
insert into antikeim values('SQL Server','Second Level Element','Databases');


insert into job values(1,'2020-1-1',1500,'Programmer','Patra','xristop','2020-1-2','2020-1-5');
insert into job values(2,'2020-2-1',1600,'Programmer','Athina','xristop','2020-2-3','2020-2-7');
insert into job values(3,'2020-3-1',1000,'Analyst','Pyrgos','xorafop','2020-3-6','2020-3-8');
insert into job values(4,'2020-4-1',1100,'Analyst','Thessaloniki','xorafop','2020-4-7','2020-4-10');
insert into job values(5,'2020-5-1',1600,'HelpDesk','Pyrgos','karela','2020-3-6','2020-5-8');
insert into job values(6,'2020-6-11',2000,'Assistant Manager','Patra','xorafop','2020-6-17','2020-6-27');
insert into job values(7,'2020-7-13',2200,'Assistant Manager','Patra','xristop','2020-7-17','2020-7-21');

insert into job values(9,'2020-2-13',1300,'Assistant Secretary','Pyrgos','papag','2020-3-13','2020-3-15');
insert into job values(10,'2020-2-3',1400,'Assistant Secretary','Thessaloniki','papad','2020-3-3','2020-3-5');
insert into job values(11,'2020-2-23',1200,'Secretary','Patra','papan','2020-3-23','2020-3-25');


insert into project values('arg',1,'Kataskeyi Topikou Diktyou','http:://www.ceid.upatras.gr');
insert into project values('arg',2,'Kataskeyi WAN','http:://www.ceid.upatras.gr');
insert into project values('manolop',1,'Kataskeyi sxesiakou RDBMS','http:://www.ceid.upatras.gr');
insert into project values('manolop',2,'Kataskeyi ierarxikou RDBMS','http:://www.ceid.upatras.gr');
insert into project values('arg',3,'Kataskeyi Topikou Diktyou','http:://www.diktya.gr');
insert into project values('arg',4,'Kataskeyi WAN','http:://www.ceid.upatras.gr');
insert into project values('manolop',5,'Kataskeyi sxesiakou RDBMS','http:://www.ece.upatras.gr');
insert into project values('manolop',6,'Kataskeyi dentrikou RDBMS','http:://www.ece.upatras.gr');
insert into project values('delig',1,'Epektasi Database Etaireias','http:://www.database.gr');
insert into project values('dalak',1,'Epektasi Diktyou Etaireias','http:://www.diktya.gr');



insert into needs values(1,'Networks');
insert into needs values(2,'Networks');
insert into needs values(3,'Local Area Networks');
insert into needs values(3,'Wide Area Networks');
insert into needs values(4,'Local Area Networks');
insert into needs values(5,'Databases');
insert into needs values(5,'SQL Server');

insert into evaluationresult values('','arg',1,5,'Ta pige metria');
insert into evaluationresult values('','manolop',2,9,'Ta pige arista');
insert into evaluationresult values('','giannakoy',3,3,'Ta pige xalia');
insert into evaluationresult values('','dimop',1,10,'Aristi eikona');
insert into evaluationresult values('','delig',2,6,'Ta pige sxerika kala');
insert into evaluationresult values('','dalak',3,7,'Ta pige kala');


insert into degree values('AEI','Panepistimio Patras');
insert into degree values('Master','Panepistimio Patras');
insert into degree values('PHD','Panepistimio Patras');
insert into degree values('AEI','Panepistimio Athinas');
insert into degree values('Master','Panepistimio Athinas');
insert into degree values('PHD','Panepistimio Athinas');


insert into has_degree values('AEI','Panepistimio Athinas','arg',2015,6.5);
insert into has_degree values('Master','Panepistimio Patras','arg',2018,6.5);
insert into has_degree values('AEI','Panepistimio Athinas','manolop',2019,6);
insert into has_degree values('PHD','Panepistimio Athinas','giannakoy',2019,8);
insert into has_degree values('AEI','Panepistimio Athinas','dimop',2020,5);
insert into has_degree values('Master','Panepistimio Patras','delig',2018,7.5);
insert into has_degree values('Master','Panepistimio Patras','dalak',2017,8.5);
insert into has_degree values('AEI','Panepistimio Patras','dalak',2015,5.5);


insert into languages values('arg','EN');
insert into languages values('manolop','GR');
insert into languages values('giannakoy','FR,GR');
insert into languages values('dimop','SP,GR');
insert into languages values('delig','EN,SP');
insert into languages values('dalak','EN,GR');

insert into requestsevaluation values('arg',1);
insert into requestsevaluation values('arg',2);
insert into requestsevaluation values('manolop',2);
insert into requestsevaluation values('manolop',3);
insert into requestsevaluation values('giannakoy',3);
insert into requestsevaluation values('giannakoy',4);
insert into requestsevaluation values('dimop',5);
insert into requestsevaluation values('dimop',6);
insert into requestsevaluation values('delig',1);
insert into requestsevaluation values('delig',6);
insert into requestsevaluation values('delig',3);
insert into requestsevaluation values('dalak',2);
insert into requestsevaluation values('dalak',4);
insert into requestsevaluation values('dalak',5);
insert into requestsevaluation values('arg',3);
insert into requestsevaluation values('giannakoy',11);
insert into requestsevaluation values('giannakoy',13);
insert into requestsevaluation values('giannakoy',23);


insert into ypoboli_aitiseon values('',6,'arg');
insert into ypoboli_aitiseon values('',7,'manolop');
insert into ypoboli_aitiseon values('',6,'arg');
insert into ypoboli_aitiseon values('',7,'manolop');
insert into ypoboli_aitiseon values('',8,'manolop');
insert into ypoboli_aitiseon values('',11,'giannakoy');
insert into ypoboli_aitiseon values('',13,'giannakoy');
insert into ypoboli_aitiseon values('',23,'giannakoy');


insert into evaluation_process values(1,1,3,'Metria Synenetyxi',3,3,'O dieythintis exei metria eikona',3,3,'Exei mono ptyxio kai den exei systatikes oute
diakriseis',null,'karela','arg',null);
insert into evaluation_process values(2,2,3,'Kali Synenetyxi',2,2,'O dieythintis exei thetiki gnomi',2,1,'Exei Master kai 2 systatikes alla oxi diakriseis',1,'karela','arg',null);
insert into evaluation_process values(3,3,4,'Aristi Synenetyxi',3,4,'O dieythintis exei aristi gnomi',3,3,'Exei PHD ptyxio kai polles systatikes kai
evresitexnia',1,'karela','manolop',null);
```

```
insert into evaluation_process values(4,1,4,'Poly Kali Synenetyxi',3,4,'O dieythintis ton proteinei gia tin thesi ayti',3,4,'Exei Master,PHD',null,'karela','manolop',null);
insert into evaluation_process values(5,3,2,null,3,null,'O dieythintis den ton proteinei gia tin thesi',3,4,'Exei Master,PHD',null,'karela','arg',null);
insert into evaluation_process values(6,1,1,'Elleipis Synenetyxi',1,2,'O dieythintis den exei kalh eikona',1,2,'Exei Ptyxio',2,'karela','giannakoy',null);

insert into evaluation_process values(7,1,1,'Kaki Synenetyxi',1,2,'O dieythintis exei metria eikona',1,4,'Polla Ptyxia',3,'papan','giannakoy',null);
insert into evaluation_process values(8,2,2,'Metria Synenetyxi',1,2,'O dieythintis exei metria eikona',1,4,'Polla Ptyxia',2,'papan','giannakoy',null);

insert into evaluation_process values(9,3,2,'kali prosopikoi synenetxyxi',3,3,'O dieythintis einai ikanopoihmeos me tin apodosi tou',4,3,'Exei 2 ptyxia',1,'papan','arg',null);
insert into evaluation_process values(10,3,3,'kali prosopikoi synenetxyxi',3,3,'O dieythintis ton sistynei',4,3,'Exei kalo bathmo sto pytxio',2,'papan','arg',null);



-- ---------------------------------------------------------------------------------------------------------------------
drop procedure if exists proc1;
delimiter $
create procedure proc1(IN ename varchar(25),IN esurname varchar(35))
begin
   declare nf int;
   declare ar_aitisis int;
   declare jid int;
   declare empl varchar(12);
   declare eval varchar(12);
   declare xs text;
   declare xr text;
   declare xl text;
   declare uname varchar(25);
   declare usname varchar(35);
```

```
    declare bcursor cursor for select distinct
evaluation_process.aa,evaluation_process.job_id,evaluation_process.username_emp,evaluation_process.username_eval,evaluation_process.xaraktirismos_synenteyxis,xaraktirismos_report,xaraktirismos_loipon,user.name,user.surname
    from user,employee,evaluation_process,evaluator,requestsevaluation
    where user.username=employee.username and employee.username=requestsevaluation.empl_username and
     employee.username=evaluation_process.username_emp and
    evaluator.username=evaluation_process.username_eval and user.name=ename and user.surname=esurname;

    declare continue handler for not found set nf=1;

    set nf=0;

    open bcursor;

    repeat

            fetch bcursor into ar_aitisis,jid,empl,eval,xs,xr,xl,uname,usname;
            select xs,xr,xl;
            if (nf=0) then

                    if (xs is not null) and (xr is not null) and (xl is not null) then
                            select concat("Completed Evaluation from Evaluator: ",eval," for Employee: ",uname," ",usname," for job: ",jid);
                    else
                            select concat("Evaluation in Progress from Evaluator: ",eval," for Employee: ",uname," ",usname," for job: ",jid);
                    end if;

            end if;
```

```
      until (nf=1)
            end repeat;

end$
delimiter ;

call proc1('Nikos','Argyriadis');
call proc1('Kostas','Manolopoulos');
call proc1('Maria','Giannakou');
-- ------------------------------------------------------------------------------


drop procedure if exists proc2;
delimiter $
create procedure proc2(IN jbdid int,IN evalusr varchar(12))
begin
   declare nf int;
   declare ar_aitisis int;
   declare jid int;
   declare empl varchar(12);
   declare eval varchar(12);
   declare bs int;
   declare br int;
   declare bl int;
   declare uname varchar(25);
   declare usname varchar(35);
   declare final_grade float(3,1);
   declare tp float(3,1);
   declare xs text;
```

```
  declare bcursor cursor for select distinct
evaluation_process.aa,evaluation_process.job_id,evaluation_process.username_emp,evaluation_process.username_eval,evaluation_process.bathmos_synenteyxis,evaluation_process.bathmos_report,evaluation_process.bathmos_loipon,evaluation_process.teliko_apotelesma,evaluation_process.sxolia_synenetyxis
  from user,employee,evaluation_process,evaluator,requestsevaluation
  where user.username=employee.username and employee.username=requestsevaluation.empl_username and
   employee.username=evaluation_process.username_emp and
  evaluation_process.username_eval=evalusr and evaluation_process.job_id=jbdid;

  declare continue handler for not found set nf=1;

  set nf=0;

  open bcursor;

  repeat

        fetch bcursor into ar_aitisis,jid,empl,eval,bs,br,bl,tp,xs;
        select ar_aitisis,jid,empl,eval,bs,br,bl,tp,xs;
        if (nf=0) then
                select bs,br,bl;
                if (bs is not null) and (br is not null) and (bl is not null) and (tp is null) then
                        set final_grade=(bs+br+bl)/3;
                        update evaluation_process set teliko_apotelesma=final_grade where evaluation_process.username_eval=evalusr and evaluation_process.job_id=jbdid
                        and evaluation_process.username_emp=empl;
                        insert into evaluationresult values(",empl,jid,final_grade,xs);
                        select concat("Final Evaluation Grade from Evaluator: ",eval," for Employee with username: ",empl," for job: ",jid," is: ",final_grade);
                else
```

```
                        set final_grade=(bs+br+bl)/3;
                                if (bs is not null) and (br is not null) and (bl is not null) and (tp is not null) then
                                        select concat("Evaluation from Evaluator: ",eval," for Employee with username: ",empl," for job: ",jid," has already grade ",final_grade);
                                end if;
                        end if;

                end if;

        until (nf=1)
                end repeat;

end$
delimiter ;

call proc2(1,'karela');
call proc2(2,'karela');
call proc2(1,'papan');
call proc2(2,'papan');
call proc2(3,'papan');
-- -----------------------------------------------------------------
drop procedure if exists proc3;
delimiter $
create procedure proc3(IN jbdid int)
begin
   declare nf int;

   declare ar_aitisis int;
   declare jid int;
```

```
  declare empl varchar(12);
  declare eval varchar(12);
  declare bs int;
  declare br int;
  declare bl int;
  declare uname varchar(25);
  declare usname varchar(35);
  declare final_grade float(3,1);
  declare ta float(3,1);
  declare counter int;

  declare bcursor cursor for select distinct
evaluation_process.aa,evaluation_process.job_id,evaluation_process.username_emp,evaluation_process.username_eval,evaluation_process.teliko_apotelesma
  from user,employee,evaluation_process,evaluator,requestsevaluation
  where user.username=employee.username and employee.username=requestsevaluation.empl_username and
   employee.username=evaluation_process.username_emp and evaluation_process.job_id=jbdid;

   declare bcursor2 cursor for select distinct
evaluation_process.aa,evaluation_process.job_id,evaluation_process.username_emp,evaluation_process.username_eval,evaluation_process.teliko_apotelesma
                                from user,employee,evaluation_process,evaluator,requestsevaluation
                                where user.username=employee.username and employee.username=requestsevaluation.empl_username and
                                employee.username=evaluation_process.username_emp and
                                evaluation_process.job_id=jbdid
                                order by evaluation_process.teliko_apotelesma desc;



        declare bcursor3 cursor for select distinct
evaluation_process.aa,evaluation_process.job_id,evaluation_process.username_emp,evaluation_process.username_eval,evaluation_process.teliko_apotelesma
```

```
                                from user,employee,evaluation_process,evaluator,requestsevaluation
                                where user.username=employee.username and employee.username=requestsevaluation.empl_username and
                                employee.username=evaluation_process.username_emp and   evaluation_process.job_id=jbdid;


 declare continue handler for not found set nf=1;



set nf=0;
set counter=0;

open bcursor;

repeat
     fetch bcursor into ar_aitisis,jid,empl,eval,ta;

     if (nf=0) then
            if (ta is null) then
               set counter=counter+1;
            end if;
     end if;
  until (nf=1)
end repeat;

     select 'Counter =',counter;

     if (counter=0) then
            select 'ORISTIKOPOIHMENOI PINAKES';
            set nf=0;
            open bcursor2;

            repeat
                    fetch bcursor2 into ar_aitisis,jid,empl,eval,ta;

                    if (nf=0) then
                            if (ta is not null) then
                                    select ar_aitisis,jid,empl,eval,ta;
                            end if;
                    end if;

            until (nf=1)
            end repeat;
    end if;



if (counter<>0) then
            select 'AXIOLOGHSH SE EXELIXI. EKKREMOUN AKOMA: ',counter;

            set nf=0;
            open bcursor3;

            repeat
                    fetch bcursor3 into ar_aitisis,jid,empl,eval,ta;
                    select ar_aitisis,jid,empl,eval,ta;
                    select '--------------------------------';
            if (nf=0) then
                if (ta is not null) then
                    select ar_aitisis,jid,empl,eval,ta;
                end if;
            end if;

        until (nf=1)
        end repeat;
    end if;

end$
delimiter ;

call proc3(1);
call proc3(2);
call proc3(3);
-- ==============

drop procedure if exists proc3b;
delimiter $
create procedure proc3b(IN jbdid int,OUT ar_aitisis int,OUT jid int,OUT empl varchar(12),OUT eval varchar(12),OUT ta float(3,1))
begin
   declare nf int;

   -- declare ar_aitisis int;
   -- declare jid int;
   -- declare empl varchar(12);
   -- declare eval varchar(12);
```

```
  declare bs int;
  declare br int;
  declare bl int;
  declare uname varchar(25);
  declare usname varchar(35);
  declare final_grade float(3,1);
  -- declare ta float(3,1);
  declare counter int;

  declare bcursor cursor for select distinct
evaluation_process.aa,evaluation_process.job_id,evaluation_process.username_emp,evaluation_process.username_eval,evaluation_process.teliko_apotelesma
  from user,employee,evaluation_process,evaluator,requestsevaluation
  where user.username=employee.username and employee.username=requestsevaluation.empl_username and
   employee.username=evaluation_process.username_emp and evaluation_process.job_id=jbdid;

   declare bcursor2 cursor for select distinct
evaluation_process.aa,evaluation_process.job_id,evaluation_process.username_emp,evaluation_process.username_eval,evaluation_process.teliko_apotelesma
                                 from user,employee,evaluation_process,evaluator,requestsevaluation
                                 where user.username=employee.username and employee.username=requestsevaluation.empl_username and
                                 employee.username=evaluation_process.username_emp and
                                 evaluation_process.job_id=jbdid
                                 order by evaluation_process.teliko_apotelesma desc;



        declare bcursor3 cursor for select distinct
evaluation_process.aa,evaluation_process.job_id,evaluation_process.username_emp,evaluation_process.username_eval,evaluation_process.teliko_apotelesma
                                 from user,employee,evaluation_process,evaluator,requestsevaluation
                                 where user.username=employee.username and employee.username=requestsevaluation.empl_username and
```

```
                                        employee.username=evaluation_process.username_emp and   evaluation_process.job_id=jbdid;



 declare continue handler for not found set nf=1;



set nf=0;
set counter=0;

open bcursor;

repeat
      fetch bcursor into ar_aitisis,jid,empl,eval,ta;

      if (nf=0) then
              if (ta is null) then
                  set counter=counter+1;
              end if;
      end if;
  until (nf=1)
end repeat;

      select 'Counter =',counter;

      if (counter=0) then
              select 'ORISTIKOPOIHMENOI PINAKES';

              set nf=0;
              open bcursor2;

              repeat
                  fetch bcursor2 into ar_aitisis,jid,empl,eval,ta;

                  if (nf=0) then
                      if (ta is not null) then
                          select ar_aitisis,jid,empl,eval,ta;
                      end if;
                  end if;

              until (nf=1)
              end repeat;
      end if;



  if (counter<>0) then
              select 'AXIOLOGHSH SE EXELIXI. EKKREMOUN AKOMA: ',counter;

              set nf=0;
              open bcursor3;

              repeat
                  fetch bcursor3 into ar_aitisis,jid,empl,eval,ta;
                  select ar_aitisis,jid,empl,eval,ta;
                  select '--------------------------------';
                  if (nf=0) then
                      if (ta is not null) then
```

```
                                        select ar_aitisis,jid,empl,eval,ta;
                                end if;
                        end if;

                until (nf=1)
                end repeat;
        end if;

end$
delimiter ;




-- ====================
drop trigger if exists jobtrig1;
delimiter $
create trigger jobtrig1 before insert on job
for each row
begin
        DECLARE jid int;
        SET jid = (SELECT job.id FROM job WHERE job.id = NEW.id);

    IF jid IS NOT NULL THEN
        insert into log values(NEW.evaluator,now(),'Failure','insert','job');
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'INSERT on JOB failed due to duplicate JOB ID';
        ELSE
                insert into log values(NEW.evaluator,now(),'Success','insert','job');
    END IF;
end$
delimiter ;

-- test
insert into job values(8,'2020-7-30',2200,'Main Manager','Patra','xristop','2020-8-1','2020-8-2');
insert into job values(8,'2020-7-31',2200,'Second Main Manager','Patra','karela','2020-8-2','2020-8-3');
-- ---------------------------------------
drop trigger if exists jobtrig2;
delimiter $
create trigger jobtrig2 before update on job
for each row
begin
    IF new.id=old.id and new.start_date=old.start_date and new.salary=old.salary and new.position=old.position and new.edra=old.edra and new.evaluator=old.evaluator
        and new.announce_date=old.announce_date and new.submission_date=old.submission_date then
        insert into log values(NEW.evaluator,now(),'Failure','update','job');
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on JOB failed due to duplicate Elements';
        ELSE
                insert into log values(NEW.evaluator,now(),'Success','update','job');
    END IF;

        IF (old.id <> new.id) THEN
                insert into log values(NEW.evaluator,now(),'Failure','update','Job');
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on Jib failed due to EXISTING JOB ID';
        end if;
end$
delimiter ;
```

```
-- test
update job set start_date='2020-07-21',salary=2200.0,position='Main Manager',edra='Patra',evaluator='xristop',announce_date='2020-08-01 00:00:00',submission_date='2020-08-02' where id=1;
update job set id=1 where id=1;

-- ---------------------------------------------------------------
drop trigger if exists jobtrig3;
delimiter $
create trigger jobtrig3 before delete on job
for each row
begin
    IF (EXISTS(SELECT count(*) FROM job inner join requestsevaluation on id=job_id)) THEN
        insert into log values(OLD.evaluator,now(),'Failure','delete','job');
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'DELETE on JOB failed due to EXISTENT REQUESTS EVALUTTION FOR THIS JOB';
        else
                insert into log values(old.evaluator,now(),'Success','delete','job');
        end if;

        IF (EXISTS(SELECT count(*) FROM job inner join evaluationresult on id=job_id)) THEN
        insert into log values(OLD.evaluator,now(),'Failure','delete','job');
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'DELETE on JOB failed due to EXISTENT EVALUTTION RESULTS FOR THIS JOB';
        else
                insert into log values(old.evaluator,now(),'Success','delete','job');
        end if;
end$
delimiter ;

-- test
```

```
delete from job where id=4;




-----------------------------------------------------
drop trigger if exists employeetrig1;
delimiter $
create trigger employeetrig1 before insert on employee
for each row
begin
        DECLARE jeuname varchar(12);
        SET jeuname = (SELECT employee.username FROM employee WHERE employee.username   = NEW.username);

    IF jeuname IS NOT NULL THEN
        insert into log values(NEW.username,now(),'Failure','insert','employee');
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'INSERT on Employee failed due to duplicate EMPLOYEE USERNAME';
        ELSE
                insert into log values(NEW.username,now(),'failure','insert','Empoyee');
    END IF;
end$
delimiter ;

--test
insert into employee values('basil','Einai kalos ypallilos','Polles systatikes','basil_certificates.pdf','basil_awards.pdf');
insert into employee values('dimop','Einai metrios ypallilos','Liges systatikes','dimop_certificates.pdf','dimop_awards.pdf');

-- ----------------------------------------
drop trigger if exists employeetrig2;
```

```
delimiter $
create trigger employeetrig2 before update on employee
for each row
begin
    IF new.username=old.username and new.bio=old.bio and new.sistatikes=old.sistatikes and new.certificates=old.certificates and new.awards=old.awards then
        insert into log values(NEW.username,now(),'Failure','update','Employee');
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on Empoyee failed due to duplicate Elements';
        ELSE
                insert into log values(NEW.username,now(),'Success','update','Employee');
    END IF;
        IF (old.username <> new.username) THEN
                insert into log values(NEW.username,now(),'Failure','update','Employee');
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on Employee failed due to EXISTING USERNAME';
        end if;
end$
delimiter ;

-- test
update employee set sistatikes='None' where username='basil';
update employee set username='basilis' where username='basil';

-- ---------------------------------------------------------------
drop trigger if exists employeetrig3;
delimiter $
create trigger employeetrig3 before delete on employee
for each row
begin
    IF (EXISTS(SELECT count(*) FROM employee inner join requestsevaluation on username=empl_username)) THEN
```

```
        insert into log values(OLD.username,now(),'Failure','delete','employee');
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'DELETION FROM EMPLOYEE failed due to EXISTING EVALUATION REQUESTS
FOR THIS EMPLOYEE';
        else
                insert into log values(old.username,now(),'Success','delete','employee');
        end if;
end$
delimiter ;

-- test
delete from employee where username='dalak';

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

drop trigger if exists requestsevaluationrig1;
delimiter $
create trigger requestsevaluationrig1 before insert on requestsevaluation
for each row
begin
        DECLARE jeuname varchar(12);
        SET jeuname = (SELECT requestsevaluation.empl_username FROM requestsevaluation WHERE requestsevaluation.empl_username=NEW.empl_username and
requestsevaluation.job_id=new.job_id);

    IF jeuname IS NOT NULL THEN
        insert into log values(NEW.empl_username,now(),'Failure','insert','requestsevaluation');
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'INSERT on requestsevaluation failed due to duplicate EMPLOYEE USERNAME and JOB
ID';
        ELSE
```

```
                insert into log values(NEW.empl_username,now(),'failure','insert','requestsevaluation');
    END IF;
end$
delimiter ;
-- test
insert into requestsevaluation values('dimop',5);

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


drop trigger if exists requestsevaluationrig2;
delimiter $
create trigger requestsevaluationrig2 before update on requestsevaluation
for each row
begin
        IF (old.empl_username = new.empl_username and old.job_id=new.job_id) THEN
                insert into log values(NEW.empl_username,now(),'Failure','Update','requestsevaluation');
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on requestsevaluation failed due to duplicate EMPLOYEE USERNAME and JOB ID';
        ELSE
                insert into log values(NEW.empl_username,now(),'failure','update','requestsevaluation');
    END IF;
end$
delimiter ;

-- test
update requestsevaluation set empl_username='dalak',job_id=2 where empl_username='dalak' and job_id=2;
 -- ------------------------
```

```
drop trigger if exists requestsevaluationrig3;
delimiter $
create trigger requestsevaluationrig3 before delete on requestsevaluation
for each row
begin
    IF (EXISTS(SELECT count(*) FROM requestsevaluation inner join employee on username=empl_username group by empl_username)) THEN
        insert into log values(OLD.empl_username,now(),'Failure','delete','requestsevaluation');
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'DELETION FROM requestsevaluation failed due to EXISTING EMPLOYEE WHO REQUESTS FOR EVALUATION';
        else
                insert into log values(old.empl_username,now(),'Success','delete','requestsevaluation');
        end if;
end$
delimiter ;

-- test
DELETE FROM requestsevaluation WHERE empl_username='dalak';

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


drop trigger if exists companytrig;
delimiter $
create trigger companytrig before update on company
for each row
begin
        IF (old.AFM <> new.AFM) THEN
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on Compnay failed due to Company AFM cannot be changed';
    END IF;
```

```
        IF (old.DOY <> new.DOY) THEN
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on Compnay failed due to Company DOY cannot be changed';
    END IF;

        IF (old.name <> new.name) THEN
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on Compnay failed due to Company NAME cannot be changed';
    END IF;
end$
delimiter ;

-- test
update company set AFM='111111111' where AFM='123456123';
update company set DOY='C PATRAS'   where AFM='123456123';
update company set name='Kataskeyastiki OE PATRAS' where AFM='123456123';
update company set phone='2610324567' where AFM='123456123';
update company set street='Agias',num=100,city='Pirgos' where AFM='123456123';
-- ----------------------------------------------------------------------------



drop trigger if exists useradmintrig;
delimiter $
create trigger useradmintrig before update on user
for each row
begin
        IF (old.username <> new.username and old.username<>'admin') THEN
                SIGNAL SQLSTATE VALUE '45000' SET MESSAGE_TEXT = 'UPDATE on User failed due to User Username cannot be changed';
    END IF;
```

```
end$
delimiter ;

-- test
update user set username='maria' where username='dalak';
update user set username='maria' where username='admin';



-- -----------------
create table tomeas(kt int not null,decription varchar(50),AFM char(9),primary key(kt), constraint wz foreign key(AFM) references company(AFM));
insert into tomeas values(1,'Kataskeyazei logistika kai emporika paketa','123456123');
insert into tomeas values(2,'Kataskeyazei paketa sxediasis gia mixanikous','123456123');
```