ALTER TABLE Περιέχει DROP FOREIGN KEY Περιέχει\_fk1

ALTER TABLE Περιέχει ADD CONSTRAINT Περιέχει\_fk1 FOREIGN KEY (Κωδ\_Βιβλιοθήκης) REFERENCES Αντίτυπο (Αριθμός\_αντιτύπου) ON UPDATE CASCADE ON DELETE CASCADE;

ALTER TABLE `Περιέχει`

ADD CONSTRAINT `Περιέχει\_fk0` FOREIGN KEY (`ISBN`) REFERENCES `Αντίτυπο` (`ISBN`) ON UPDATE CASCADE

SELECT \*

FROM `Δανεισμός`

WHERE `Ημερομηνία\_που\_επιστράφηκε` is not NULL AND Βιβλιοθήκη\_καταχώρησης\_επιστροφής=1

(

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδικος\_βιβλιοθήκης\_αντιτύπου` as `Κωδ\_Βιβλιοθήκης`,`Ημερομηνία\_που\_επιστράφηκε` as return\_date

FROM `Δανεισμός`

WHERE `Ημερομηνία\_που\_επιστράφηκε` is not NULL AND Βιβλιοθήκη\_καταχώρησης\_επιστροφής=1 ORDER BY `Ημερομηνία\_που\_επιστράφηκε`

)

UNION

(

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδ\_Βιβλιοθήκης`,`Ημερομηνία` as return\_date

FROM Μεταφορά JOIN Περιέχει ON `Κωδικός\_μεταφοράς`=`Κωδ\_μεταφοράς`

WHERE `Κατάσταση\_παραλαβής`=1

)

SELECT \*

FROM

((

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδικος\_βιβλιοθήκης\_αντιτύπου` as `Κωδ\_Βιβλιοθήκης`,`Ημερομηνία\_που\_επιστράφηκε` as return\_date

FROM `Δανεισμός`

WHERE `Ημερομηνία\_που\_επιστράφηκε` is not NULL AND Βιβλιοθήκη\_καταχώρησης\_επιστροφής=1 ORDER BY `Ημερομηνία\_που\_επιστράφηκε`

)

UNION

(

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδ\_Βιβλιοθήκης`,`Ημερομηνία` as return\_date

FROM Μεταφορά JOIN Περιέχει ON `Κωδικός\_μεταφοράς`=`Κωδ\_μεταφοράς`

WHERE `Κατάσταση\_παραλαβής`=1

))

WHERE 1-- (`ISBN`,`Αρ\_αντιτύπου`,`Κωδ\_Βιβλιοθήκης`,`Ημερομηνία` as return\_date) NOT IN

(

SELECT \*,'a' as 'b' FROM Έντυπο

)

SELECT \*

FROM

(select \*

FROM (

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδικος\_βιβλιοθήκης\_αντιτύπου` as `Κωδ\_Βιβλιοθήκης`,`Ημερομηνία\_που\_επιστράφηκε` as return\_date

FROM `Δανεισμός`

WHERE `Ημερομηνία\_που\_επιστράφηκε` is not NULL AND Βιβλιοθήκη\_καταχώρησης\_επιστροφής=1 ORDER BY `Ημερομηνία\_που\_επιστράφηκε`

) as `returned`

UNION

(

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδ\_Βιβλιοθήκης`,`Ημερομηνία` as return\_date

FROM Μεταφορά JOIN Περιέχει ON `Κωδικός\_μεταφοράς`=`Κωδ\_μεταφοράς`

WHERE `Κατάσταση\_παραλαβής`=1

)) as `here`

RIGHT OUTER JOIN Έντυπο ON here.ISBN=Έντυπο.ISBN

SELECT \*

FROM

(select \*

FROM (

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδικος\_βιβλιοθήκης\_αντιτύπου` as `Κωδ\_Βιβλιοθήκης`,`Ημερομηνία\_που\_επιστράφηκε` as return\_date

FROM `Δανεισμός`

WHERE `Ημερομηνία\_που\_επιστράφηκε` is not NULL AND Βιβλιοθήκη\_καταχώρησης\_επιστροφής=1 ORDER BY `Ημερομηνία\_που\_επιστράφηκε`

) as `returned`

UNION

(

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδ\_Βιβλιοθήκης`,`Ημερομηνία` as return\_date

FROM Μεταφορά JOIN Περιέχει ON `Κωδικός\_μεταφοράς`=`Κωδ\_μεταφοράς`

WHERE `Κατάσταση\_παραλαβής`=1

)) as `books\_came\_here`

RIGHT OUTER JOIN

Αντίτυπο ON books\_came\_here.ISBN=Αντίτυπο.ISBN and books\_came\_here.Αρ\_αντιτύπου=Αντίτυπο.Αριθμός\_αντιτύπου and books\_came\_here.Κωδ\_Βιβλιοθήκης=Αντίτυπο.Κωδικός\_βιβλιοθήκης

WHERE ((books\_came\_here.return\_date IS NULL AND Κωδικός\_βιβλιοθήκης=1) OR books\_came\_here.return\_date IS NOT NULL)

Σχεδόν τελικό(δες κάτω κάτω έχω γράψει ένα κείμενο που το εξηγεί λίγο,προσπαθώ να βρώ την λογική ακριβώς)

SELECT \*

FROM

(

SELECT Αντίτυπο.ISBN,Αντίτυπο.Αριθμός\_αντιτύπου,Αντίτυπο.Κωδικός\_βιβλιοθήκης,return\_date

FROM

(select \*

FROM (

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδικός\_βιβλιοθήκης\_αντιτύπου` as `Κωδ\_Βιβλιοθήκης`,`Ημερομηνία\_που\_επιστράφηκε` as return\_date

FROM `Δανεισμός`

WHERE `Ημερομηνία\_που\_επιστράφηκε` is not NULL AND Βιβλιοθήκη\_καταχώρησης\_επιστροφής=1 ORDER BY `Ημερομηνία\_που\_επιστράφηκε`

) as `returned`

UNION

(

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδ\_Βιβλιοθήκης`,`Ημερομηνία` as return\_date

FROM Μεταφορά JOIN Περιέχει ON `Κωδικός\_μεταφοράς`=`Κωδ\_μεταφοράς`

WHERE `Κατάσταση\_παραλαβής`=1

)) as `books\_came\_here`

RIGHT OUTER JOIN

Αντίτυπο ON books\_came\_here.ISBN=Αντίτυπο.ISBN and books\_came\_here.Αρ\_αντιτύπου=Αντίτυπο.Αριθμός\_αντιτύπου and books\_came\_here.Κωδ\_Βιβλιοθήκης=Αντίτυπο.Κωδικός\_βιβλιοθήκης

WHERE ((books\_came\_here.return\_date IS NULL AND Κωδικός\_βιβλιοθήκης=1) OR books\_came\_here.return\_date IS NOT NULL)

) as `books\_here`

left OUTER JOIN

(SELECT \*

FROM Δανεισμός

WHERE Βιβλιοθήκη\_καταχώρησης\_δανεισμού=1

) as `borrowed`

ON books\_here.ISBN=borrowed.ISBN AND books\_here.Αριθμός\_αντιτύπου=borrowed.Αρ\_αντιτύπου

libraries

SELECT \*

FROM (Αρ\_Τηλ\_Βιβλιοθήκης NATURAL JOIN Βιβλιοθήκη)

Υπολογισμός εναπομείναντα χρόνου συνδρομής ([users-staff.html](http://127.0.0.1:5500/users-staff.html))

τελικο

SELECT Κωδ\_μέλους,

TIMESTAMPDIFF(DAY,date\_add(NOW(),interval 2 HOUR),max(date\_add(Ημερομηνία\_έναρξης, interval Διάρκεια MONTH))) AS Ημέρες\_ακόμα, max(date\_add(Ημερομηνία\_έναρξης, interval Διάρκεια MONTH)) as `Ημερομηνία\_λήξης`, max(date\_add(Ημερομηνία\_έναρξης, interval Διάρκεια MONTH)) - CURRENT\_TIMESTAMP() >0 as Ενεργή

FROM Συνδρομή as s,Επιλογές\_Συνδρομής as c

WHERE s.Κωδ\_συνδρομής=c.Κωδικός\_συνδρομής

GROUP BY Κωδ\_μέλους

//SELECT Κωδικός\_μέλους,

// TIMESTAMPDIFF(DAY,date\_add(NOW(),interval 2 HOUR),Ημερομηνία\_λήξης) AS DAYS,

//TIMESTAMPDIFF(HOUR,date\_add(NOW(),interval 2 //HOUR),Ημερομηνία\_λήξης)-TIMESTAMPDIFF(DAY,date\_add(NOW(),interval 2 //HOUR),Ημερομηνία\_λήξης)\*24 AS HOURS,

//TIMESTAMPDIFF(MINUTE,date\_add(NOW(),interval 2 //HOUR),Ημερομηνία\_λήξης)-TIMESTAMPDIFF(HOUR,date\_add(NOW(),interval 2 //HOUR),Ημερομηνία\_λήξης)\*60 AS MINUTES

//FROM Συνδρομή JOIN Μέλος ON Κωδ\_μέλους=Κωδικός\_μέλους

//WHERE Ημερομηνία\_λήξης>CURRENT\_TIMESTAMP()

##################################################################

Κατι που προσπαθησα για το παραπάνω απλά το βάζω αν προκύψει κάτι λάθος με το παραπάνω

SELECT Κωδικός\_μέλους,datediff(Ημερομηνία\_λήξης,CURRENT\_TIMESTAMP()) as days,

FLOOR(TIMESTAMPDIFF(HOUR,CURRENT\_TIMESTAMP(),Ημερομηνία\_λήξης)/24) AS DAY,

abs(24-abs(24-TIMESTAMPDIFF(HOUR,date\_add(CURRENT\_TIMESTAMP(),interval datediff(Ημερομηνία\_λήξης,CURRENT\_TIMESTAMP()) DAY),Ημερομηνία\_λήξης))-2) AS JJ,

TIMESTAMPDIFF(second,

date\_add(

date\_add(CURRENT\_TIMESTAMP(),interval datediff(Ημερομηνία\_λήξης,CURRENT\_TIMESTAMP()) DAY),interval -TIMESTAMPDIFF(HOUR,date\_add(CURRENT\_TIMESTAMP(),interval datediff(Ημερομηνία\_λήξης,CURRENT\_TIMESTAMP()) DAY),Ημερομηνία\_λήξης)-2 HOUR),Ημερομηνία\_λήξης) AS JJ1

#(ceiling(TIMESTAMPDIFF(HOUR,CURRENT\_TIMESTAMP,Ημερομηνία\_λήξης)/24)-(TIMESTAMPDIFF(HOUR,CURRENT\_TIMESTAMP,Ημερομηνία\_λήξης))/24) AS HOURS #FLOOR((CONVERT(Ημερομηνία\_λήξης,date)-CONVERT(CURRENT\_TIMESTAMP,date))) AS minutes

FROM Συνδρομή JOIN Μέλος ON Κωδ\_μέλους=Κωδικός\_μέλους

WHERE Ημερομηνία\_λήξης>CURRENT\_TIMESTAMP()

##################################################################

Για αναζήτηση όπου εντός των %% θα μπεί αναζήτηση στην μπάρα αναζήτησης

SELECT Τίτλος FROM Έντυπο WHERE Τίτλος like "%%"

//Προσθήκη ημερομηνίας λήξης στη συνδρομή, ανάλογα την συνδρομή

//UPDATE Συνδρομή as s,Επιλογές\_Συνδρομής as c

//SET Ημερομηνία\_λήξης=date\_add(Ημερομηνία\_έναρξης, interval Διάρκεια MONTH)

//WHERE s.Κωδ\_συνδρομής=c.Κωδικός\_συνδρομής ;

—> σαν πίνακας (για να βγάλουμε το πεδιο ημερομηνια ληξης):

SELECT Κωδ\_μέλους,max(date\_add(Ημερομηνία\_έναρξης, interval Διάρκεια MONTH)) as Ημερομηνία\_λήξης\_συνδρομής

FROM Συνδρομή as s,Επιλογές\_Συνδρομής as c

WHERE s.Κωδ\_συνδρομής=c.Κωδικός\_συνδρομής

GROUP BY Κωδ\_μέλους

Επιστρέφει ημερομηνία λήξης και στοιχεία μέλους

SELECT Κωδικός\_Μέλους, Όνομα, Επίθετο, email, Ημερομηνία\_έναρξης, Κωδ\_συνδρομής, Διάρκεια, date\_add(Ημερομηνία\_έναρξης, interval Διάρκεια MONTH) as Ημερομηνία\_λήξης, Διάρκεια\_δανεισμού, Όριο\_δανεισμών, date\_add(Ημερομηνία\_έναρξης, interval Διάρκεια MONTH) > now() as Ενεργή

FROM

(SELECT Κωδ\_μέλους, max(Ημερομηνία\_έναρξης) as Ημερομηνία\_έναρξης

FROM Συνδρομή JOIN Επιλογές\_Συνδρομής ON Συνδρομή.Κωδ\_συνδρομής=Επιλογές\_Συνδρομής.Κωδικός\_συνδρομής

GROUP BY Κωδ\_μέλους) as syndromes

JOIN Συνδρομή USING(Κωδ\_μέλους,Ημερομηνία\_έναρξης) JOIN Επιλογές\_Συνδρομής ON Κωδ\_συνδρομής=Κωδικός\_συνδρομής

RIGHT OUTER JOIN Μέλος ON Κωδ\_μέλους=Κωδικός\_Μέλους

Υπολογισμός διάρκειας δανεισμού, ανάλογα με την συνδρομή που έχει επιλέξει το μέλος

UPDATE Δανεισμός,Συνδρομή AS S ,Επιλογές\_Συνδρομής

SET Ημερομηνία\_Επιστροφής=date\_add(Ημερομηνία\_δανεισμού, interval Διάρκεια\_δανεισμού DAY )

WHERE Δανεισμός.Κωδικός\_μέλους=S.Κωδ\_μέλους

AND Διάρκεια\_δανεισμού IN ( SELECT Επιλογές\_Συνδρομής.Διάρκεια\_δανεισμού

FROM Επιλογές\_Συνδρομής

WHERE S.Κωδ\_συνδρομής=Κωδικός\_συνδρομής)

Αναλυτικές πληροφορίες για επιβαρύνσεις μέλους

SELECT DISTINCT nt.Κωδικός\_μέλους,state,Κωδικός\_δανεισμού,FLOOR(sc.Επιβάρυνση\_καθυστέρησης\_ασυνέπειας\*datediff(CURRENT\_TIMESTAMP(),Δανεισμός.Ημερομηνία\_επιστροφής)) as extra

FROM(

SELECT m.Κωδικός\_μέλους,CEILING(COUNT(\*)/(COUNT(\*)+1)) AS state

FROM Μέλος as m,Δανεισμός as d

WHERE m.Κωδικός\_μέλους=d.Κωδικός\_μέλους AND d.Ημερομηνία\_που\_επιστράφηκε IS NULL AND now()>d.Ημερομηνία\_επιστροφής

group by m.Κωδικός\_μέλους) as nt,Δανεισμός, Επιλογές\_Συνδρομής as sc , Συνδρομή as s

WHERE nt.Κωδικός\_μέλους=Δανεισμός.Κωδικός\_μέλους AND sc.Κωδικός\_συνδρομής=s.Κωδ\_συνδρομής AND s.Κωδ\_μέλους=nt.Κωδικός\_μέλους

#`Κατάσταση\_Επιβάρυνσης` ,`#Πηγές` ,`Ποσό\_επιβάρυνσης\_δανεισμού`

Ίσως προσθέσουμε σαν απαίτηση ότι όταν κάποιος ξεπεράσει τα 20 ευρώ χρωστούμενα συνολικά(ένα ρεαλιστικό ποσό) και έχει κωδικό συνδρομής απο 2 και πάνω,να του εμφανίζεται ένα κείμενο να ανανεώσει την συνδρομή για να ξεχρεώσει. Και αυτό ίσως συμφέρει μία βιβλιοθήκη γιατί αν κάποιος καθυστερεί

μια φορά, ίσως καθυστερεί τακτικά, αρα θα ξεχρεώσει με 4 ευρώ(αν επιλέξει την 1η συνδρομή), αλλα θα αυξηθεί μετά η επιβάρυνση , αφού είχε >2 κωδικό συνδρομής και τώρα το πιο πιθανό να επιλέξει την 1,(το βαζουμε στο τέλος , εκτός αν φανεί να είναι απλό,και δουλεύει καλά).

+πως αποκλειεται καποιος…

SELECT SUM(extra)

FROM(

SELECT DISTINCT nt.Κωδικός\_μέλους,state,Κωδικός\_δανεισμού,FLOOR(sc.Επιβάρυνση\_καθυστέρησης\_ασυνέπειας\*datediff(CURRENT\_TIMESTAMP(),Δανεισμός.Ημερομηνία\_επιστροφής)) AS extra

FROM(

SELECT m.Κωδικός\_μέλους,CEILING(COUNT(\*)/(COUNT(\*)+1)) AS state

FROM Μέλος as m,Δανεισμός AS d

WHERE m.Κωδικός\_μέλους=d.Κωδικός\_μέλους AND d.Ημερομηνία\_που\_επιστράφηκε IS NULL AND now()>d.Ημερομηνία\_επιστροφής

GROUP by m.Κωδικός\_μέλους) AS nt,Δανεισμός, Επιλογές\_Συνδρομής AS sc , Συνδρομή AS s

WHERE nt.Κωδικός\_μέλους=Δανεισμός.Κωδικός\_μέλους AND sc.Κωδικός\_συνδρομής=s.Κωδ\_συνδρομής AND s.Κωδ\_μέλους=nt.Κωδικός\_μέλους) AS Main

GROUP BY Main.Κωδικός\_μέλους

HAVING SUM(extra)>20

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%\

ολα τα αντιτυπα που εχουν επιστραφεί στην βιβλιοθήκη 1 !(δεν αποκλείεται όμως να έχει ξαναδανειστεί απο κάποιον,δηλαδη να εμπλεκεται σε αλλον δανεισμό)

UNION

ολα τα αντίτυπα που έχουν φτάσει με μεταφορά που έχει ώς προορισμό την βιβλιοθήκη 1 !(δεν αποκλείεται αυτό το αντίτυπο να βρίσκεται σε άλλη βιβλιοθήκη(με άλλη μεταφορά)

αρα εχουμε αυτα που έχουν μεταφερθει και επιστραφει στην βιβλιοθηκη 1 και υποτιθεται οτι υπαρχουν(μπορεί και να μην υπαρχουν ομως (στην 1))

right outer join με όλα τα αντίτυπα οπότε παίρνουμε όλα τα υπολοιπα αντίτυπα

αλλα κρατάμε απο τα παραπάνω, [αυτά που δεν έχουν επιστραφεί και ανήκουν στην βιβλιοθήκη] δηλαδή ότι αντίτυπο ανήκει στην 1 πέραν αυτών που υπαρχουν

στον left πίνακα,

και αυτά που έχουν επιστραφεί δηλαδή αυτά που υπάρχουν στον πάνω πίνακα

οπότε παίρνουμε 7 γραμμές(που περιέχουν ό,τι υποτίθεται οτι υπαρχει στην βιβλιοθήκη 1 ως αποτελεσμα μιας μεταφοράς η επιστροφής σε αυτήν

και ότι υποτίθεται ότι υπάρχει αφού ανήκει στην 1)

κάνουμε left outer join όλον τον παραπάνω πίνακα με όλα τα αντίτυπα που είναι δανεισμένα απο την βιβλιοθήκη 1

ON ίδιο αντιτυπο και isbn αρα αυτά που είναι δανεισμένα αυτήν την στιγμή και ανήκουν στην 1 φαίνονται στον τελικο πίνακα

Άρα κάνοντας το left outer join, ό,τι υποτίθεται οτι βρίσκεται στην 1 ενώνεται με τον τυχόν (όχι τρέχων όμως) δανεισμό απο την 1, άρα λύνεται το πρώτο πρόβλημα στο πάνω μέρος του union απότε φαίνεται πλέον αν αυτό το αντίτυπο που επιστράφηκε στην 1, εμπλέκεται ή όχι σε άλλον μετέπειτα δανεισμό.

Πρέπει να λύσουμε και το πρόβλημα με τον δεύτερο πίνακα στο union.

.οΟ(μπορεί κάποιος να δανειστεί απο την 1 , να επιστρέψει στην 2, και άρα θα γίνει μεταφορά απο την 2 στην 1 του αντιτύπου άν δεν υπάρχει κάποια αίτηση με κατάσταση 0. )

Πρόσθεσα και αυτό



Και βγάζει αυτό 

%%%%%%

Όμως έχει επιστραφεί. Άρα προσθέτω περιορισμό για τρέχων δανεισμούς

Οτι εβγαζε με το αρχικό

%%%%%%%%%%



+

+

SELECT \*

FROM

(SELECT \*

FROM

(

SELECT Αντίτυπο.ISBN as `ISBN1`,Αντίτυπο.Αριθμός\_αντιτύπου,Αντίτυπο.Κωδικός\_βιβλιοθήκης,return\_date

FROM

(select \*

FROM (

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδικός\_βιβλιοθήκης\_αντιτύπου` as `Κωδ\_Βιβλιοθήκης`,`Ημερομηνία\_που\_επιστράφηκε` as return\_date

FROM `Δανεισμός`

WHERE `Ημερομηνία\_που\_επιστράφηκε` is not NULL AND Βιβλιοθήκη\_καταχώρησης\_επιστροφής=1 ORDER BY `Ημερομηνία\_που\_επιστράφηκε`

) as `returned`

UNION

(

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδ\_Βιβλιοθήκης`,`Ημερομηνία` as return\_date

FROM Μεταφορά JOIN Περιέχει ON `Κωδικός\_μεταφοράς`=`Κωδ\_μεταφοράς`

WHERE `Κατάσταση\_παραλαβής`=1 and Κωδικός\_Βιβλιοθήκης\_προορισμού=1

)) as `books\_came\_here`

RIGHT OUTER JOIN

Αντίτυπο ON books\_came\_here.ISBN=Αντίτυπο.ISBN and books\_came\_here.Αρ\_αντιτύπου=Αντίτυπο.Αριθμός\_αντιτύπου and books\_came\_here.Κωδ\_Βιβλιοθήκης=Αντίτυπο.Κωδικός\_βιβλιοθήκης

WHERE ((books\_came\_here.return\_date IS NULL AND Κωδικός\_βιβλιοθήκης=1) OR books\_came\_here.return\_date IS NOT NULL)

) as `books\_here`

left OUTER JOIN

(SELECT \*

FROM Δανεισμός

WHERE Βιβλιοθήκη\_καταχώρησης\_δανεισμού=1 AND Ημερομηνία\_που\_επιστράφηκε IS NULL

) as `borrowed`

ON ISBN1=borrowed.ISBN AND books\_here.Αριθμός\_αντιτύπου=borrowed.Αρ\_αντιτύπου) as `fstage`

LEFT OUTER JOIN

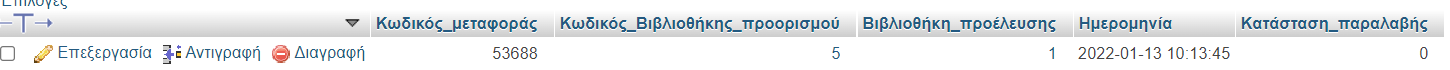
(SELECT \*

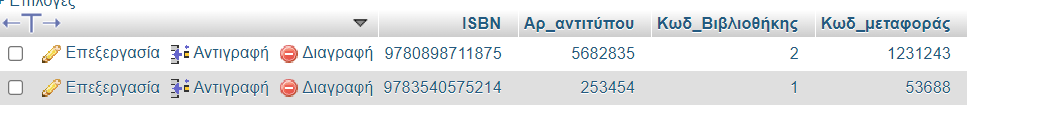
FROM Μεταφορά,Περιέχει

WHERE `Κωδικός\_μεταφοράς`=`Κωδ\_μεταφοράς` AND Βιβλιοθήκη\_προέλευσης=1 AND Κατάσταση\_παραλαβής=0 ) AS `mstage`

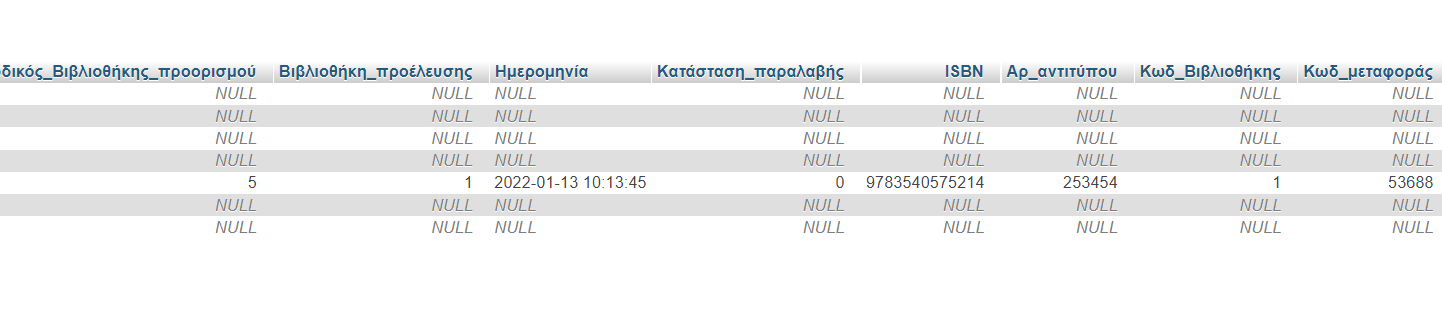
ON fstage.Αριθμός\_αντιτύπου=mstage.Αρ\_αντιτύπου

Σε μία μεταφορά που είχε φύγει απο την 1, και περιείχε ένα αντίτυπο απο αυτά που έβγαζε το προηγούμενο query, το αντίτυπο εμφανιζόταν όπως πριν. Τώρα προσθέτω και την extra πληροφορια της μεταφοράς αν το αντίτυπο είναι σε μεταφορά που έχει φύγει απο την βιβλιοθήκη









%%%%%%%%%%%%%%%%%%%%%%%%

Αναζήτηση διαθέσιμων εντύπων

SELECT DISTINCT 1 AS `Κωδικός\_Βιβλιοθήκης`,ISBN1 AS `ISBN`, Τίτλος, Κατηγορία,Συγγραφέας

FROM(

(SELECT \*

FROM

(

SELECT Αντίτυπο.ISBN as `ISBN1`,Αντίτυπο.Αριθμός\_αντιτύπου,Αντίτυπο.Κωδικός\_βιβλιοθήκης,return\_date

FROM

(select \*

FROM (

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδικός\_βιβλιοθήκης\_αντιτύπου` as `Κωδ\_Βιβλιοθήκης`,`Ημερομηνία\_που\_επιστράφηκε` as return\_date

FROM `Δανεισμός`

WHERE `Ημερομηνία\_που\_επιστράφηκε` is not NULL AND Βιβλιοθήκη\_καταχώρησης\_επιστροφής=1 ORDER BY `Ημερομηνία\_που\_επιστράφηκε`

) as `returned`

UNION

(

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδ\_Βιβλιοθήκης`,`Ημερομηνία` as return\_date

FROM Μεταφορά JOIN Περιέχει ON `Κωδικός\_μεταφοράς`=`Κωδ\_μεταφοράς`

WHERE `Κατάσταση\_παραλαβής`=1 and Κωδικός\_Βιβλιοθήκης\_προορισμού=1

)) as `books\_came\_here`

RIGHT OUTER JOIN

Αντίτυπο ON books\_came\_here.ISBN=Αντίτυπο.ISBN and books\_came\_here.Αρ\_αντιτύπου=Αντίτυπο.Αριθμός\_αντιτύπου and books\_came\_here.Κωδ\_Βιβλιοθήκης=Αντίτυπο.Κωδικός\_βιβλιοθήκης

WHERE ((books\_came\_here.return\_date IS NULL AND Κωδικός\_βιβλιοθήκης=1) OR books\_came\_here.return\_date IS NOT NULL)

) as `books\_here`

left OUTER JOIN

(SELECT \*

FROM Δανεισμός

WHERE Βιβλιοθήκη\_καταχώρησης\_δανεισμού=1 AND Ημερομηνία\_που\_επιστράφηκε IS NULL

) as `borrowed`

ON ISBN1=borrowed.ISBN AND books\_here.Αριθμός\_αντιτύπου=borrowed.Αρ\_αντιτύπου) as `fstage`

LEFT OUTER JOIN

(SELECT \*

FROM Μεταφορά,Περιέχει

WHERE `Κωδικός\_μεταφοράς`=`Κωδ\_μεταφοράς` AND Βιβλιοθήκη\_προέλευσης=1 AND Κατάσταση\_παραλαβής=0 ) AS `mstage`

ON fstage.Αριθμός\_αντιτύπου=mstage.Αρ\_αντιτύπου), Έντυπο, Περιλαμβλανει,Συγγραφείς

WHERE (Κωδ\_μεταφοράς IS NULL AND Κωδικός\_δανεισμού IS NULL) AND Έντυπο.ISBN=isbn1 AND Έντυπο.ISBN=Περιλαμβλανει.ISBN

AND Συγγραφείς.ISBN=ISBN1

# AND ISBN1 LIKE "%INPUT%" AND Συγγραφέας LIKE "%INPUT%" AND Κατηγορία LIKE "%INPUT%" AND Κωδικός\_Βιβλιοθήκης LIKE "%1%"

Βιβλία που βρίσκονται στην βιλιοθήκη 1 (search, book.html, book-staff.html)

SELECT books\_been\_here.ISBN, books\_been\_here.Αριθμός\_αντιτύπου, books\_been\_here.Κωδικός\_βιβλιοθήκης, max(return\_date), max(date\_moved), max(date\_left)

FROM

(

SELECT books\_here.ISBN, books\_here.Αριθμός\_αντιτύπου, books\_here.Κωδικός\_βιβλιοθήκης, return\_date, date\_left

FROM

(

SELECT Αντίτυπο.ISBN,Αντίτυπο.Αριθμός\_αντιτύπου,Αντίτυπο.Κωδικός\_βιβλιοθήκης, return\_date

FROM

(select \*

FROM (

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδικός\_βιβλιοθήκης\_αντιτύπου` as `Κωδ\_Βιβλιοθήκης`,`Ημερομηνία\_που\_επιστράφηκε` as return\_date

FROM `Δανεισμός`

WHERE `Ημερομηνία\_που\_επιστράφηκε` is not NULL AND Βιβλιοθήκη\_καταχώρησης\_επιστροφής=1 ORDER BY `Ημερομηνία\_που\_επιστράφηκε`

) as `returned`

UNION

(

SELECT `ISBN`,`Αρ\_αντιτύπου`,`Κωδ\_Βιβλιοθήκης`,`Ημερομηνία` as return\_date

FROM Μεταφορά JOIN Περιέχει ON `Κωδικός\_μεταφοράς`=`Κωδ\_μεταφοράς`

WHERE `Κατάσταση\_παραλαβής`=1 AND Κωδικός\_Βιβλιοθήκης\_προορισμού=1

)) as `books\_came\_here`

RIGHT OUTER JOIN

Αντίτυπο ON books\_came\_here.ISBN=Αντίτυπο.ISBN and books\_came\_here.Αρ\_αντιτύπου=Αντίτυπο.Αριθμός\_αντιτύπου and books\_came\_here.Κωδ\_Βιβλιοθήκης=Αντίτυπο.Κωδικός\_βιβλιοθήκης

WHERE ((books\_came\_here.return\_date IS NULL AND Κωδικός\_βιβλιοθήκης=1) OR books\_came\_here.return\_date IS NOT NULL)

) as `books\_here`

LEFT OUTER JOIN

(SELECT ISBN, Αρ\_αντιτύπου, Ημερομηνία\_δανεισμού as date\_left, Κωδικός\_βιβλιοθήκης\_αντιτύπου

FROM Δανεισμός

WHERE Βιβλιοθήκη\_καταχώρησης\_δανεισμού=1

) as `borrowed`

ON books\_here.ISBN=borrowed.ISBN AND books\_here.Αριθμός\_αντιτύπου=borrowed.Αρ\_αντιτύπου AND books\_here.Κωδικός\_βιβλιοθήκης=borrowed.Κωδικός\_βιβλιοθήκης\_αντιτύπου

) as `books\_been\_here`

LEFT OUTER JOIN

(SELECT ISBN, Κωδ\_βιβλιοθήκης, Αρ\_αντιτύπου, Ημερομηνία as date\_moved

FROM Μεταφορά,Περιέχει

WHERE `Κωδικός\_μεταφοράς`=`Κωδ\_μεταφοράς` AND Βιβλιοθήκη\_προέλευσης=1) AS `moved`

ON books\_been\_here.`ISBN`=moved.`ISBN` AND books\_been\_here.Αριθμός\_αντιτύπου=moved.Αρ\_αντιτύπου AND books\_been\_here.Κωδικός\_βιβλιοθήκης=moved.Κωδ\_βιβλιοθήκης

GROUP BY books\_been\_here.ISBN, books\_been\_here.Αριθμός\_αντιτύπου, books\_been\_here.Κωδικός\_βιβλιοθήκης

HAVING (

(

( ( max(return\_date) > max(date\_left) )

OR

max(date\_left)=NULL )

AND

( ( max(return\_date) > max(date\_moved) )

OR

max(date\_moved)=NULL )

)

OR (max(date\_left) is NULL AND max(date\_moved) IS NULL)

)

**Που βρίσκεται το κάθε αντίτυπο NULL σε μεταφορά ή δανεισμό**

– CREATE VIEW Θέσεις\_βιβλίων

– AS

SELECT all\_books.ISBN, all\_books.Αριθμός\_αντιτύπου, all\_books.Κωδικός\_Βιβλιοθήκης,

CASE

WHEN (Δανεισμός.Κωδικός\_δανεισμού is NULL and Κωδικός\_μεταφοράς is NULL) THEN Κωδικός\_Βιβλιοθήκης

WHEN (Δανεισμός.Κωδικός\_δανεισμού is NOT NULL and Κωδικός\_μεταφοράς is NULL) THEN Βιβλιοθήκη\_καταχώρησης\_επιστροφής

-- CASE

-- WHEN (Βιβλιοθήκη\_καταχώρησης\_επιστροφής IS NULL) THEN NULL

-- WHEN (Βιβλιοθήκη\_καταχώρησης\_επιστροφής IS NOT NULL) THEN Βιβλιοθήκη\_καταχώρησης\_επιστροφής

-- END

WHEN (Δανεισμός.Κωδικός\_δανεισμού is NULL and Κωδικός\_μεταφοράς is NOT NULL) THEN

CASE

WHEN (Κατάσταση\_παραλαβής=0) THEN NULL

WHEN (Κατάσταση\_παραλαβής=1) THEN Κωδικός\_Βιβλιοθήκης\_προορισμού

END

WHEN (Δανεισμός.Κωδικός\_δανεισμού is NOT NULL and Κωδικός\_μεταφοράς is NOT NULL) THEN

CASE

WHEN (Ημερομηνία\_που\_επιστράφηκε>=date\_moved) THEN Βιβλιοθήκη\_καταχώρησης\_επιστροφής

WHEN (Ημερομηνία\_που\_επιστράφηκε<date\_moved) THEN Κωδικός\_Βιβλιοθήκης\_προορισμού

END

END AS Βιβλιοθήκη\_τώρα,

IF (Δανεισμός.Κωδικός\_δανεισμού is NULL and Κωδικός\_μεταφοράς is NOT NULL AND Κατάσταση\_παραλαβής=0, Κωδικός\_Βιβλιοθήκης\_προορισμού, NULL) AS Μεταφέρεται\_σε,

IF (Δανεισμός.Κωδικός\_δανεισμού is NOT NULL and Κωδικός\_μεταφοράς is NULL AND Βιβλιοθήκη\_καταχώρησης\_επιστροφής is NULL, Δανεισμός.Κωδικός\_μέλους, NULL) AS Δανεισμένο\_σε

FROM

( -- Πρώτα βρίσκω τα τελευταία απο τελευταιους δανεισμούς και μεταφορές και μετά βρίσκω λεπτομέρειες

SELECT books\_and\_borrows.ISBN, books\_and\_borrows.Αριθμός\_αντιτύπου, books\_and\_borrows.Κωδικός\_Βιβλιοθήκης, max(Κωδικός\_δανεισμού) as Κωδικός\_δανεισμού, max(Κωδικός\_μεταφοράς) as Κωδικός\_μεταφοράς

FROM (

SELECT Αντίτυπο.ISBN, Αντίτυπο.Αριθμός\_αντιτύπου, Αντίτυπο.Κωδικός\_Βιβλιοθήκης, Κωδικός\_δανεισμού, Ημερομηνία\_που\_επιστράφηκε

FROM Αντίτυπο LEFT JOIN Δανεισμός ON Αντίτυπο.ISBN=Δανεισμός.ISBN AND Αντίτυπο.Αριθμός\_αντιτύπου=Δανεισμός.Αρ\_αντιτύπου AND Αντίτυπο.Κωδικός\_Βιβλιοθήκης=Δανεισμός.Κωδικός\_βιβλιοθήκης\_αντιτύπου) as books\_and\_borrows

LEFT JOIN (

SELECT ISBN, Κωδ\_βιβλιοθήκης, Αρ\_αντιτύπου, Κωδικός\_μεταφοράς

FROM Μεταφορά,Περιέχει

WHERE Κωδικός\_μεταφοράς=Κωδ\_μεταφοράς) AS moved

ON books\_and\_borrows.ISBN=moved.ISBN and books\_and\_borrows.Αριθμός\_αντιτύπου=moved.Αρ\_αντιτύπου AND books\_and\_borrows.Κωδικός\_Βιβλιοθήκης=moved.Κωδ\_βιβλιοθήκης

GROUP BY books\_and\_borrows.ISBN, books\_and\_borrows.Αριθμός\_αντιτύπου, books\_and\_borrows.Κωδικός\_Βιβλιοθήκης

) as all\_books LEFT OUTER JOIN

Δανεισμός

ON all\_books.ISBN=Δανεισμός.ISBN AND all\_books.Αριθμός\_αντιτύπου=Δανεισμός.Αρ\_αντιτύπου AND all\_books.Κωδικός\_Βιβλιοθήκης=Δανεισμός.Κωδικός\_βιβλιοθήκης\_αντιτύπου AND all\_books.Κωδικός\_δανεισμού=Δανεισμός.Κωδικός\_δανεισμού

LEFT OUTER JOIN

(SELECT ISBN, Κωδ\_βιβλιοθήκης, Αρ\_αντιτύπου, Ημερομηνία as date\_moved, Κατάσταση\_παραλαβής, Κωδ\_μεταφοράς, Κωδικός\_Βιβλιοθήκης\_προορισμού

FROM Μεταφορά,Περιέχει

WHERE Κωδικός\_μεταφοράς=Κωδ\_μεταφοράς) AS moved

ON all\_books.ISBN=moved.ISBN AND all\_books.Αριθμός\_αντιτύπου=moved.Αρ\_αντιτύπου AND all\_books.Κωδικός\_Βιβλιοθήκης=moved.Κωδ\_Βιβλιοθήκης AND all\_books.Κωδικός\_μεταφοράς=moved.Κωδ\_μεταφοράς

Book counts

SELECT all\_before\_kratiseis.ISBN, all\_before\_kratiseis.Βιβλιοθήκη\_τώρα, Όνομα, Ποσότητα - IFNULL(Κρατήσεις,0) AS Ποσότητα, IFNULL(Κρατήσεις,0) as Κρατήσεις

FROM(

SELECT new\_t.ISBN, new\_t.Βιβλιοθήκη\_τώρα, Όνομα, COUNT(\*) as Ποσότητα

FROM

(SELECT all\_books.ISBN, all\_books.Αριθμός\_αντιτύπου, all\_books.Κωδικός\_Βιβλιοθήκης,

CASE

WHEN (Δανεισμός.Κωδικός\_δανεισμού is NULL and Κωδικός\_μεταφοράς is NULL) THEN Κωδικός\_Βιβλιοθήκης

WHEN (Δανεισμός.Κωδικός\_δανεισμού is NOT NULL and Κωδικός\_μεταφοράς is NULL) THEN Βιβλιοθήκη\_καταχώρησης\_επιστροφής

-- CASE

-- WHEN (Βιβλιοθήκη\_καταχώρησης\_επιστροφής IS NULL) THEN NULL

-- WHEN (Βιβλιοθήκη\_καταχώρησης\_επιστροφής IS NOT NULL) THEN Βιβλιοθήκη\_καταχώρησης\_επιστροφής

-- END

WHEN (Δανεισμός.Κωδικός\_δανεισμού is NULL and Κωδικός\_μεταφοράς is NOT NULL) THEN

CASE

WHEN (Κατάσταση\_παραλαβής=0) THEN NULL

WHEN (Κατάσταση\_παραλαβής=1) THEN Κωδικός\_Βιβλιοθήκης\_προορισμού

END

WHEN (Δανεισμός.Κωδικός\_δανεισμού is NOT NULL and Κωδικός\_μεταφοράς is NOT NULL) THEN

CASE

WHEN (Ημερομηνία\_που\_επιστράφηκε>=date\_moved) THEN Βιβλιοθήκη\_καταχώρησης\_επιστροφής

WHEN (Ημερομηνία\_που\_επιστράφηκε<date\_moved) THEN Κωδικός\_Βιβλιοθήκης\_προορισμού

END

END AS Βιβλιοθήκη\_τώρα,

IF (Δανεισμός.Κωδικός\_δανεισμού is NULL and Κωδικός\_μεταφοράς is NOT NULL AND Κατάσταση\_παραλαβής=0,Κωδικός\_Βιβλιοθήκης\_προορισμού, NULL) AS Μεταφέρεται\_σε,

IF (Δανεισμός.Κωδικός\_δανεισμού is NOT NULL and Κωδικός\_μεταφοράς is NULL AND Βιβλιοθήκη\_καταχώρησης\_επιστροφής is NULL, Δανεισμός.Κωδικός\_μέλους, NULL) AS Δανεισμένο\_σε

FROM

( -- Πρώτα βρίσκω τα τελευταία απο τελευταιους δανεισμούς και μεταφορές και μετά βρίσκω λεπτομέρειες

SELECT books\_and\_borrows.ISBN, books\_and\_borrows.Αριθμός\_αντιτύπου, books\_and\_borrows.Κωδικός\_Βιβλιοθήκης, max(Κωδικός\_δανεισμού) as Κωδικός\_δανεισμού, max(Κωδικός\_μεταφοράς) as Κωδικός\_μεταφοράς

FROM (

SELECT Αντίτυπο.ISBN, Αντίτυπο.Αριθμός\_αντιτύπου, Αντίτυπο.Κωδικός\_Βιβλιοθήκης, Κωδικός\_δανεισμού, Ημερομηνία\_που\_επιστράφηκε

FROM Αντίτυπο LEFT JOIN Δανεισμός ON Αντίτυπο.ISBN=Δανεισμός.ISBN AND Αντίτυπο.Αριθμός\_αντιτύπου=Δανεισμός.Αρ\_αντιτύπου AND Αντίτυπο.Κωδικός\_Βιβλιοθήκης=Δανεισμός.Κωδικός\_βιβλιοθήκης\_αντιτύπου) as books\_and\_borrows

LEFT JOIN (

SELECT ISBN, Κωδ\_βιβλιοθήκης, Αρ\_αντιτύπου, Κωδικός\_μεταφοράς

FROM Μεταφορά,Περιέχει

WHERE Κωδικός\_μεταφοράς=Κωδ\_μεταφοράς) AS moved

ON books\_and\_borrows.ISBN=moved.ISBN and books\_and\_borrows.Αριθμός\_αντιτύπου=moved.Αρ\_αντιτύπου AND books\_and\_borrows.Κωδικός\_Βιβλιοθήκης=moved.Κωδ\_βιβλιοθήκης

GROUP BY books\_and\_borrows.ISBN, books\_and\_borrows.Αριθμός\_αντιτύπου, books\_and\_borrows.Κωδικός\_Βιβλιοθήκης

) as all\_books LEFT OUTER JOIN

Δανεισμός

ON all\_books.ISBN=Δανεισμός.ISBN AND all\_books.Αριθμός\_αντιτύπου=Δανεισμός.Αρ\_αντιτύπου AND all\_books.Κωδικός\_Βιβλιοθήκης=Δανεισμός.Κωδικός\_βιβλιοθήκης\_αντιτύπου AND all\_books.Κωδικός\_δανεισμού=Δανεισμός.Κωδικός\_δανεισμού

LEFT OUTER JOIN

(SELECT ISBN, Κωδ\_βιβλιοθήκης, Αρ\_αντιτύπου, Ημερομηνία as date\_moved, Κατάσταση\_παραλαβής, Κωδ\_μεταφοράς, Κωδικός\_Βιβλιοθήκης\_προορισμού

FROM Μεταφορά,Περιέχει

WHERE Κωδικός\_μεταφοράς=Κωδ\_μεταφοράς) AS moved

ON all\_books.ISBN=moved.ISBN AND all\_books.Αριθμός\_αντιτύπου=moved.Αρ\_αντιτύπου AND all\_books.Κωδικός\_Βιβλιοθήκης=moved.Κωδ\_Βιβλιοθήκης AND all\_books.Κωδικός\_μεταφοράς=moved.Κωδ\_μεταφοράς

) as new\_t JOIN Βιβλιοθήκη ON new\_t.Βιβλιοθήκη\_τώρα=Βιβλιοθήκη.Κωδικός\_Βιβλιοθήκης

WHERE new\_t.Βιβλιοθήκη\_τώρα IS NOT NULL

GROUP BY new\_t.ISBN, new\_t.Βιβλιοθήκη\_τώρα

) as all\_before\_kratiseis LEFT OUTER JOIN

(SELECT ISBN, Βιβλιοθήκη\_κράτησης, COUNT(\*) as Κρατήσεις

FROM Κράτηση

WHERE hour(TIMEDIFF(CURRENT\_TIMESTAMP(),Ημερομηνία\_κράτησης))/24 < 7 AND Κατάσταση\_ολοκλήρωσης=0

GROUP BY ISBN, Βιβλιοθήκη\_κράτησης) as krathseis

ON all\_before\_kratiseis.ISBN=krathseis.ISBN AND all\_before\_kratiseis.Βιβλιοθήκη\_τώρα=krathseis.Βιβλιοθήκη\_κράτησης

**Συνδεση μελους**

SELECT Κωδικός\_μέλους, email, Αρ\_Τηλ, Κωδικός\_πρόσβασης

FROM Μέλος JOIN Τηλ\_Μέλους ON Κωδικός\_μέλους=ID\_μέλους

WHERE email='' OR Αρ\_Τηλ='' – OR Κωδικός\_μέλους=''

Μπορει να δανειστει

SELECT (Όριο\_δανεισμών - borrows ) > 0 as possible

FROM (

SELECT Όριο\_δανεισμών

FROM Μέλος, Συνδρομή, Επιλογές\_Συνδρομής

WHERE Κωδικός\_μέλους=934632 AND Κωδ\_μέλους=934632 AND Κωδ\_συνδρομής=Κωδικός\_συνδρομής

ORDER BY Αριθμός\_συνδρομής

LIMIT 1) as max\_possible,

(SELECT COUNT(\*) as borrows

FROM Μέλος NATURAL JOIN Δανεισμός

WHERE Ημερομηνία\_που\_επιστράφηκε is NULL AND Κωδικός\_μέλους=934632) as made

Κοστος ασυνεπειας

SELECT IF (days>Διάρκεια\_δανεισμού, ceil((days-Διάρκεια\_δανεισμού) \* Επιβάρυνση\_καθυστέρησης\_ασυνέπειας), 0) AS cost

FROM

(SELECT Διάρκεια\_δανεισμού, Επιβάρυνση\_καθυστέρησης\_ασυνέπειας

FROM Μέλος, Συνδρομή, Επιλογές\_Συνδρομής

WHERE Κωδικός\_μέλους=934632 AND Κωδ\_μέλους=934632 AND Κωδ\_συνδρομής=Κωδικός\_συνδρομής

ORDER BY Αριθμός\_συνδρομής

LIMIT 1) as t1,

(SELECT datediff(CURRENT\_TIMESTAMP(),Ημερομηνία\_δανεισμού) as days, Ημερομηνία\_δανεισμού

FROM `Δανεισμός`

WHERE `Κωδικός\_μέλους`=934632 AND `ISBN`=9780060749910 AND `Αρ\_αντιτύπου`=835676770 AND`Ημερομηνία\_που\_επιστράφηκε` IS NULL) as t2