PROJECT ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

TRAVEL_AGENCY

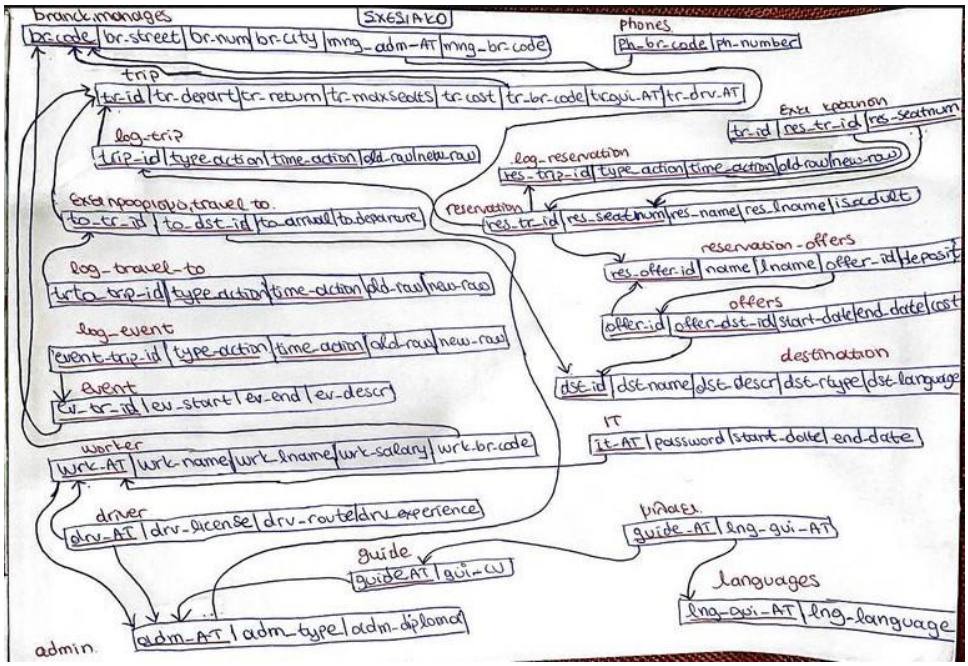
Μαριάνθη Θώδη ΑΜ:1084576

Ελπίδα Κόκκαλη ΑΜ: 1084648

3ο έτος

Α' μέρος :

Κεφάλαιο 1



- **<u>table it</u> : Είναι ένας πίνακας που θα περιλαμβάνει όλους τους διαχειριστές της βάσης δεδομένων travel_agency. Περιέχει το αφμ και τον κωδικό πρόσβασης κάθε διαχειριστή που με αυτά θα μπορεί να συνδεθεί ο εκάστοτε στην διεπαφή GUI . Επίσης περιέχει την ημερομηνία έναρξης ,δηλαδή πότε ξεκίνησε ένας worker να είναι διαχειριστής της βάσης καθώς και την ημερομηνία που σταμάτησε να επιτελεί έργα διαχειριστή .Αν αυτό το πεδίο είναι NULL ,αυτό σημαίνει ότι ακόμα επιτελεί έργο διαχειριστή.**

- **<u>table offers</u> : Είναι ένας πίνακας που περιλαμβάνει κάποιες ειδικές προσφορές ,που σχετίζονται με συγκεκριμένο προορισμό ,που μπορούν να πραγματοποιηθούν κατόπιν κράτησης , σε συγκεκριμένο χρονικό διάστημα .Το offer_id αναφέρεται σε ένα συγκεκριμένο προορισμό από τον πίνακα destination.**

- **<u>table reservation_offers:</u> Είναι ένας πίνακας που περιλαμβάνει τις κρατήσεις για κάθε προσφορά για έναν συγκεκριμένο προορισμό και περιέχει τα στοιχεία κάθε πελάτη για την κράτηση.**

- **<u>table log_trip/ log_event/log_reservation/log_travel_to</u> : Είναι πίνακες που όταν ένα από τα αντίστοιχα triggers(update/insert/delete) ενεργοποιηθεί ,θα καταγράφουν τις ενέργειες που πραγματοποιήθηκαν από τον διαχειριστή .**

– **Λεπτομέρειες:**

1.

```
import random //κώδικας στην python για να κάνω τις 60.000 εγγραφές ,όπου letssee.txt το αρχείο με τα 1000 ονόματα

file=open('letssee.txt','r')

data=file.readlines()

firstnames=[]

lastnames=[]


i=0

for line in data:
   names=line.strip('\n').split(',')
   for string in names:
      x=random.randint(50, 200)
      name=string.split('\t')[::-1]
      for c in name:
```

```
        firstname=string.split('\t')[0]
        firstnames.append(firstname)
        lastname=string.split('\t')[-1]
        lastnames.append(lastname)
        break
    print(f'INSERT INTO reservation_offers(name,lname,offer_id,deposit) VALUES
("{firstnames[i]}","{lastnames[i]}",1or2or3****, {x} );')
    i=i+1
```

*******1 για τις πρωτες 20.000 εγραφες ,2 για τις επομενες 20.000 και 3 για τις επομενες 20.000**

2. Στον πίνακα admin ,οι 20 πρώτες εγγραφές είναι πρόσθετες και αναφέρονται σε διοικητικούς υπαλλήλους που έχουν αναλάβει την διοίκηση ενός υποκαταστήματος.

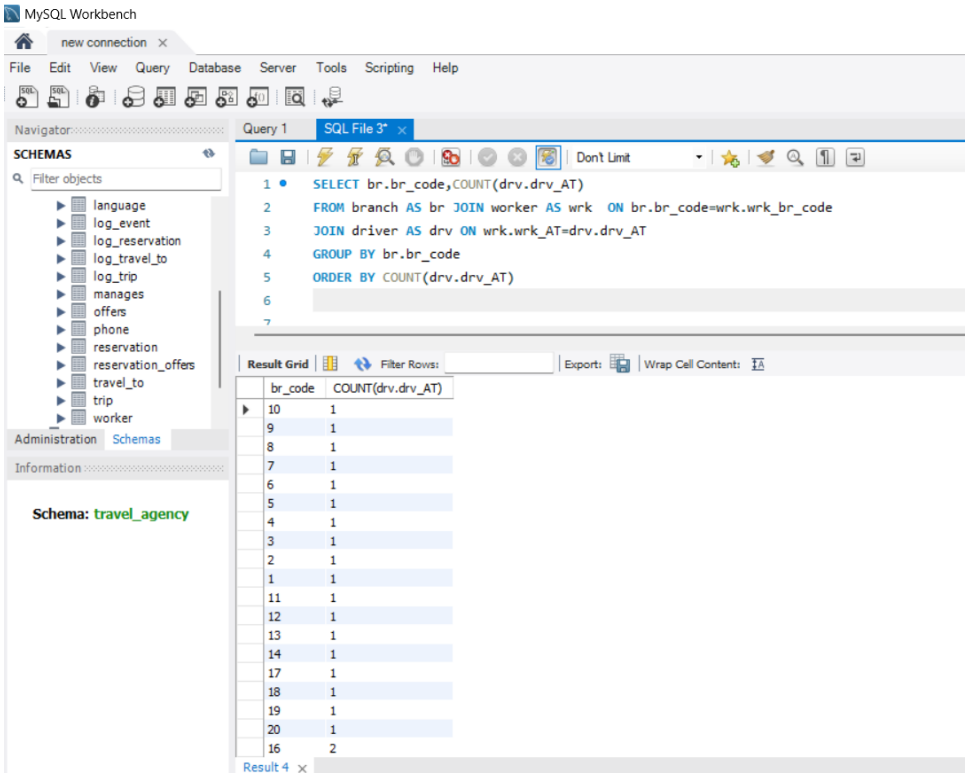
Κεφάλαιο 2

STORE PROCEDURES:

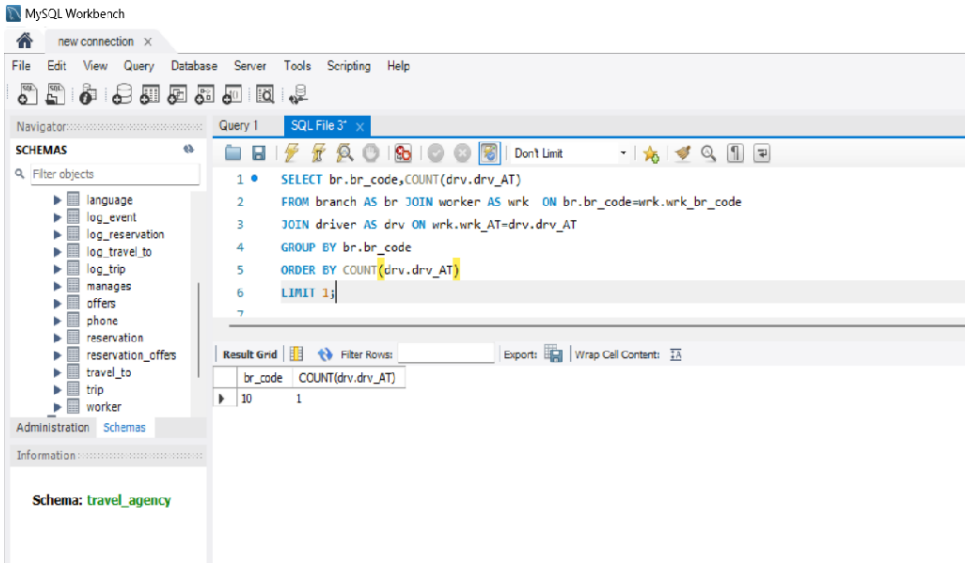
3.1.3.1

```
DELIMITER $
CREATE PROCEDURE new_driver(IN driver_name VARCHAR(20), IN driver_lname
VARCHAR(20), IN driver_salary FLOAT(7,2),
IN driver_AT VARCHAR(10),IN driver_license ENUM('A','B','C','D'),
IN driver_route ENUM('LOCAL', 'ABROAD'),IN driver_experience TINYINT)
BEGIN
DECLARE branch_id INT;
DECLARE sum_drivers INT ;
```

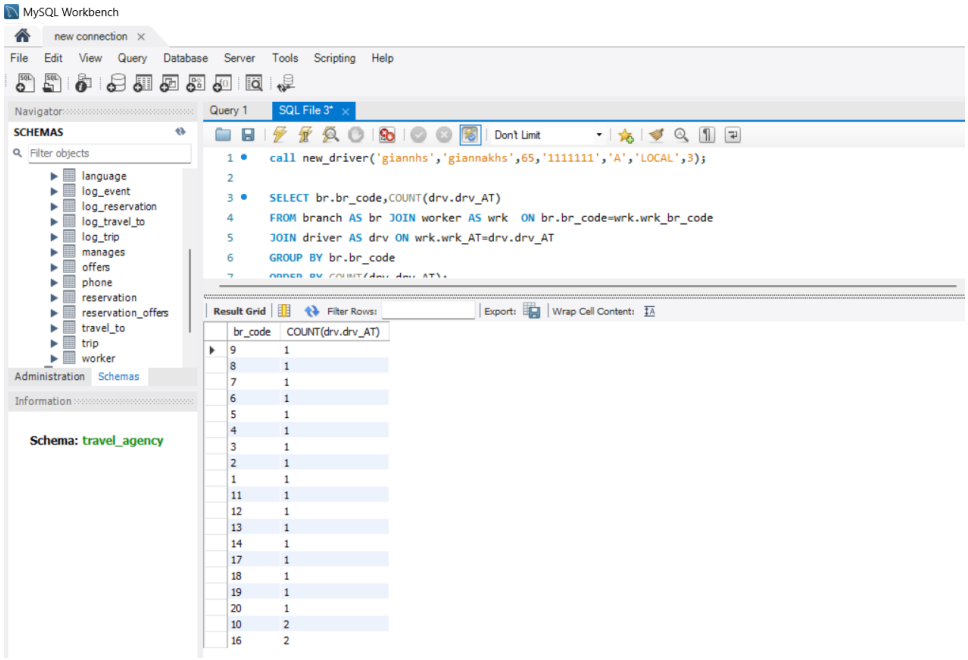
```
SELECT br.br_code,COUNT(drv.drv_AT)
INTO branch_id,sum_drivers
FROM branch AS br JOIN worker AS wrk  ON br.br_code=wrk.wrk_br_code
JOIN driver AS drv ON wrk.wrk_AT=drv.drv_AT
GROUP BY br.br_code
ORDER BY COUNT(drv.drv_AT)
LIMIT 1;

INSERT INTO worker VALUES(driver_AT,driver_name,driver_lname,driver_salary,branch_id);
INSERT INTO driver VALUES(driver_AT,driver_license,driver_route,driver_experience);
END$
DELIMITER;
```

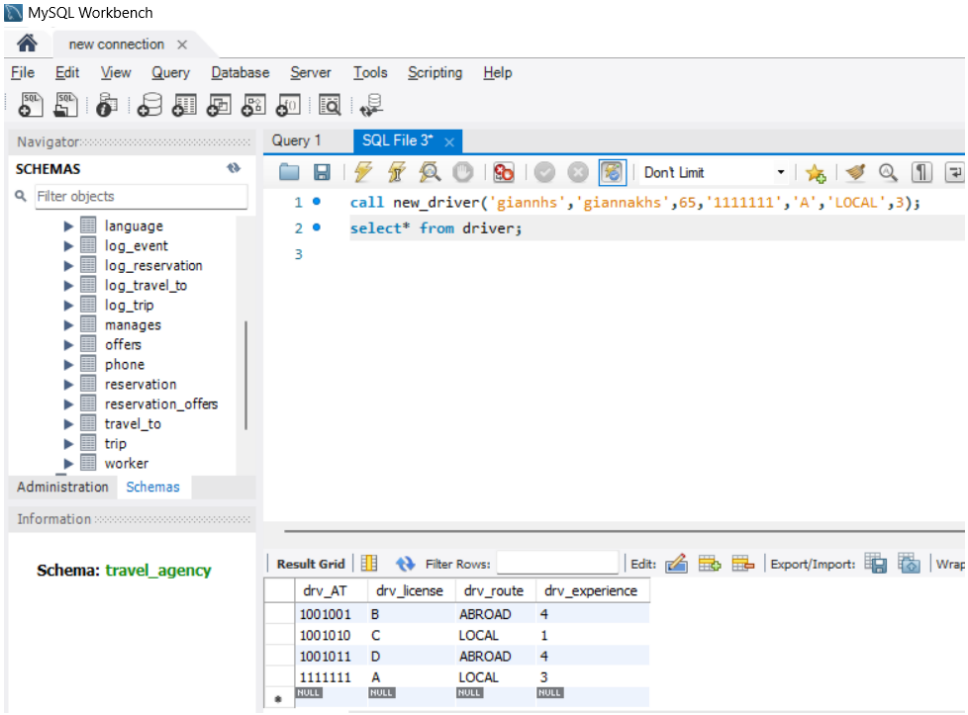


//Επιλέγει το πρώτο υποκατάστημα με τους λιγότερους οδηγούς



//Εγινε η εισαγωγή





3.1.3.3

DELIMITER $

CREATE PROCEDURE delete_worker(IN name_worker VARCHAR(20), IN lname_worker VARCHAR(20))

BEGIN

DECLARE admtype VARCHAR(20);

DECLARE keyadm VARCHAR(10);

DECLARE result VARCHAR(255);

SELECT adm.adm_type,adm.adm_AT

INTO admtype,keyadm

FROM worker AS wrk JOIN admin AS adm ON wrk.wrk_AT=adm.adm_AT

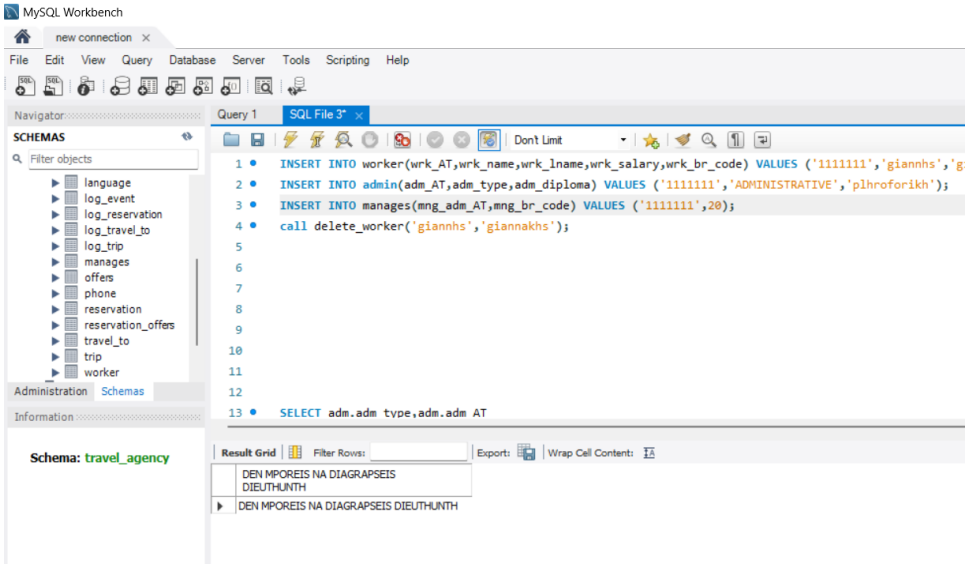
WHERE wrk.wrk_name=name_worker AND wrk.wrk_lname=lname_worker;

SELECT IF(EXISTS (SELECT adm.adm_AT

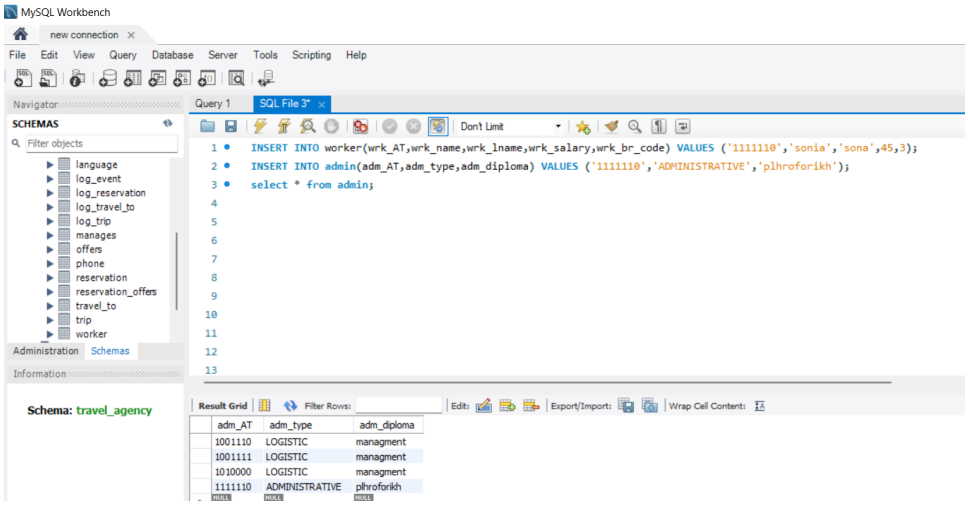
FROM admin AS adm JOIN manages AS mn ON adm.adm_AT=mn.mng_adm_AT

```
WHERE mn.mng_adm_AT=keyadm),'O SUGKEKRIMENOS UPALLHLOS EINAI
DIEUTHUNTHS','O SUGKEKRIMENOS UPALLHLOS DEN EINAI DIEUTHUNTHS') INTO result;


IF (admtype='ADMINISTRATIVE' AND result='O SUGKEKRIMENOS UPALLHLOS EINAI
DIEUTHUNTHS') THEN

SELECT 'DEN MPOREIS NA DIAGRAPSEIS DIEUTHUNTH';

ELSE

DELETE FROM worker WHERE wrk_AT=keyadm ;

SELECT 'DIAGRAFTHKE O UPALLHLOS' ;

END IF;

END$

DELIMITER;
```

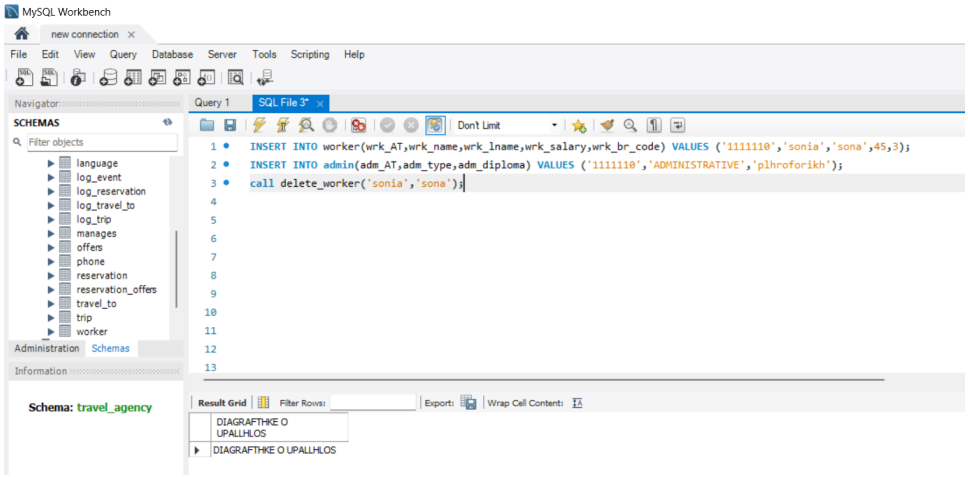
//Εστω ότι προσθέτω έναν υπάλληλο που είναι και διοικητικός και είναι διευθυντής του υποκαταστήματος με κωδικό br_code=20. Όταν πάω να τον διαγράψω ,εμφανίζεται κατάλληλο μήνυμα.



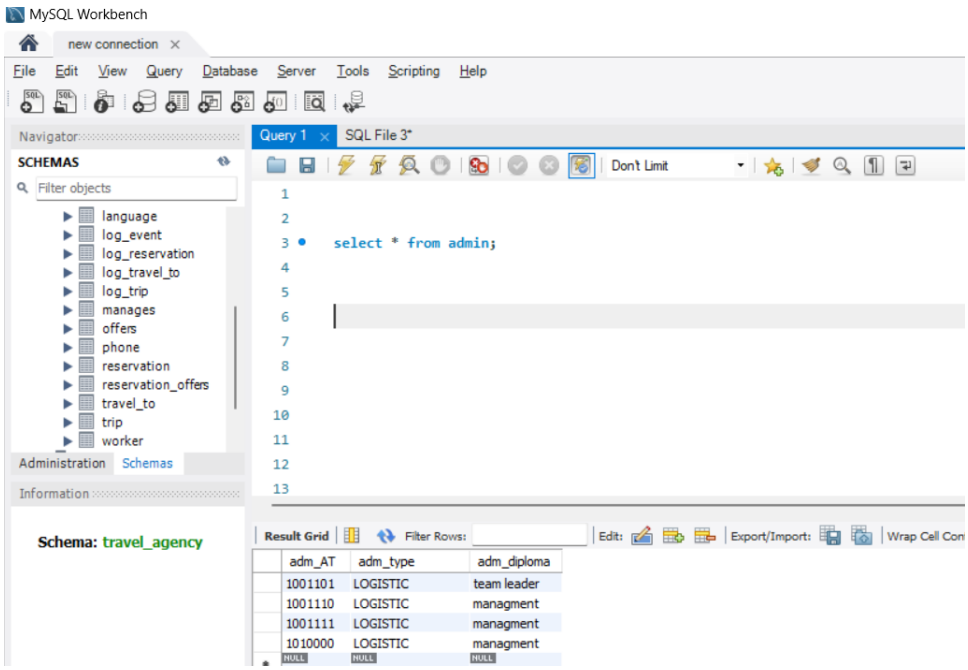
//Εστω ότι προσθέτω απλά μία διοικητική υπάλληλο.



//διαγράφτηκε η υπάλληλος ,εφόσον δεν ήταν διευθύντρια



//Επαλήθευση



3.1.3.2

//Ο χρήστης ψάχνει να βρει διαθέσιμα ταξίδια στις ημερομηνίες που βάζουμε σαν ορίσματα καθώς και τις υπόλοιπες πληροφορίες για τα ταξίδια αυτά.

//Έκανα μία πρόσθετη εγγραφή στο υποκατάστημα με br_code=1 ώστε ένα υποκατάστημα να διοργανώνει 2 ταξίδια

INSERT INTO destination(dst_id,dst_name,dst_descr,dst_rtype,dst_language,dst_location)

VALUES(20,'hawaii','xalarwtikes diakopes','ABROAD','agglika',20);

INSERT INTO
trip(tr_departure,tr_return,tr_maxseats,tr_cost,tr_br_code,tr_gui_AT,tr_drv_AT)

VALUES('2023-05-05 09:00:00','2023-10-05 18:00:00',10,500,1,'0100100','0111000');

INSERT INTO travel_to(to_tr_id,to_dst_id,to_arrival,to_departure)

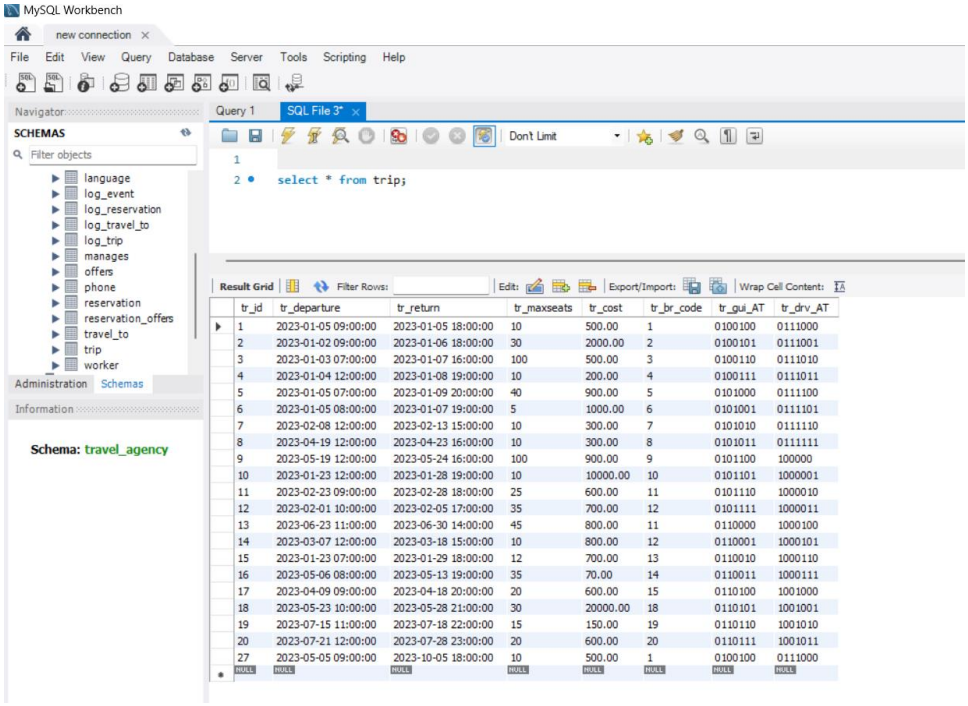
VALUES (21,20,'2023-01-01 17:00:00','2023-01-05 19:00:00');

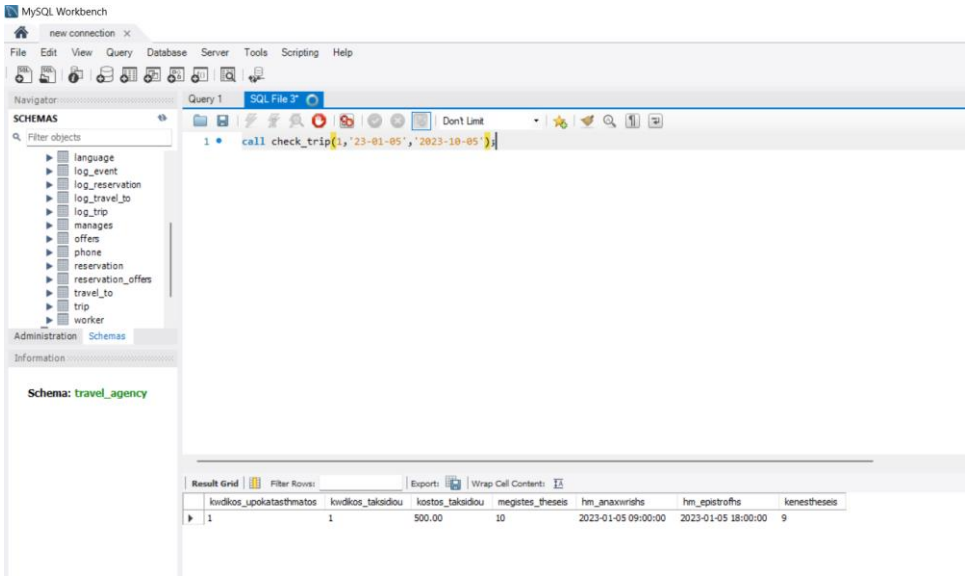
```
DELIMITER $
CREATE PROCEDURE check_trip( IN brcode INT , IN date1 DATE, IN date2 DATE)
BEGIN
DECLARE flag INT;
DECLARE trid INT;

DECLARE cr CURSOR FOR
SELECT tr.tr_id
FROM branch AS br JOIN trip AS tr ON br.br_code=tr.tr_br_code
WHERE br_code=brcode AND DATE(tr.tr_departure)  BETWEEN date1  AND date2;

DECLARE CONTINUE HANDLER FOR NOT FOUND SET flag=1;
OPEN cr;
```

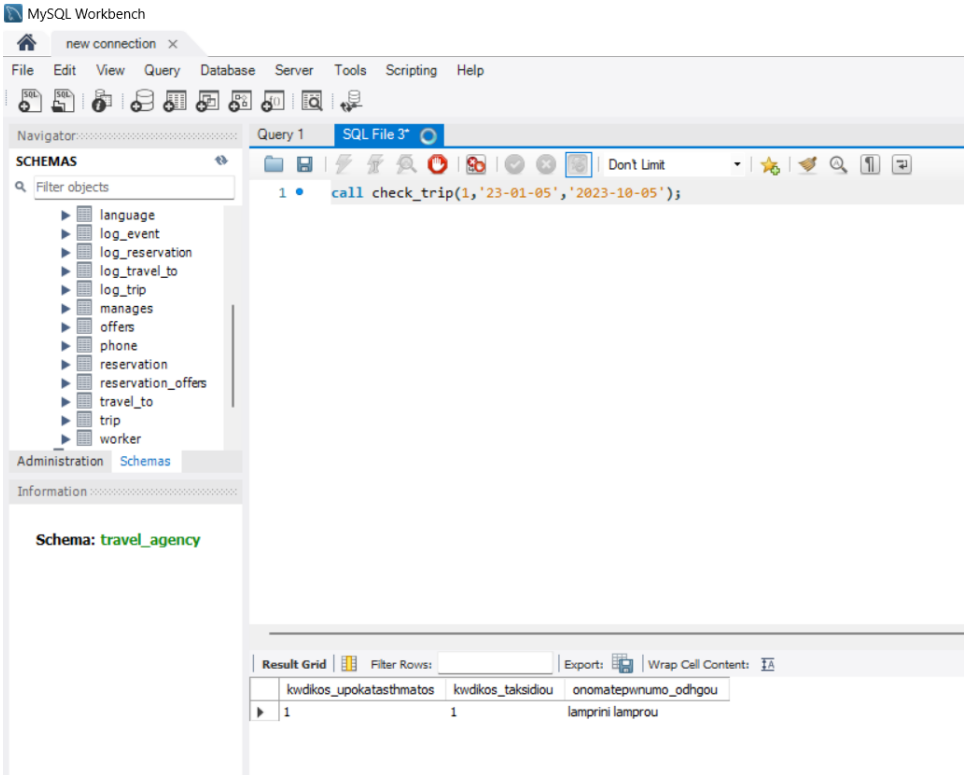
```
SET flag=0;

WHILE(flag=0) DO

FETCH cr INTO trid;

SELECT  brcode AS kwdikos_upokatasthmatos,tr.tr_id AS kwdikos_taksidiou,tr.tr_cost AS kostos_taksidiou  ,tr.tr_maxseats AS megistes_theseis,tr.tr_departure AS hm_anaxwrishs ,tr.tr_return AS hm_epistrofhs,(tr.tr_maxseats-COUNT(res.res_tr_id)) AS kenestheseis

FROM trip AS tr JOIN reservation AS res ON tr.tr_id=res.res_tr_id

WHERE tr.tr_id=trid

GROUP BY res.res_tr_id;

DO SLEEP(7);

SELECT  brcode AS kwdikos_upokatasthmatos,tr.tr_id AS kwdikos_taksidiou,CONCAT(wrk_name ,' ',wrk.wrk_lname) AS onomatepwnumo_odhgou

FROM trip AS tr JOIN driver AS drv ON tr.tr_drv_AT=drv.drv_AT JOIN worker AS wrk ON drv.drv_AT=wrk.wrk_AT

WHERE tr.tr_id=trid;

DO SLEEP(7);

SELECT  brcode AS kwdikos_upokatasthmatos,tr.tr_id AS kwdikos_taksidiou,CONCAT(wrk.wrk_name,' ',wrk.wrk_lname) AS onomatepwnumo_ksenagou

FROM trip AS tr JOIN guide AS gui ON tr.tr_gui_AT=gui.gui_AT JOIN worker AS wrk ON gui.gui_AT=wrk.wrk_AT

WHERE tr.tr_id=trid;

DO SLEEP(7);

END WHILE;

CLOSE cr;

END$

DELIMITER;
```

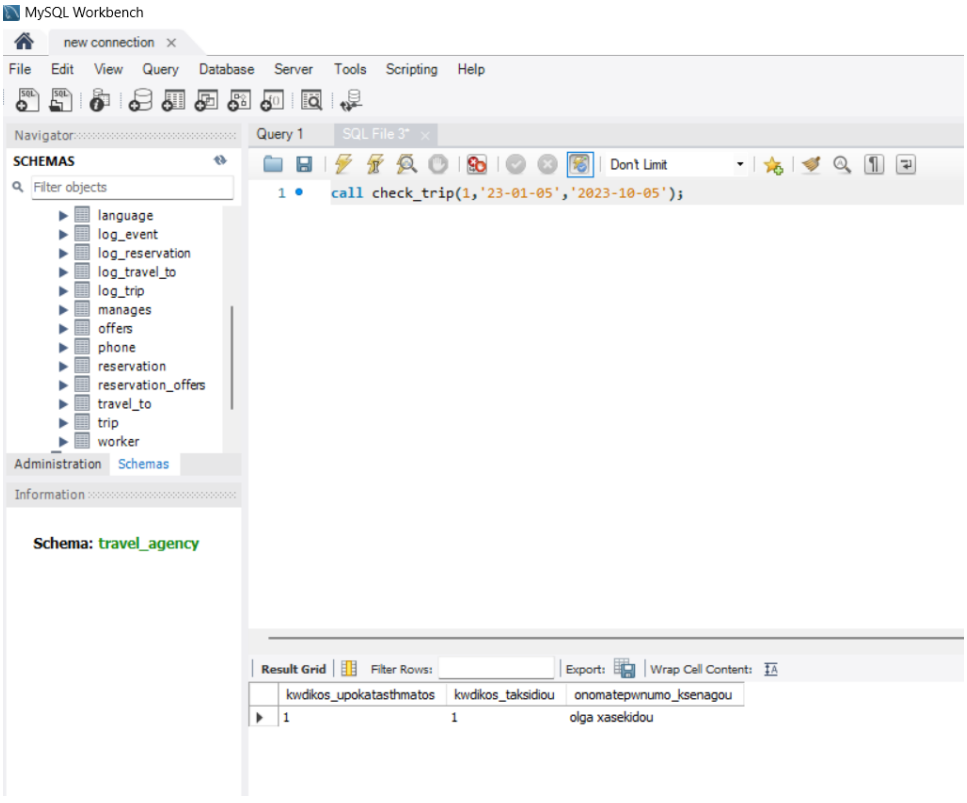
//tr_id=1 ,tr_id=27 ανήκουν στο υποκατάστημα με br_code=1

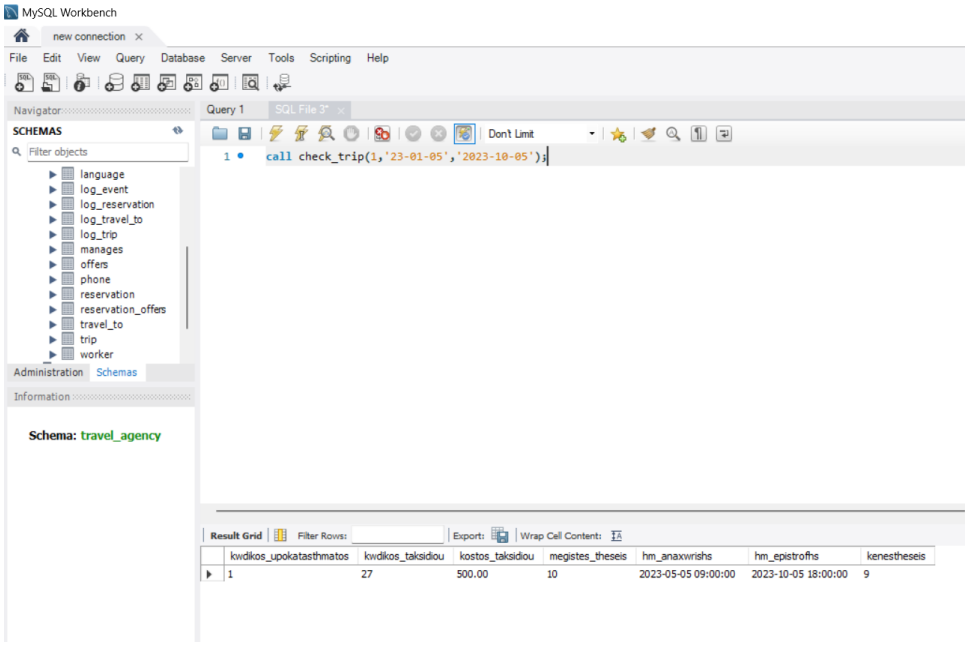
//ο cursor εκτυπώνει τα αποτελέσματα και έχω βάλει τον ίδιο οδηγό και ξεναγό και στα δύο ταξίδια

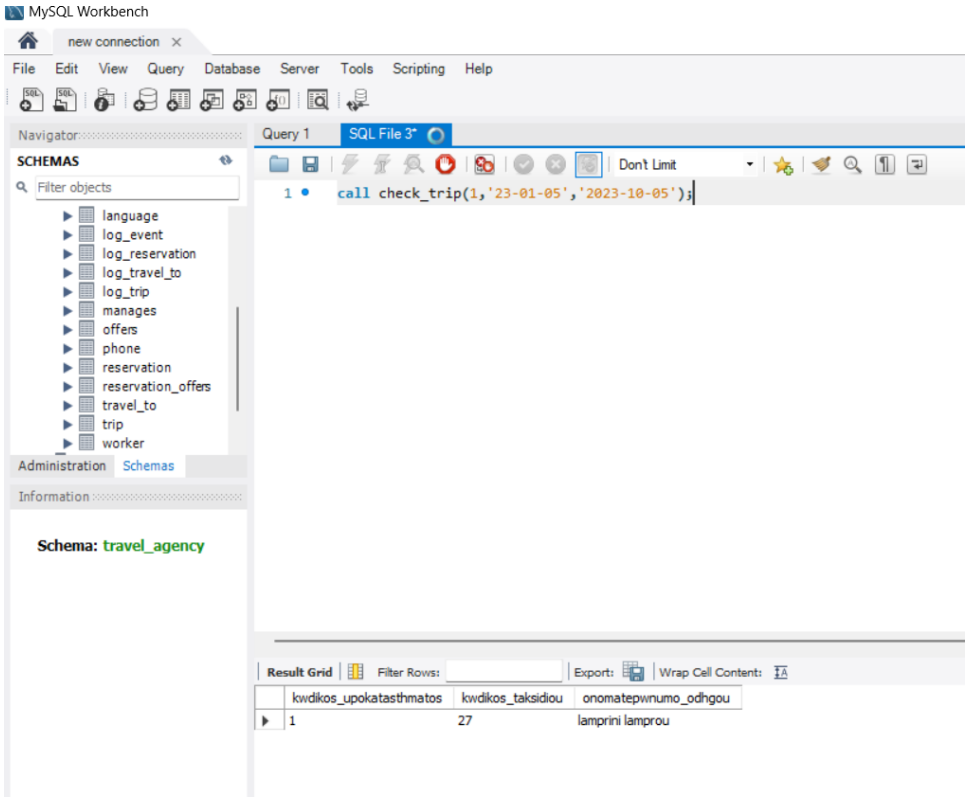

MySQL Workbench
new connection
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
language
log_event
log_reservation
log_travel_to
log_trip
manages
offers
phone
reservation
reservation_offers
travel_to
trip
worker
Administration Schemas
Information
Schema: travel_agency
Query 1
SQL File 3*
Don't Limit
1
2 • select * from trip;
Result Grid
Filter Rows:
Edit:
Export/Import:
Wrap Cell Content:
tr_id tr_departure tr_return tr_maxseats tr_cost tr_br_code tr_gui_AT tr_drv_AT
1 2023-01-05 09:00:00 2023-01-05 18:00:00 10 500.00 1 0100100 0111000
2 2023-01-02 09:00:00 2023-01-06 18:00:00 30 2000.00 2 0100101 0111001
3 2023-01-03 07:00:00 2023-01-07 16:00:00 100 500.00 3 0100110 0111010
4 2023-01-04 12:00:00 2023-01-08 19:00:00 10 200.00 4 0100111 0111011
5 2023-01-05 07:00:00 2023-01-09 20:00:00 40 900.00 5 0101000 0111100
6 2023-01-05 08:00:00 2023-01-07 19:00:00 5 1000.00 6 0101001 0111101
7 2023-02-08 12:00:00 2023-02-13 15:00:00 10 300.00 7 0101010 0111110
8 2023-04-19 12:00:00 2023-04-23 16:00:00 10 300.00 8 0101011 0111111
9 2023-05-19 12:00:00 2023-05-24 16:00:00 100 900.00 9 0101100 100000
10 2023-01-23 12:00:00 2023-01-28 19:00:00 10 10000.00 10 0101101 1000001
11 2023-02-23 09:00:00 2023-02-28 18:00:00 25 600.00 11 0101110 1000010
12 2023-02-01 10:00:00 2023-02-05 17:00:00 35 700.00 12 0101111 1000011
13 2023-06-23 11:00:00 2023-06-30 14:00:00 45 800.00 11 0110000 1000100
14 2023-03-07 12:00:00 2023-03-18 15:00:00 10 800.00 12 0110001 1000101
15 2023-01-23 07:00:00 2023-01-29 18:00:00 12 700.00 13 0110010 1000110
16 2023-05-06 08:00:00 2023-05-13 19:00:00 35 70.00 14 0110011 1000111
17 2023-04-09 09:00:00 2023-04-18 20:00:00 20 600.00 15 0110100 1001000
18 2023-05-23 10:00:00 2023-05-28 21:00:00 30 20000.00 18 0110101 1001001
19 2023-07-15 11:00:00 2023-07-18 22:00:00 15 150.00 19 0110110 1001010
20 2023-07-21 12:00:00 2023-07-28 23:00:00 20 600.00 20 0110111 1001011
27 2023-05-05 09:00:00 2023-10-05 18:00:00 10 500.00 1 0100100 0111000
NULL

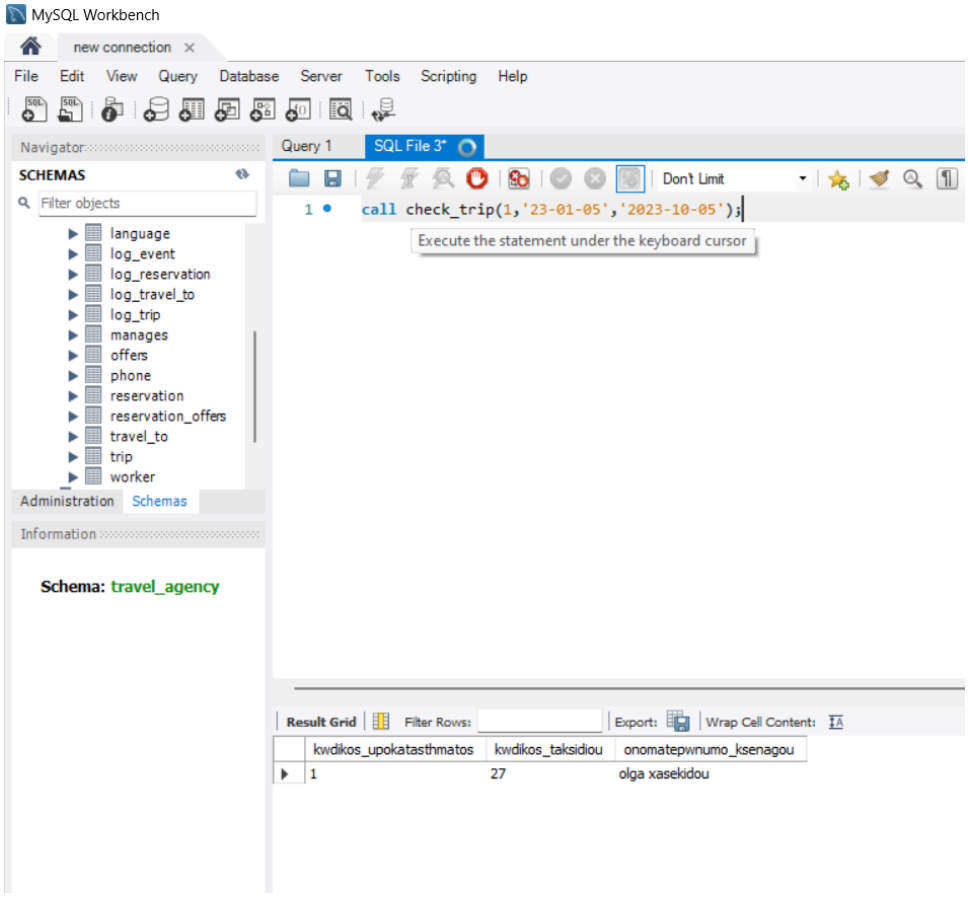

MySQL Workbench
new connection
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
language
log_event
log_reservation
log_travel_to
log_trip
manages
offers
phone
reservation
reservation_offers
travel_to
trip
worker
Administration Schemas
Information
Schema: travel_agency
Query 1
SQL File 3*
Don't Limit
1 • call check_trip(1,'23-01-05','2023-10-05');
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
kwdikos_upokatasthmatos kwdikos_taksidiou kostos_taksidiou megistes_theseis hm_anaxwrishs hm_epistrofhs kenestheseis
1 1 500.00 10 2023-01-05 09:00:00 2023-01-05 18:00:00 9


MySQL Workbench
new connection
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
language
log_event
log_reservation
log_travel_to
log_trip
manages
offers
phone
reservation
reservation_offers
travel_to
trip
worker
Administration Schemas
Information
Schema: travel_agency
Query 1
SQL File 3*
Don't Limit
1 call check_trip(1,'23-01-05','2023-10-05');
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
kwdikos_upokatasthmatos
kwdikos_taksidiou
onomatepwnumo_odhgou
1
1
lamprini lamprou


MySQL Workbench
new connection
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
language
log_event
log_reservation
log_travel_to
log_trip
manages
offers
phone
reservation
reservation_offers
travel_to
trip
worker
Administration Schemas
Information
Schema: travel_agency
Query 1
SQL File 3*
Don't Limit
1 call check_trip(1,'23-01-05','2023-10-05');
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
kwdikos_upokatasthmatos
kwdikos_taksidiou
onomatepwnumo_ksenagou
1
1
olga xasekidou


MySQL Workbench
new connection
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
language
log_event
log_reservation
log_travel_to
log_trip
manages
offers
phone
reservation
reservation_offers
travel_to
trip
worker
Administration Schemas
Information
Schema: travel_agency
Query 1
SQL File 3*
Don't Limit
1 call check_trip(1,'23-01-05','2023-10-05');
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
kwdikos_upokatasthmatos kwdikos_taksidiou kostos_taksidiou megistes_theseis hm_anaxwrishs hm_epistroffhs kenestheseis
1 27 500.00 10 2023-05-05 09:00:00 2023-10-05 18:00:00 9


MySQL Workbench
new connection
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
language
log_event
log_reservation
log_travel_to
log_trip
manages
offers
phone
reservation
reservation_offers
travel_to
trip
worker
Administration Schemas
Information
Schema: travel_agency
Query 1
SQL File 3*
Don't Limit
1 call check_trip(1,'23-01-05','2023-10-05');
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
kwdikos_upokatasthmatos kwdikos_taksidiou onomatepwnumo_odhgou
1 27 lamprini lamprou



3.1.3.4

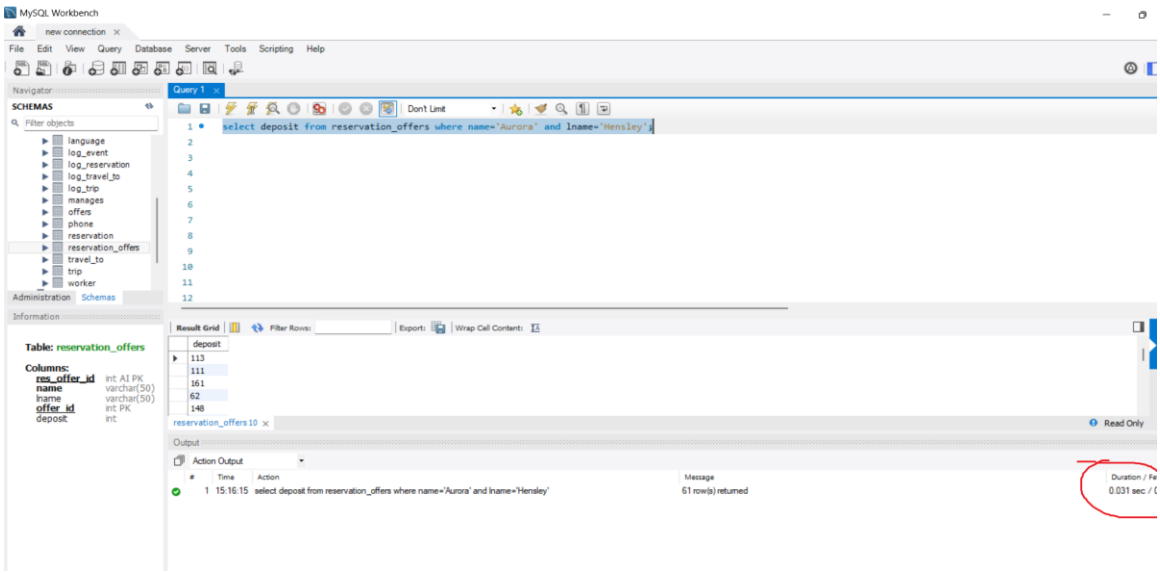
CREATE INDEX id_nlname ON reservation_offers(name,lname);

//Επιλέγω να δημιουργήσω ευρετήριο ,βάση του ονόματος και του επιθέτου, διότι πολλές φορές θα χρειαστεί να κάνω μια αναζήτηση στον πίνακα reservation_offers βάση του ονόματος και όχι του res_offer_id.

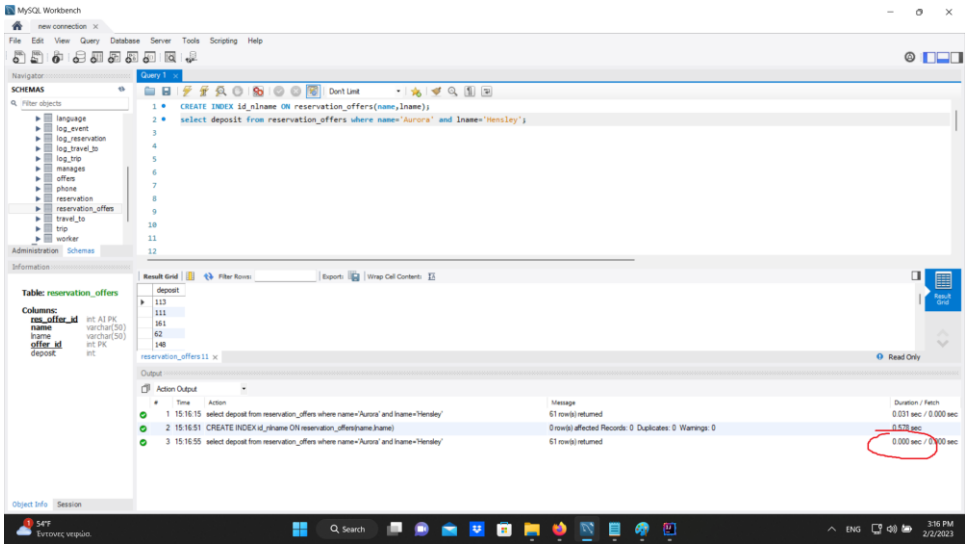
//Παρατηρώ την διαφορά στους χρόνους εκτέλεσης ενός τυχαίου ερωτήματος .Αυτό οφείλεται στο ότι τα ευρητήρια δημιουργούν μικρότερους πίνακες από τις καθορισμένες στηλες name,lname του πίνακα reservation_offers. Δηλαδή μεταφέρουμε ένα μη ταξινομημένο πίνακα σε μια σειρά που θα μεγιστοποιήσει την αποτελεσματικότητα του ερωτήματος κατά την αναζήτηση.

//σε κάποια ερωτήματα στη συνέχεια ,θα εμφανίζονται τα ίδια ονόματα γιατί έχω περάσει τα 1000*60 φορές

//0.031 sec



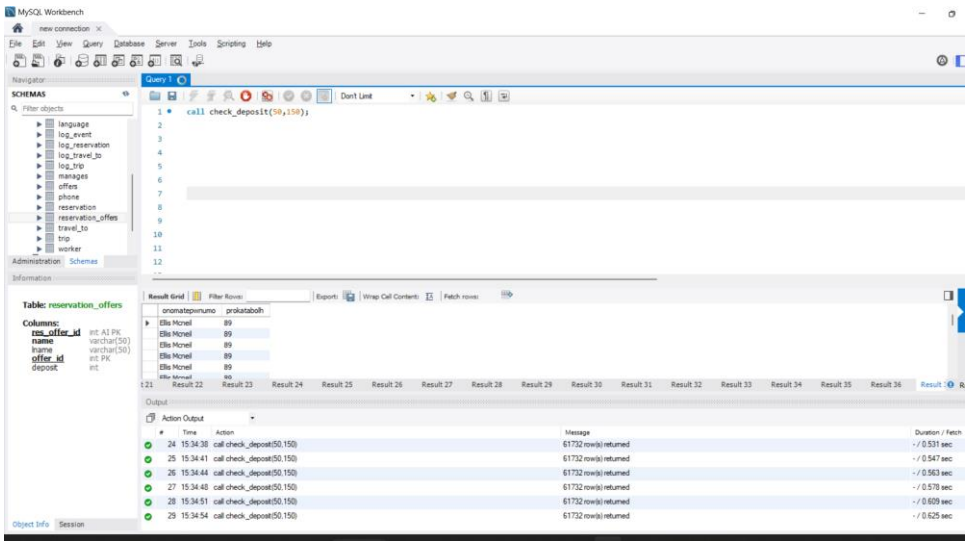
//0.00 sec

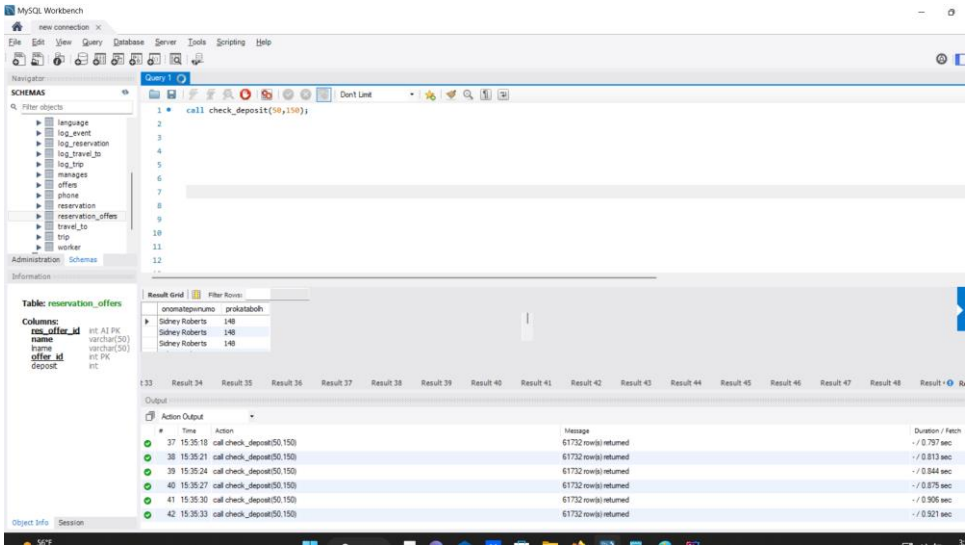


α.

```
DELIMITER $

CREATE PROCEDURE  check_deposit( IN num1 INT, IN num2 INT )

BEGIN

DECLARE name_offer VARCHAR(20);

DECLARE lname_offer VARCHAR(20);

DECLARE deposit_offer INT;

DECLARE flag INT ;
```

```
DECLARE cr CURSOR FOR

SELECT name,lname,deposit

FROM reservation_offers

WHERE deposit BETWEEN num1 AND num2;



DECLARE CONTINUE HANDLER FOR NOT FOUND SET flag=1;

OPEN cr;

SET flag=0;

FETCH cr INTO name_offer,lname_offer,deposit_offer;

WHILE(flag=0) DO

FETCH cr INTO name_offer,lname_offer,deposit_offer;

SELECT CONCAT(name_offer,' ',lname_offer) AS onomatepwnumo, deposit_offer AS
prokatabolh

FROM reservation_offers

WHERE deposit_offer BETWEEN num1 AND num2;

DO SLEEP(3);

END WHILE;

CLOSE cr;

END$

DELIMITER;
```





β.

```
CREATE PROCEDURE lname_offer( IN lname_offer VARCHAR(20))

BEGIN

DECLARE  myname VARCHAR(20);

DECLARE  mylname VARCHAR(20);

DECLARE  myofferid INT ;

DECLARE flag INT;


DECLARE cr CURSOR FOR

SELECT name,lname,offer_id FROM reservation_offers WHERE lname=lname_offer;


DECLARE CONTINUE HANDLER FOR NOT FOUND SET flag=1;

OPEN cr;

SET flag=0;

WHILE(flag=0) DO

FETCH cr INTO myname,mylname,myofferid;

SELECT CONCAT(myname,' ',mylname) AS onomatepwnumo,myofferid AS
kwdikos_prosforas

FROM reservation_offers
```

WHERE lname=mylname;

DO SLEEP(1);

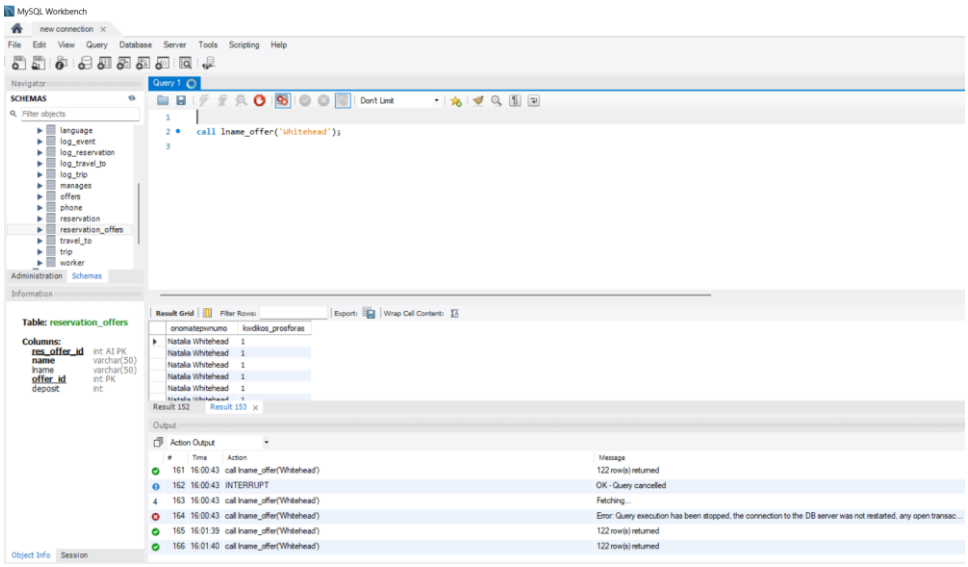
END WHILE;

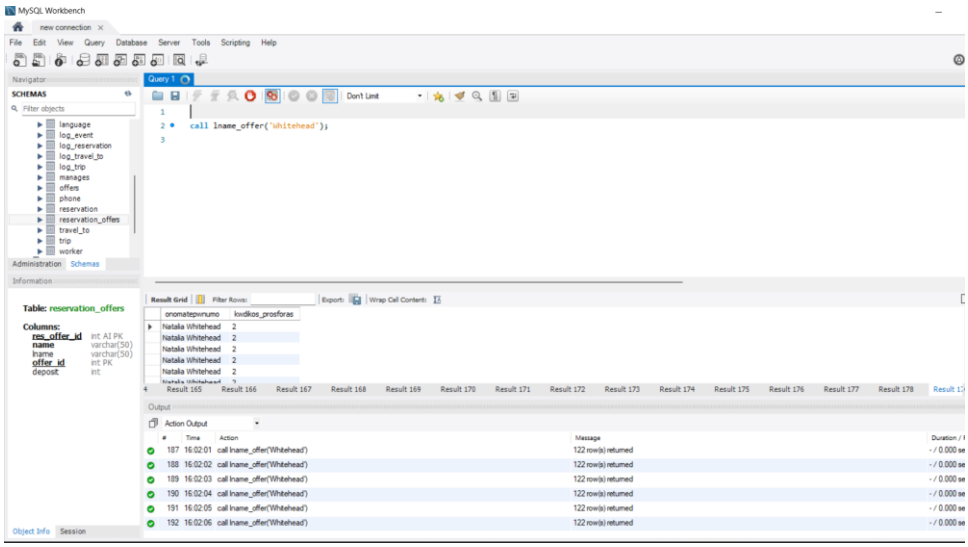
CLOSE cr;

END$

DELIMITER;

//Έστω ότι θέλω να βρω τους πελάτες/πελάτη με το επίθετο ‘Whitehead’ και την προσφορά ταξιδιού που έχουν δηλώσει.





Κεφάλαιο 3

TRIGGERS :

3.1.4.1 // δεν έχω κάνει την σύνδεση ώστε να δείχνει ποιος εκτέλεσε μία ενέργεια

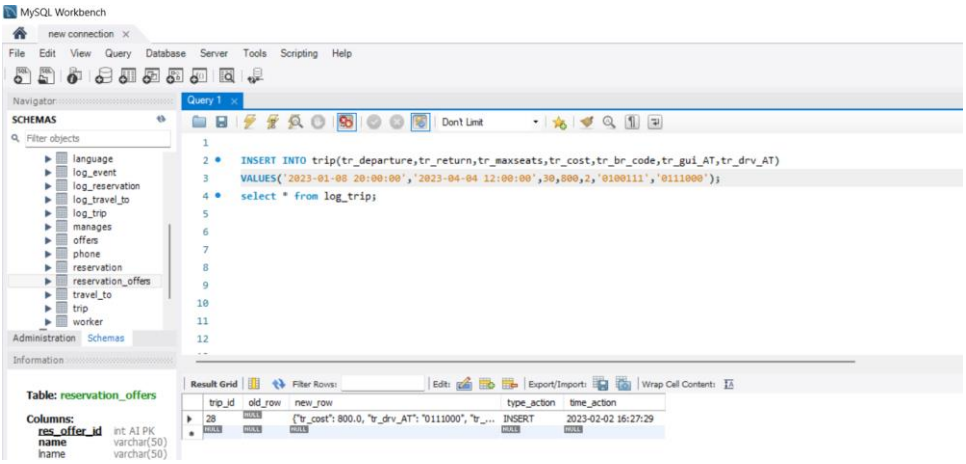
α. table trip

1. DELIMITER $ // insert

```
CREATE TRIGGER trip_insert_log_trigger
AFTER INSERT ON trip FOR EACH ROW
BEGIN
  INSERT INTO log_trip (
    trip_id,
    old_row,
    new_row,
    type_action,
    time_action
   )
  VALUES(
    NEW.tr_id,
    null,
    JSON_OBJECT(
      "tr_departure", NEW.tr_departure,
      "tr_return", NEW.tr_return,
      "tr_maxseats", NEW.tr_maxseats,
      "tr_cost", NEW.tr_cost,
      "tr_br_code",NEW.tr_br_code,
      "tr_gui_AT", NEW.tr_gui_AT,
```

```
        "tr_drv_AT",NEW.tr_drv_AT),
    'INSERT',
CURRENT_TIMESTAMP
  );
END $
DELIMITER ;
```

//Κάνω μία εισαγωγή για να δω πως τρέχει ο πίνακας log_trip

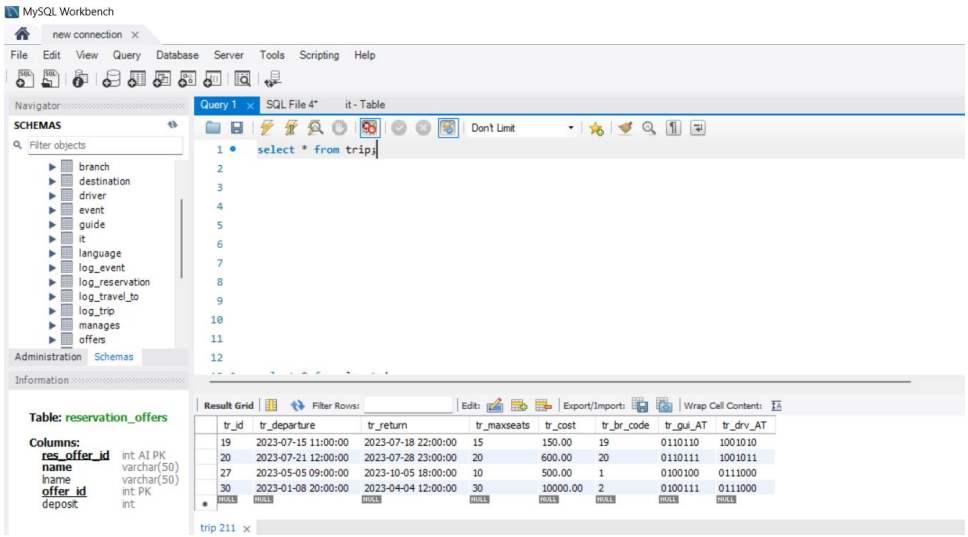


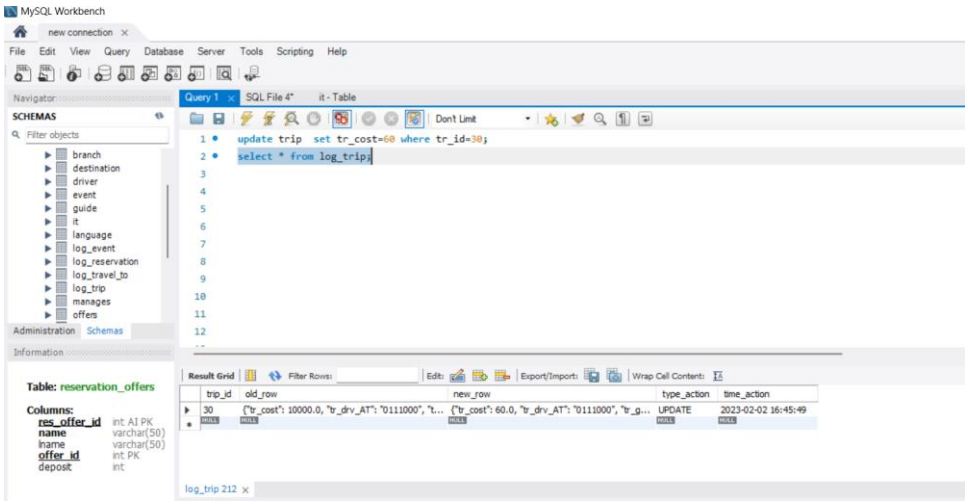
2. CREATE TRIGGER trip_update_log_trigger //update

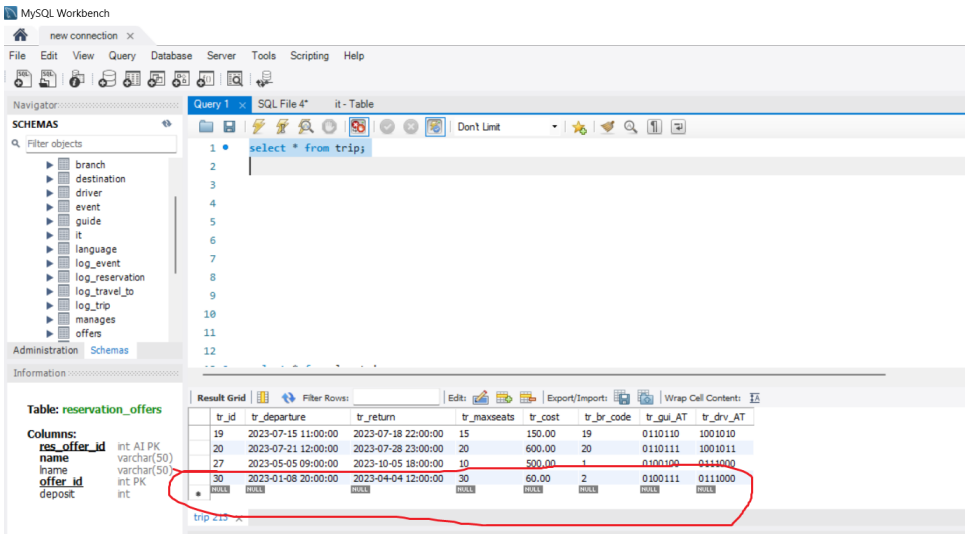
```
AFTER UPDATE ON trip FOR EACH ROW
BEGIN
  INSERT INTO log_trip (
trip_id,
      old_row,
      new_row,
      type_action,
      time_action
  )
  VALUES(
      NEW.tr_id,
```

```
    JSON_OBJECT(
      "tr_departure", OLD.tr_departure,
      "tr_return", OLD.tr_return,
      "tr_maxseats", OLD.tr_maxseats,
      "tr_cost", OLD.tr_cost,
      "tr_br_code",OLD.tr_br_code,
      "tr_gui_AT", OLD.tr_gui_AT,
      "tr_drv_AT",OLD.tr_drv_AT
    ),
    JSON_OBJECT(
      "tr_departure", NEW.tr_departure,
      "tr_return", NEW.tr_return,
      "tr_maxseats", NEW.tr_maxseats,
      "tr_cost", NEW.tr_cost,
      "tr_br_code",NEW.tr_br_code,
      "tr_gui_AT", NEW.tr_gui_AT,
      "tr_drv_AT",NEW.tr_drv_AT
    ),
    'UPDATE',
    CURRENT_TIMESTAMP
  );
END $
DELIMITER ;
```

//Παρατηρώ την τελευταία εισαγωγή με tr_id=30



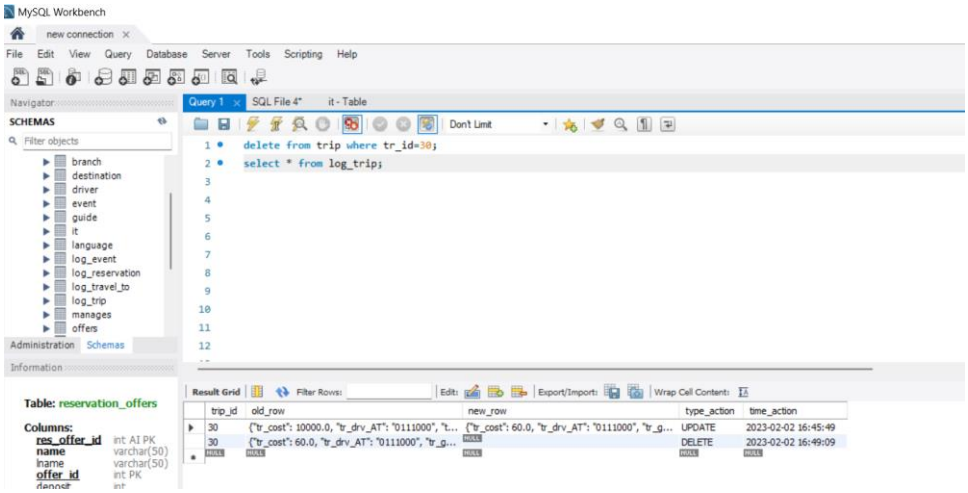




3. DELIMITER $ //delete

CREATE TRIGGER trip_delete_log_trigger

```
AFTER DELETE ON trip FOR EACH ROW
BEGIN
   INSERT INTO log_trip(
      trip_id,
      old_row,
      new_row,
      type_action,
      time_action
   )
   VALUES(
      OLD.tr_id,
      JSON_OBJECT(
        "tr_departure", OLD.tr_departure,
         "tr_return", OLD.tr_return,
         "tr_maxseats", OLD.tr_maxseats,
         "tr_cost", OLD.tr_cost,
         "tr_br_code",OLD.tr_br_code,
         "tr_gui_AT", OLD.tr_gui_AT,
         "tr_drv_AT",OLD.tr_drv_AT
      ),
      null,
      'DELETE',
      CURRENT_TIMESTAMP
   );
END $
DELIMITER ;
```



***Αντίστοιχα και στους επόμενους log πίνακες reservation,travel_to,event

β. table reservation

1. DELIMITER $

```
CREATE TRIGGER rsv_insert_log_trigger //insert
AFTER INSERT ON reservation FOR EACH ROW
BEGIN
   INSERT INTO log_reservation (
      res_trip_id,
      old_row,
      new_row,
      type_action,
      time_action
   )
   VALUES(
      NEW.res_tr_id,
      null,
      JSON_OBJECT(
         "res_tr_id", NEW.res_tr_id,
         "res_seatnum", NEW.res_seatnum,
         "res_name", NEW.res_name,
```

```
        "res_lname", NEW.res_lname,
        "res_isadult",NEW.res_isadult
      ),
      'INSERT',
      CURRENT_TIMESTAMP
  );
END $
DELIMITER ;
```

2. DELIMITER $ //update

```
CREATE TRIGGER rsv_update_log_trigger
AFTER UPDATE ON reservation FOR EACH ROW
BEGIN
  INSERT INTO log_reservation (
res_trip_id,
      old_row,
      new_row,
      type_action,
      time_action
  )
  VALUES(
      NEW.res_tr_id,
      JSON_OBJECT(
        "res_tr_id", OLD.res_tr_id,
        "res_seatnum", OLD.res_seatnum,
        "res_name", OLD.res_name,
        "res_lname", OLD.res_lname,
        "res_isadult",OLD.res_isadult
      ),
```

```
      JSON_OBJECT(
"res_tr_id", NEW.res_tr_id,
         "res_seatnum", NEW.res_seatnum,
         "res_name", NEW.res_name,
         "res_lname", NEW.res_lname,
         "res_isadult",NEW.res_isadult
      ),
      'UPDATE',
      CURRENT_TIMESTAMP
   );
END $
DELIMITER ;
```

3. DELIMITER $ //delete

```
CREATE TRIGGER rsv_delete_log_trigger
AFTER DELETE ON reservation FOR EACH ROW
BEGIN
   INSERT INTO log_reservation(
      res_trip_id,
      old_row,
      new_row,
      type_action,
      time_action
   )
   VALUES(
      OLD.res_tr_id,
      JSON_OBJECT(
         "res_tr_id", OLD.res_tr_id,
         "res_seatnum", OLD.res_seatnum,
```

```
        "res_name", OLD.res_name,
        "res_lname", OLD.res_lname,
        "res_isadult",OLD.res_isadult
      ),
      null,
      'DELETE',
      CURRENT_TIMESTAMP
  );
END $
DELIMITER ;
```

γ. table event

```
1. DELIMITER $ //insert
CREATE TRIGGER event_insert_log_trigger
AFTER INSERT ON event FOR EACH ROW
BEGIN
  INSERT INTO log_event (
      event_trip_id,
      old_row,
      new_row,
      type_action,
      time_action
  )
  VALUES(
      NEW.ev_tr_id,
      null,
      JSON_OBJECT(
```

```
        "ev_start", NEW.ev_start,
        "ev_end", NEW.ev_end,
        "ev_descr", NEW.ev_descr

    ),
    'INSERT',
    CURRENT_TIMESTAMP
  );
END $
DELIMITER ;
```

2. DELIMITER $ //update

```
CREATE TRIGGER event_update_log_trigger
AFTER UPDATE ON event FOR EACH ROW
BEGIN
  INSERT INTO log_event (
  event_trip_id,
    old_row,
    new_row,
    type_action,
    time_action
  )
  VALUES(
    NEW.ev_tr_id,
    JSON_OBJECT(
      "ev_start", OLD.ev_start,
      "ev_end", OLD.ev_end,
      "ev_descr", OLD.ev_descr
```

```
        ),
        JSON_OBJECT(
 "ev_start", NEW.ev_start,
            "ev_end", NEW.ev_end,
            "ev_descr", NEW.ev_descr
        ),
        'UPDATE',
        CURRENT_TIMESTAMP
    );
END $
DELIMITER ;
```

3. DELIMITER $ //delete

```
CREATE TRIGGER event_delete_log_trigger
AFTER DELETE ON event FOR EACH ROW
BEGIN
    INSERT INTO log_event(
        event_trip_id,
        old_row,
        new_row,
        type_action,
        time_action
    )
    VALUES(
        OLD.ev_tr_id,
        JSON_OBJECT(
"ev_start", OLD.ev_start,
```

```
        "ev_end", OLD.ev_end,
        "ev_descr", OLD.ev_descr
      ),
      null,
      'DELETE',
      CURRENT_TIMESTAMP
  );
END $
DELIMITER ;
```

δ. table travel_to

1. DELIMITER $$ //insert

```
CREATE TRIGGER trto_insert_log_trigger
AFTER INSERT ON travel_to FOR EACH ROW
BEGIN
  INSERT INTO log_travel_to (
    trto_trip_id,
    old_row,
    new_row,
    type_action,
    time_action
  )
  VALUES(
    NEW.to_tr_id,
    null,
    JSON_OBJECT(
      "to_dst_id", NEW.to_dst_id,
      "to_arrival", NEW.to_arrival,
```

```
        "to_departure", NEW.to_departure

      ),
      'INSERT',
      CURRENT_TIMESTAMP
  );
END $
DELIMITER ;
```

2. DELIMITER $$ //update

```
CREATE TRIGGER trto_update_log_trigger
AFTER UPDATE ON travel_to FOR EACH ROW
BEGIN
  INSERT INTO log_travel_to (
  trto_trip_id,
    old_row,
    new_row,
    type_action,
    time_action
  )
  VALUES(
    NEW.to_tr_id,
    JSON_OBJECT(
      "to_dst_id", OLD.to_dst_id,
        "to_arrival", OLD.to_arrival,
        "to_departure", OLD.to_departure
    ),
    JSON_OBJECT(
```

```
    "to_dst_id", NEW.to_dst_id,
        "to_arrival", NEW.to_arrival,
        "to_departure", NEW.to_departure
      ),
      'UPDATE',
      CURRENT_TIMESTAMP
    );
END $
DELIMITER ;
```

3. DELIMITER $ //delete

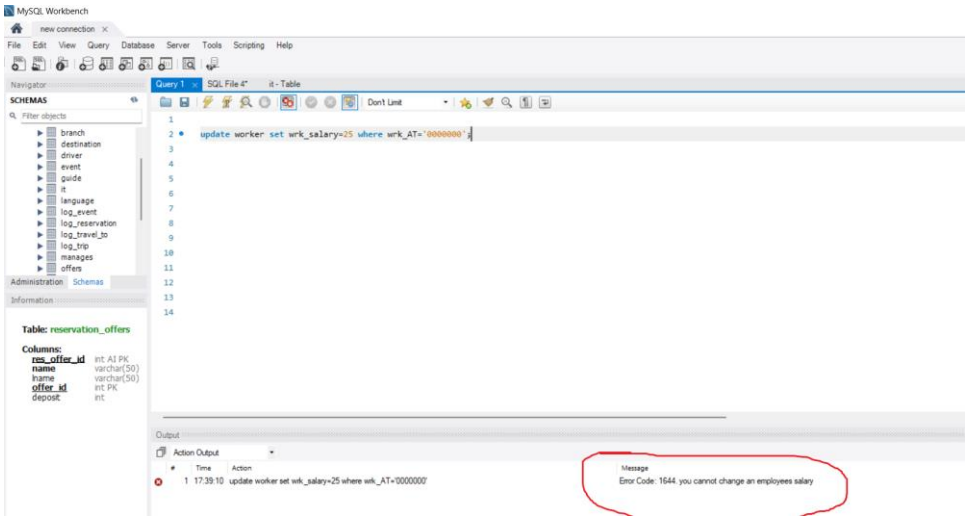
```
CREATE TRIGGER trto_delete_log_trigger
AFTER DELETE ON travel_to FOR EACH ROW
BEGIN
    INSERT INTO log_travel_to(
      trto_trip_id,
      old_row,
      new_row,
      type_action,
      time_action
    )
    VALUES(
      OLD.to_tr_id,
      JSON_OBJECT(
"to_dst_id", OLD.to_dst_id,
        "to_arrival", OLD.to_arrival,
        "to_departure", OLD.to_departure
      ),
      null,
```

```
    'DELETE',
    CURRENT_TIMESTAMP
  );
END  $
DELIMITER ;
```

3.1.4.3

α.

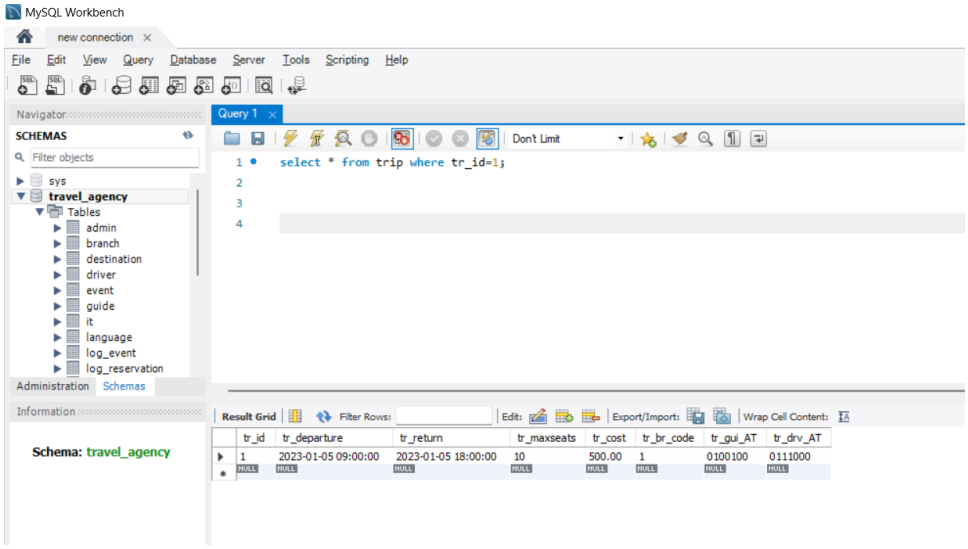
```
DELIMITER $
CREATE TRIGGER salary
BEFORE UPDATE ON worker
FOR EACH ROW
BEGIN

IF NEW.wrk_salary NOT LIKE OLD.wrk_salary THEN
SIGNAL SQLSTATE VALUE '45000'
SET MESSAGE_TEXT = 'YOU CANT CHANGE AN EMPLOYEES SALARY';
END IF;
END $
DELIMITER;
```



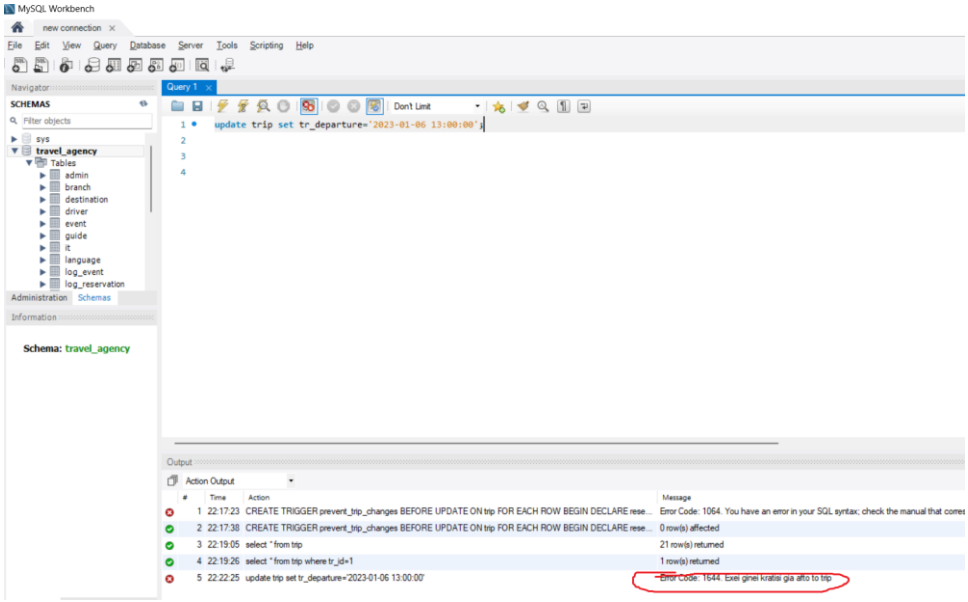
β.

```
DELIMITER $
CREATE TRIGGER prevent_trip_changes
BEFORE UPDATE ON trip
FOR EACH ROW
BEGIN
DECLARE reservation_count INT;

SELECT COUNT(*) INTO reservation_count
FROM reservation
WHERE res_tr_id = OLD.tr_id;

IF reservation_count > 0 THEN
IF NEW.tr_departure != OLD.tr_departure OR
NEW.tr_return != OLD.tr_return OR
NEW.tr_cost != OLD.tr_cost THEN
SIGNAL SQLSTATE "45000" SET MESSAGE_TEXT = "Exei ginei kratisi gia afto to trip";
END IF;
END IF;
END$
DELIMITER ;
```

//πχ αν θελω να αλλαξω την ημερομηνια του ταξιδιου με tr_id=1



//μου εμφανίζει κατάλληλο μήνυμα ότι δεν είναι δυνατή αυτή η ενέργεια

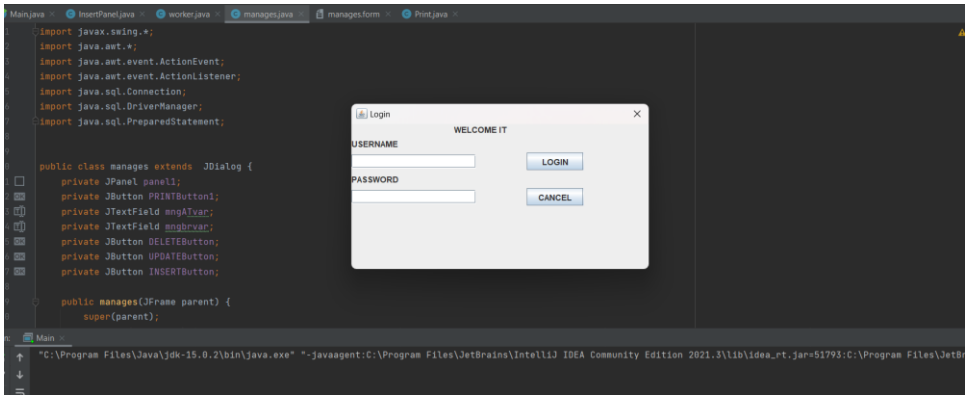


Β Μέρος:

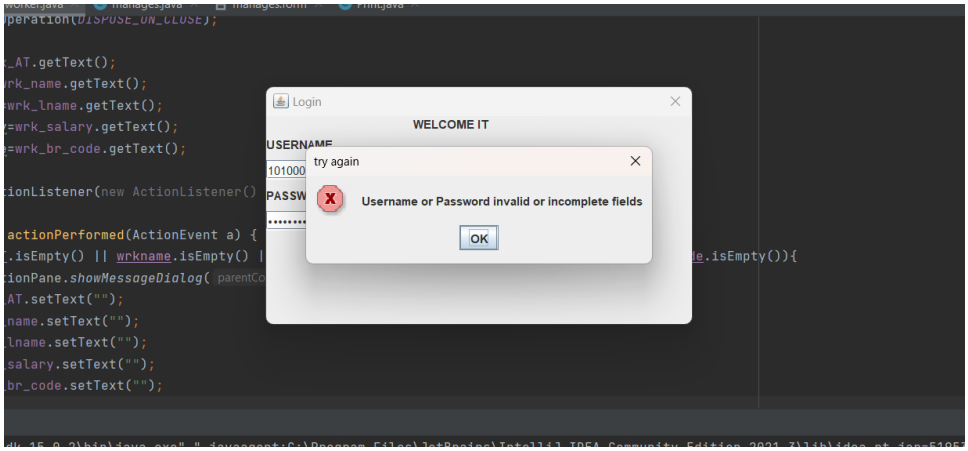
***η εκτύπωση των πινάκων δεν μου τρέχει οπότε θα σας δείξω τα αποτελέσματα στο mysql workbench**

***τα delete/update θα λειτουργούν ακόμα και αν ο it δεν βάλει όλα τα στοιχεία**

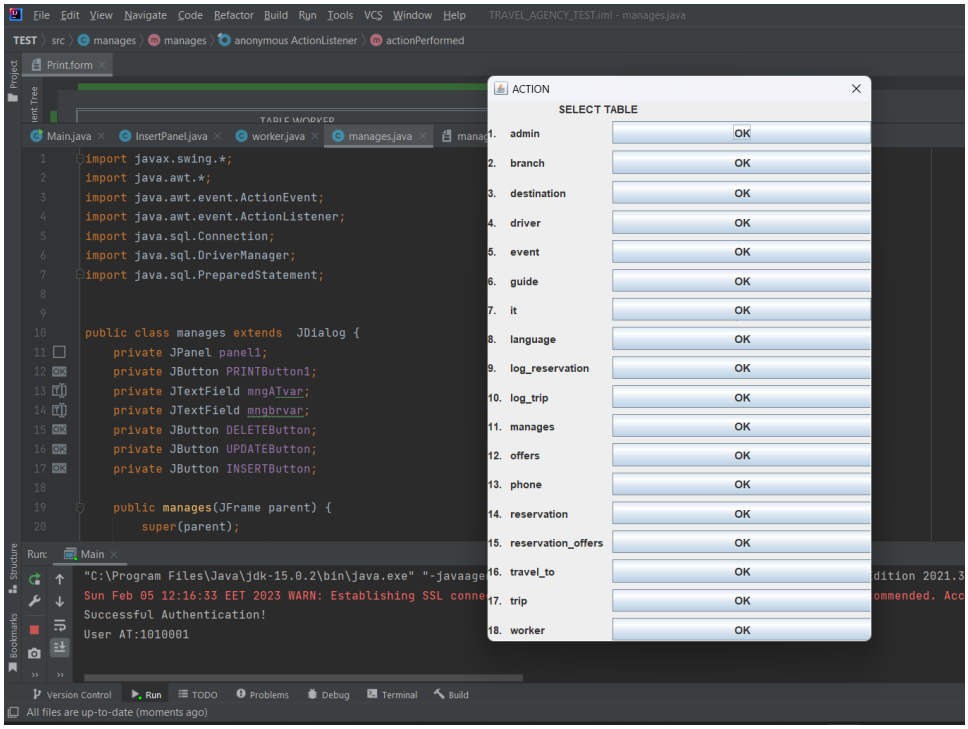
//Στο login panel θα συνδέεται ο κάθε it με το AT.

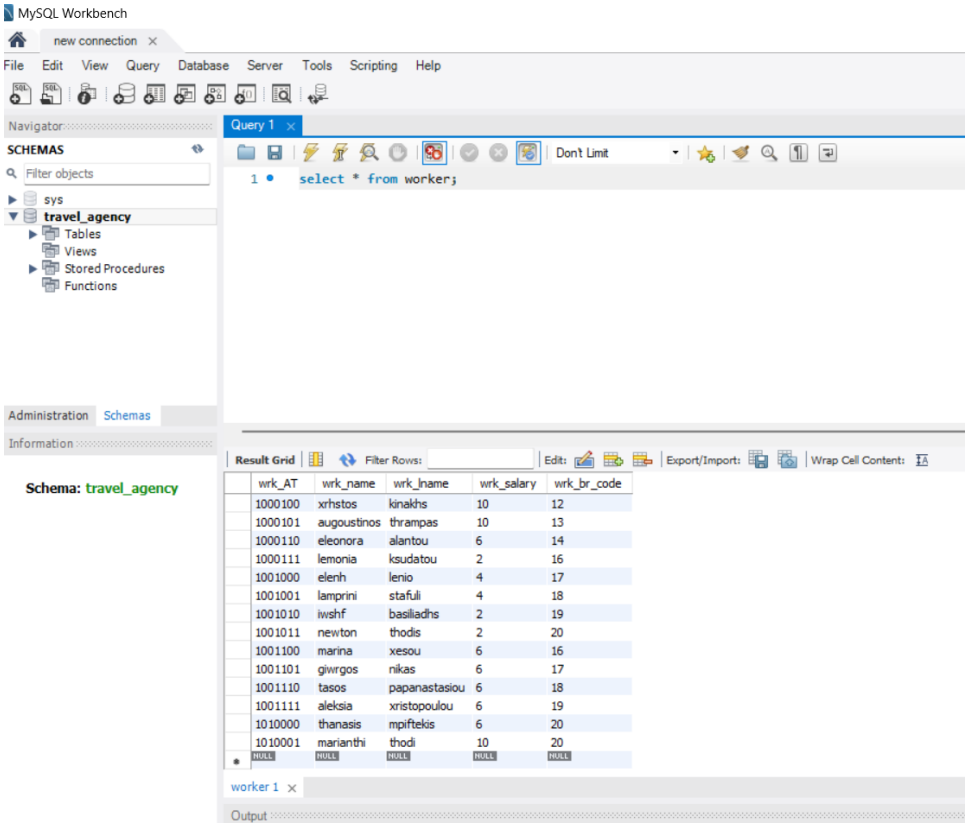


//σε περίπτωση που το όνομα ή ο κωδικός δεν είναι σωστά ή δεν έχουν συμπληρωθεί όλα τα στοιχεία

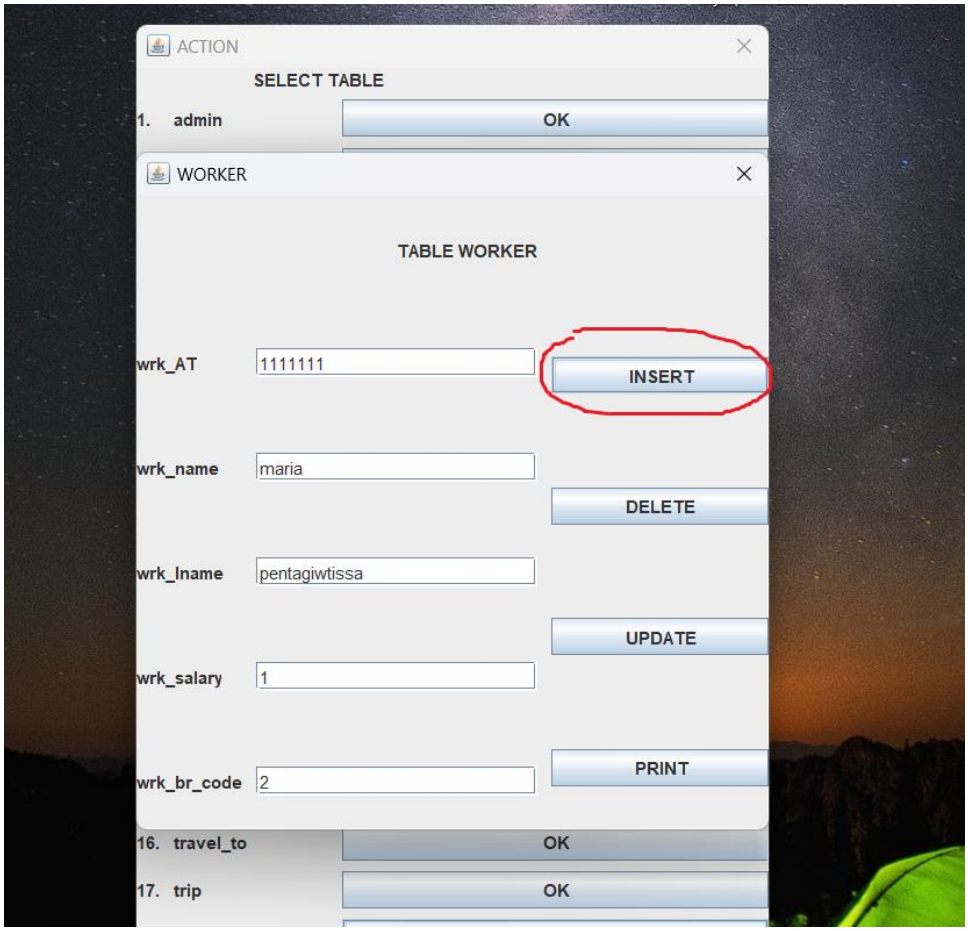


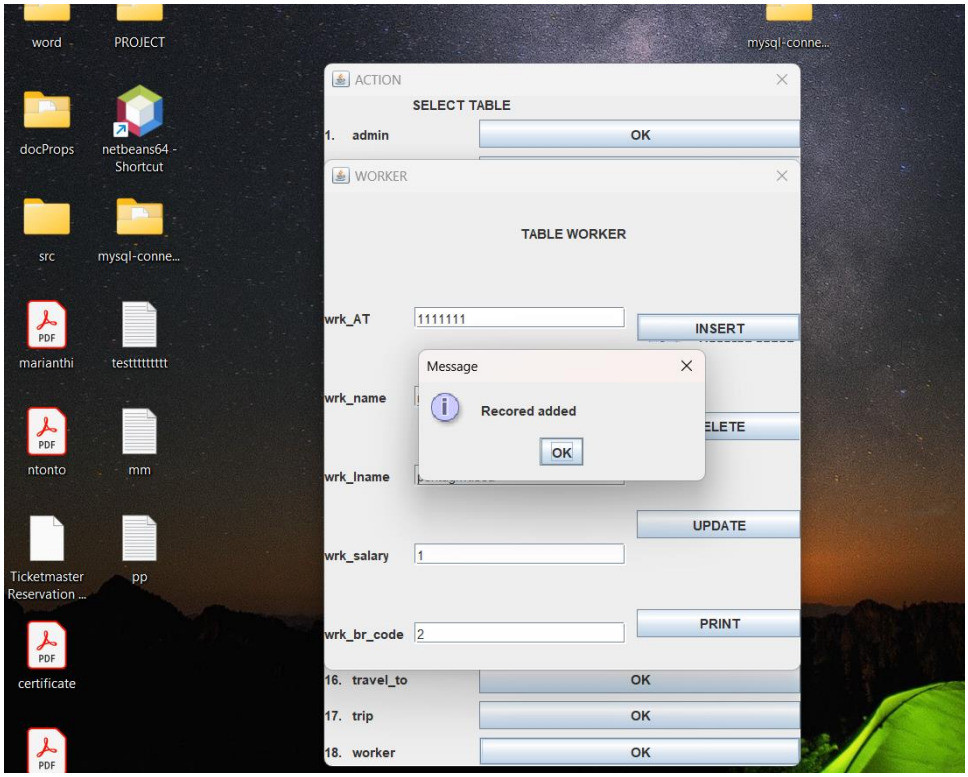
//θα διαλέγω ένα πίνακα που θέλω να κάνω μία ενέργεια πχ τον worker

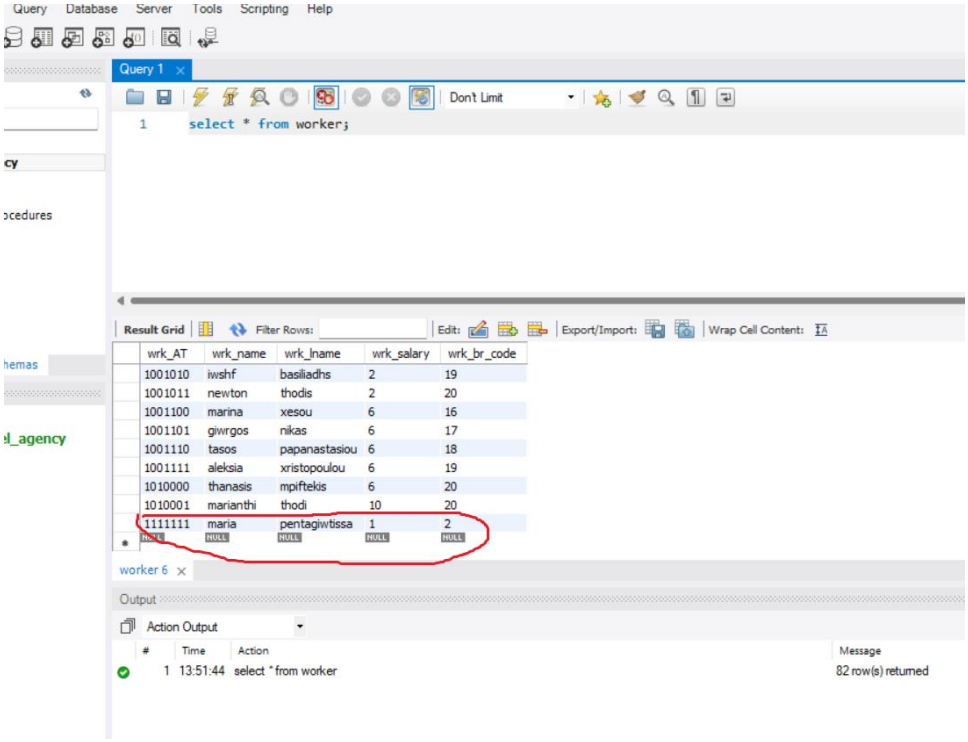

File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
TRAVEL_AGENCY_TEST.iml - manages.java
TEST src manages manages anonymous ActionListener actionPerformed
Print.form
Main.java InsertPanel.java worker.java manages.java
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
public class manages extends JDialog {
private JPanel panel1;
private JButton PRINTButton1;
private JTextField mngATvar;
private JTextField mngbrvar;
private JButton DELETEButton;
private JButton UPDATEButton;
private JButton INSERTButton;
public manages(JFrame parent) {
super(parent);
Run: Main
"C:\Program Files\Java\jdk-15.0.2\bin\java.exe"
Sun Feb 05 12:16:33 EET 2023 WARN: Establishing SSL
Successful Authentication!
User AT:1010001
Version Control Run TODO Problems Debug Terminal Build
All files are up-to-date (moments ago)
ACTION
SELECT TABLE
1. admin OK
2. branch OK
3. destination OK
4. driver OK
5. event OK
6. guide OK
7. it OK
8. language OK
9. log_reservation OK
10. log_trip OK
11. manages OK
12. offers OK
13. phone OK
14. reservation OK
15. reservation_offers OK
16. travel_to OK
17. trip OK
18. worker OK


MySQL Workbench
new connection
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
sys
travel_agency
Tables
Views
Stored Procedures
Functions
Administration Schemas
Information
Schema: travel_agency
Query 1
Don't Limit
1 select * from worker;
Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:
wrk_AT wrk_name wrk_lname wrk_salary wrk_br_code
1000100 xrhstos kinakhs 10 12
1000101 augoustinos thrampas 10 13
1000110 eleonora alantou 6 14
1000111 lemonia ksudatou 2 16
1001000 elenh lenio 4 17
1001001 lamprini stafuli 4 18
1001010 iwshf basiliadhs 2 19
1001011 newton thodis 2 20
1001100 marina xesou 6 16
1001101 giwrgos nikas 6 17
1001110 tasos papanastasiou 6 18
1001111 aleksia xristopoulou 6 19
1010000 thanasis mpiftekis 6 20
1010001 marianthi thodi 10 20
worker 1
Output

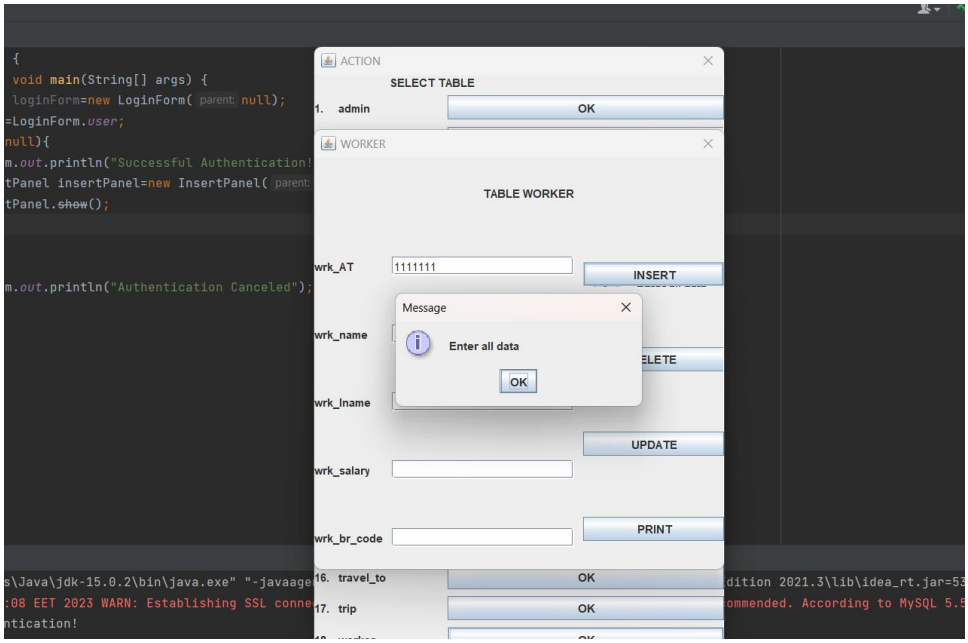
//insert


ACTION
SELECT TABLE
1. admin
OK
WORKER
TABLE WORKER
wrk_AT
1111111
INSERT
wrk_name
maria
DELETE
wrk_lname
pentagiwtissa
UPDATE
wrk_salary
1
wrk_br_code
2
PRINT
16. travel_to
OK
17. trip
OK

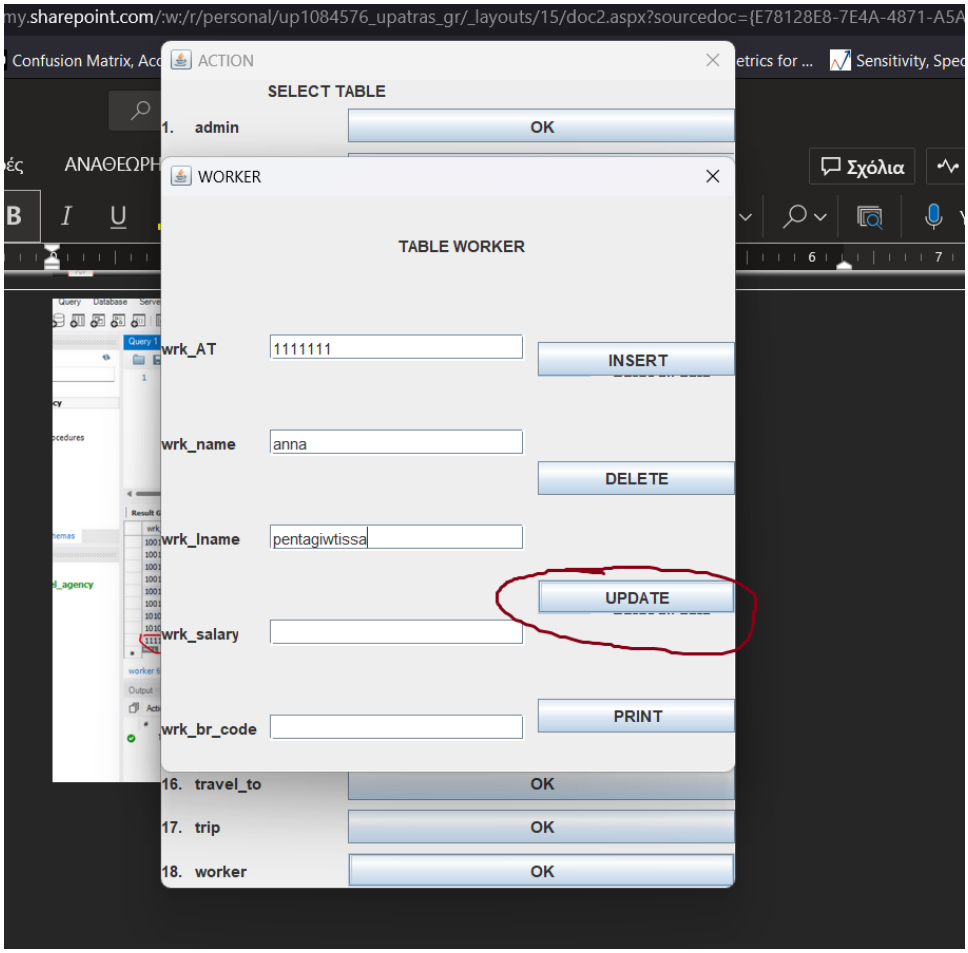

word
PROJECT
mysql-conne...
docProps
netbeans64 - Shortcut
src
mysql-conne...
marianthi
testttttttttt
ntonto
mm
Ticketmaster Reservation ...
pp
certificate
ACTION
SELECT TABLE
1. admin
OK
WORKER
TABLE WORKER
wrk_AT
1111111
INSERT
Message
Recored added
OK
wrk_name
wrk_lname
UPDATE
wrk_salary
1
wrk_br_code
2
PRINT
16. travel_to
OK
17. trip
OK
18. worker
OK

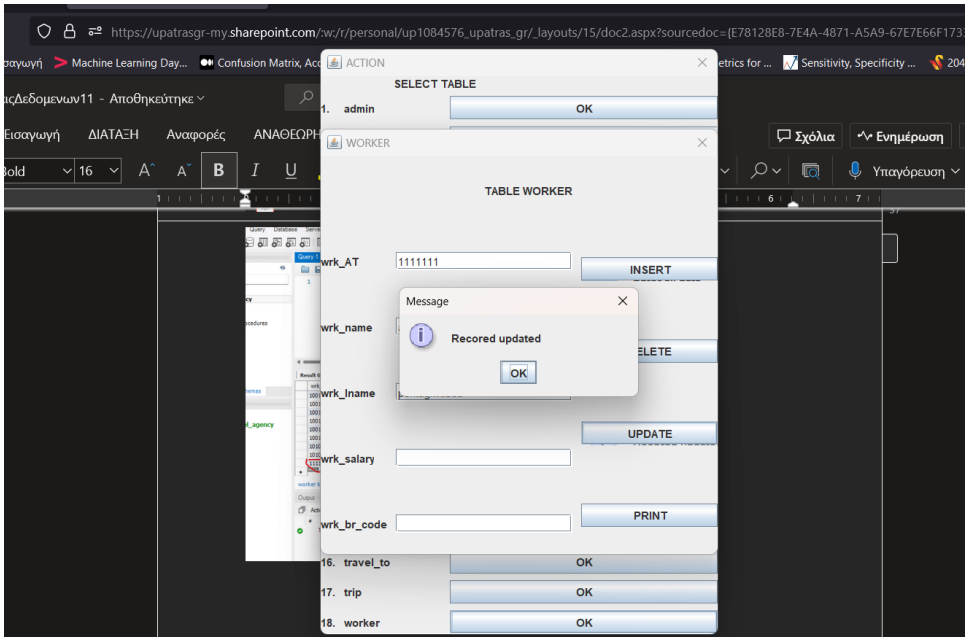


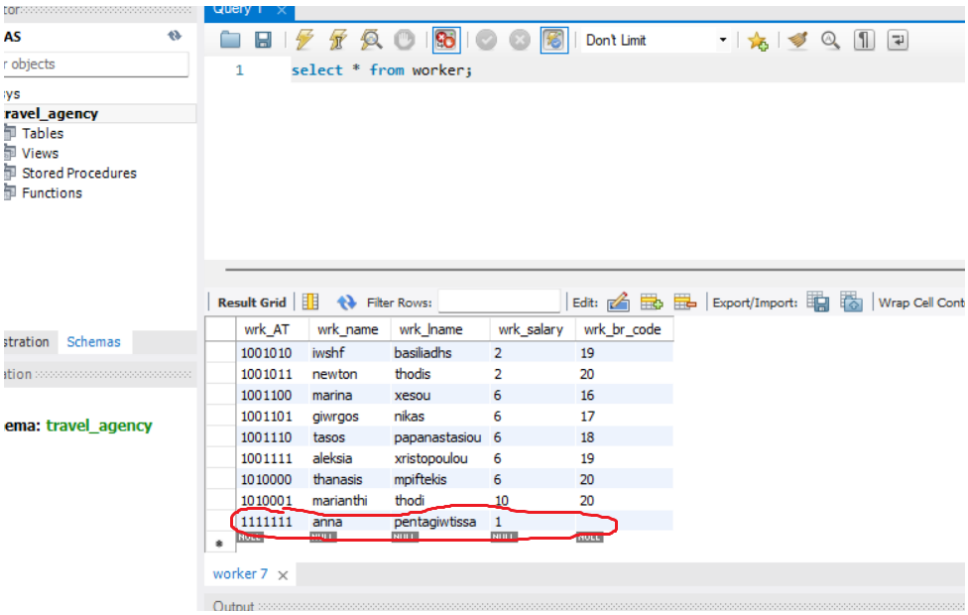
//στην περίπτωση του insert πρέπει να τοποθετηθούν όλα τα στοιχεία ενός worker



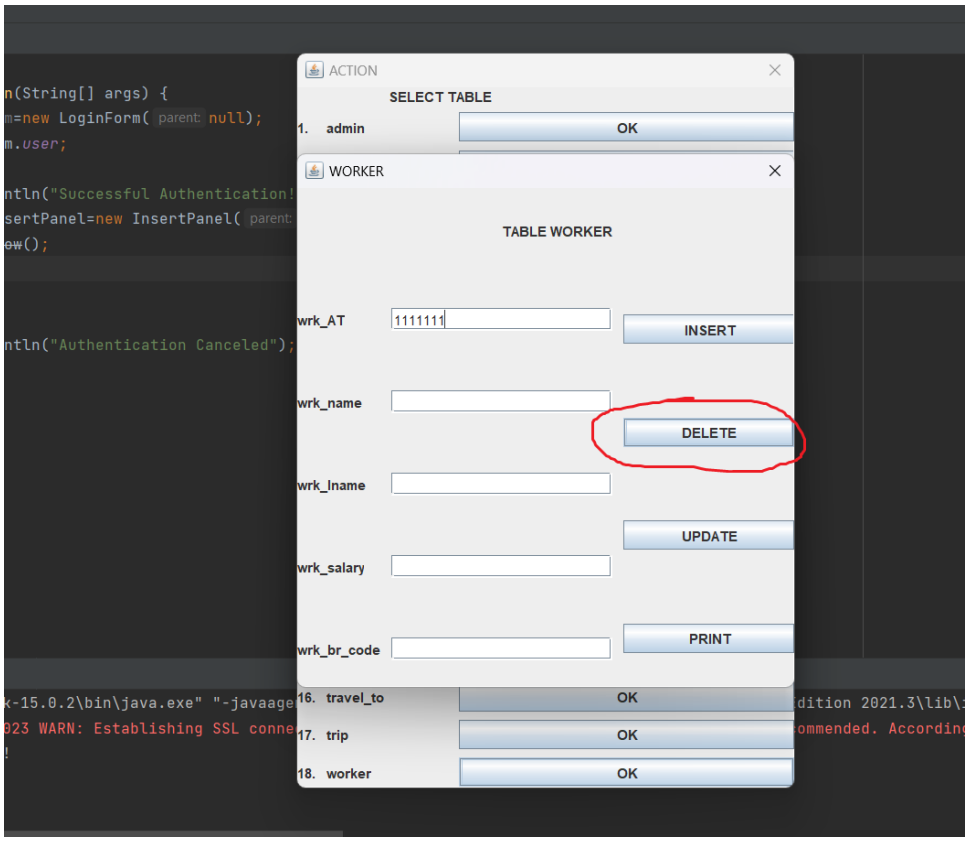
//τροποποιώ την παραπάνω εγγραφή και από wrk_name=”maria” , το αλλάζω σε wrk_name=”anna”.Δεν χρειάζεται η τοποθέτηση όλων των στοιχειών αρκεί το wrk_AT. // update

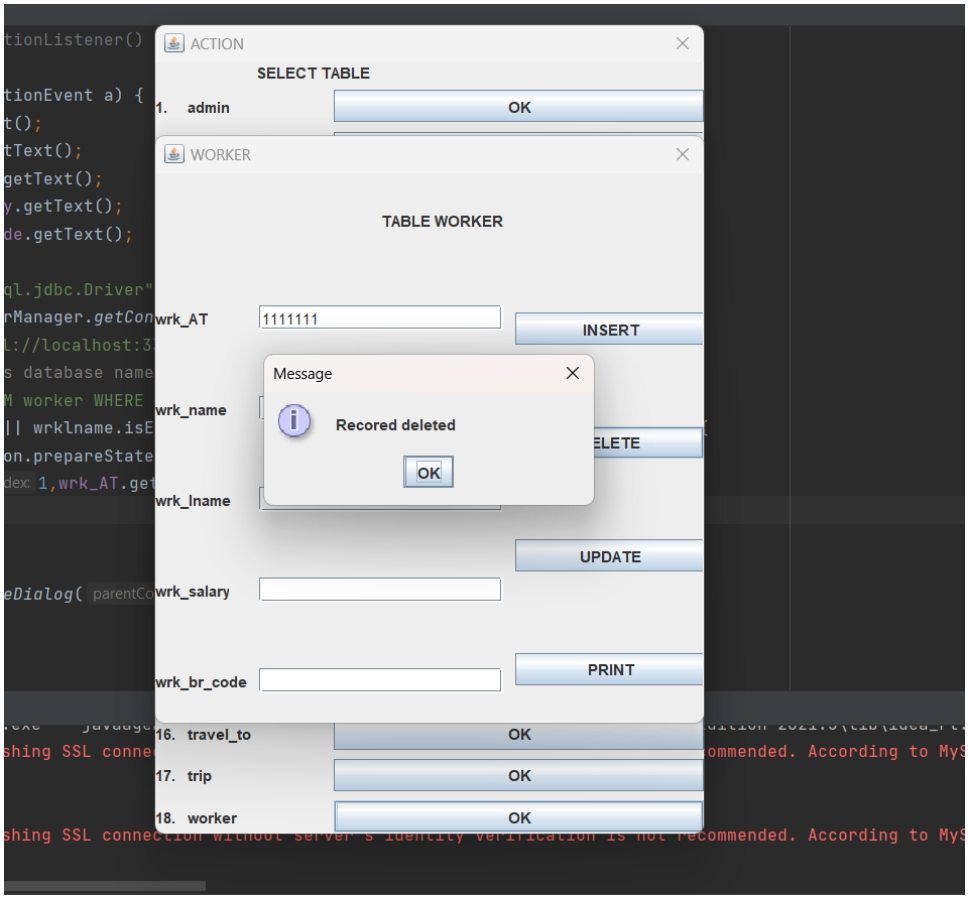

ACTION
SELECT TABLE
1. admin
OK
WORKER
TABLE WORKER
wrk_AT
1111111
INSERT
wrk_name
anna
DELETE
wrk_lname
pentagiwtissa
UPDATE
wrk_salary
PRINT
wrk_br_code
16. travel_to
OK
17. trip
OK
18. worker
OK

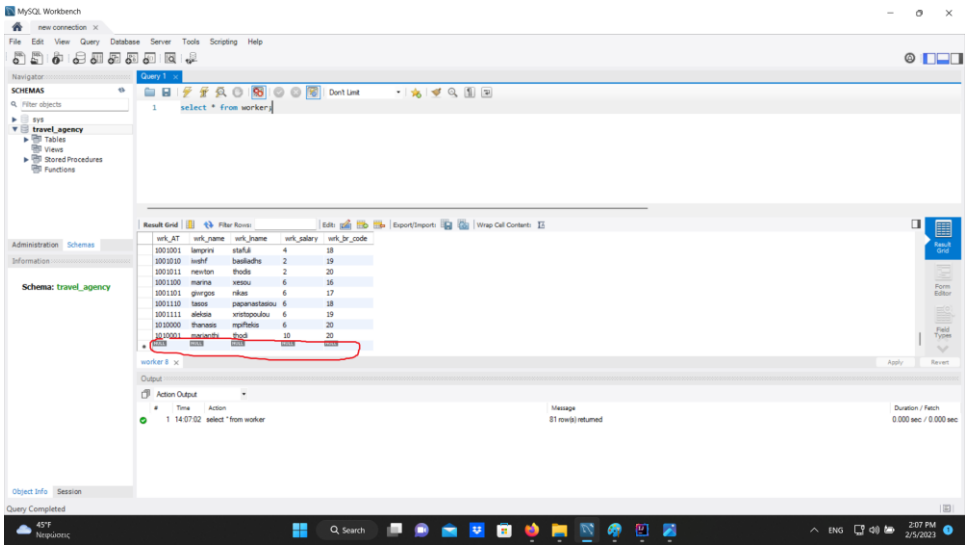

ACTION
SELECT TABLE
1. admin
OK
WORKER
TABLE WORKER
wrk_AT
1111111
INSERT
Message
Recored updated
OK
wrk_name
wrk_lname
UPDATE
wrk_salary
PRINT
wrk_br_code
16. travel_to
OK
17. trip
OK
18. worker
OK


select * from worker;
Result Grid
wrk_AT wrk_name wrk_lname wrk_salary wrk_br_code
1001010 iwshf basiliadhs 2 19
1001011 newton thodis 2 20
1001100 marina xesou 6 16
1001101 giwrgos nikas 6 17
1001110 tasos papanastasiou 6 18
1001111 aleksia xristopoulou 6 19
1010000 thanasis mpiftekis 6 20
1010001 marianthi thodi 10 20
1111111 anna pentagiwtissa 1
worker 7

//delete


ACTION
SELECT TABLE
1. admin
OK
WORKER
TABLE WORKER
wrk_AT
1111111
INSERT
wrk_name
DELETE
wrk_lname
UPDATE
wrk_salary
PRINT
wrk_br_code
16. travel_to
OK
17. trip
OK
18. worker
OK





Όμοια λοιπόν για όλους τους υπόλοιπους πίνακες.