

Συστήματα Διαχείρισης και Ανάλυσης Δεδομένων
Διδάσκων: Ιωάννης Κωτίδης

Εαρινό εξάμηνο 2021-2022

Πρώτη Σειρά Ασκήσεων

Ανάθεση: 05-05-2022

Παράδοση: 17-05-2022 Ώρα (23:55)

Οδηγίες

- *Η εργασία είναι ατομική και υποχρεωτική.*
- *Η υποβολή της εργασίας πρέπει να γίνει στο eclass.*
- *Το παραδοτέο σας θα πρέπει να είναι ένα αρχείο PDF με όνομα AM.pdf (όπου AM είναι ο αριθμός μητρώου σας. π.χ. "3190001.pdf").*
- *Πιθανή αντιγραφή θα τιμωρείται με μηδενισμό όλων των εμπλεκομένων.*

"StackOverflow Database"

Στόχος της εργασίας είναι η πρακτική εφαρμογή των γνώσεων που αποκομίσατε από τις διαλέξεις του μαθήματος σχετικά με την δημιουργία ευρετηρίων και την βελτιστοποίηση των ερωτήσεων SQL. Για τον σκοπό της εργασίας θα χρησιμοποιήσετε την βάση δεδομένων του ιστότοπου StackOverFlow, η οποία περιέχει ερωτήσεις και απαντήσεις για ένα ευρύ φάσμα θεμάτων σχετικά με τον προγραμματισμό υπολογιστών. Πρόκειται ουσιαστικά για μια απλοποιημένη έκδοση η οποία υλοποιεί μέρος του λογικού σχήματος της πραγματικής βάσης του ιστότοπου και περιέχει δοκιμαστικά δεδομένα για τα έτη 2008 έως και 2010. Για περισσότερες πληροφορίες σχετικά με τη βάση δεδομένων StackOverflow ακολουθείστε τους παρακάτω συνδέσμους:

https://en.wikipedia.org/wiki/Stack_Overflow

<https://stackoverflow.com/>

Αρχικά θα δημιουργήσετε την βάση δεδομένων και θα φορτώσετε τα δεδομένα στους πίνακες, ακολουθώντας τις παρακάτω οδηγίες. Στη συνέχεια θα απαντήσετε στα ζητούμενα της εργασίας.

1. Οδηγίες για την δημιουργία της βάσης δεδομένων StackOverFlow.

Για να δημιουργήσετε την βάση δεδομένων και να φορτώσετε τις εγγραφές ακολουθείστε **ΠΡΟΣΕΚΤΙΚΑ** τα παρακάτω βήματα:

Βήμα 1: Κατεβάστε το αρχείο **sodata.zip** (μέγεθος αρχείου 1.12 GB) από τον παρακάτω σύνδεσμο:

<http://pages.aueb.gr/users/mkap/sodata.zip>

Βήμα 2: Αποσυμπίεστε το αρχείο **sodata.zip** στον φάκελο **C:\sodata** (μέγεθος φακέλου 5GB).

Βήμα 3: Από το περιβάλλον του Microsoft Sql Server Management Studio εκτελέστε το SQL script "**CreateStackOverflow.sql**" που δημιουργεί το λογικό σχήμα της βάσης.

Βήμα 4: Εκτελέστε το SQL script "**LoadSOData.sql**" το οποίο θα φορτώσει δεδομένα στους πίνακες της βάσης. Το συγκεκριμένο script περιέχει εντολές της μορφής:

```
BULK INSERT Users          ! Πίνακας στον οποίο θα φορτωθούν τα δεδομένα
FROM 'C:\sodata\Users.txt' ! Αρχείο το οποίο περιέχει τα δεδομένα.
WITH (DATAFILETYPE = 'widechar', FIRSTROW =2, FIELDTERMINATOR='|', ROWTERMINATOR = '\n');
```

Παράμετροι:

DATAFILETYPE = 'widechar' : Το αρχείο περιέχει δεδομένα σε μορφή unicode.

FIRSTROW=2 : Η πρώτη γραμμή του αρχείου περιέχει τα ονόματα των πεδίων και αγνοείται.

FIELDTERMINATOR = '|' : Ο χαρακτήρας '|' δηλώνει το τέλος κάθε πεδίου της εγγραφής.

ROWTERMINATOR='\n' : Ο χαρακτήρας αλλαγής γραμμής δηλώνει το τέλος κάθε εγγραφής του αρχείου.

ΠΡΟΣΟΧΗ: Αν τοποθετήσετε τα δεδομένα σε φάκελο διαφορετικό από τον '**C:\sodata**' θα πρέπει να τροποποιήσετε ανάλογα το path. Για παράδειγμα αν τοποθετήσετε τα δεδομένα στον φάκελο '**C:\DATA**' η παραπάνω εντολή πρέπει να αλλάξει ως εξής:

```
BULK INSERT Users
FROM 'C:\DATA\Users.txt'
WITH (DATAFILETYPE = 'widechar', FIRSTROW =2, FIELDTERMINATOR='|', ROWTERMINATOR = '\n');
```

Σημείωση: Για την διαδικασία της μαζικής εισαγωγής των δεδομένων απαιτούνται περίπου 7 λεπτά σε ένα υπολογιστή με δίσκο SSD και περίπου 11 λεπτά σε έναν υπολογιστή με συμβατικό σκληρό δίσκο. Το μέγεθος της βάσης είναι περίπου 10 GB (data & log files).

2. Περιγραφή των πινάκων της βάσης

Ακολουθεί η περιγραφή των πινάκων και των δεδομένων της βάσης.

PostTypes: Πίνακας με τους τύπους των αναρτήσεων. Αριθμός Εγγραφών=8	
postTypeId	Κωδικός
PostTypeName	Τύπος Ανάρτησης (π.χ. Question, Answer, Wiki κ.λπ.).

VoteTypes: Πίνακας με τα είδη των ψήφων. Αριθμός Εγγραφών=8	
VoteTypeId	Κωδικός
VoteTypeName	Είδος (π.χ. UpVote, DownVote κ.λπ.).

Users: Πίνακας με τα στοιχεία των χρηστών. Αριθμός εγγραφών=299397	
UserId	Κωδικός Χρήστη
AboutMe	Πληροφορίες που καταχωρεί ο χρήστης στην ενότητα "AboutMe" του προφίλ του.
CreationDate	Ημερομηνία δημιουργίας του χρήστη.
displayName	Όνομα Χρήστη.
DownVotes	Αριθμός αρνητικών ψήφων που έχει δώσει ο χρήστης σε αναρτήσεις άλλων χρηστών.
LastAccessDate	Ημερομηνία της πιο πρόσφατης επίσκεψης του χρήστη στον ιστότοπο.
UserLocation	Τοποθεσία Χρήστη.
Reputation	Φήμη. Μέτρηση η οποία εκφράζει τον βαθμό εμπιστοσύνης των μελών της κοινότητας στον συγκεκριμένο χρήστη. Αντικατοπτρίζει, ως ένα βαθμό, την εξοικείωση του χρήστη με τον ιστότοπο, την εξειδίκευσή του πάνω σε συγκεκριμένα ζητήματα και γενικότερα την συνεισφορά του στην κοινότητα.
UpVotes	Αριθμός θετικών ψήφων που έχει δώσει ο χρήστης σε αναρτήσεις άλλων χρηστών.
ProfileViews	Αριθμός προβολών του προφίλ του χρήστη
WebsiteUrl	Διεύθυνση Ιστοσελίδας του Χρήστη

Badges: Πίνακας με τα σήματα (διακριτικά) των χρηστών (π.χ. Guru, Teacher, Popular Question, Nice Answer κ.λπ.). Τα σήματα είναι "ψήγματα" ψηφιακής αίσθησης που κερδίζουν οι χρήστες για σχεδόν κάθε είδος δραστηριότητας στον ιστότοπο stackOverflow. Αριθμός Εγγραφών=1102017	
Bid	Μοναδικός Κωδικός Εγγραφής
Bname	Ονομασία σήματος
UserId	Κωδικός χρήστη
Bdate	Ημερομηνία απόδοσης του διακριτικού

Posts: Πίνακας με τις αναρτήσεις των χρηστών. Αριθμός εγγραφών=3729155	
PostId	Κωδικός ανάρτησης
AcceptedAnswerID	Κωδικός αποδεκτής απάντησης. Το πεδίο έχει τιμή μόνο για αναρτήσεις οι οποίες είναι ερωτήσεις (Question). Διαφορετικά έχει την τιμή NULL.
AnswerCount	Συνολικός αριθμός απαντήσεων για την συγκεκριμένη ανάρτηση εφόσον είναι ερώτηση. Για αναρτήσεις άλλου τύπου το πεδίο έχει την τιμή 0 (μηδέν).
Body	Κείμενο ανάρτησης
CommentCount	Δηλώνει τον αριθμό των σχολίων (comments) των χρηστών για την συγκεκριμένη ανάρτηση.

CreationDate	Ημερομηνία δημιουργίας της ανάρτησης.
LastActivityDate	Δηλώνει την πιο πρόσφατη ημερομηνία κατά την οποία έλαβε χώρα μια δραστηριότητα σχετική με την ανάρτηση (π.χ. δημιουργία, τροποποίηση κ.λπ.).
FavoriteCount	Αριθμός που δείχνει πόσο δημοφιλής είναι η ανάρτηση.
LastEditDate	Ημερομηνία πιο πρόσφατης τροποποίησης της ανάρτησης.
LastEditorDisplayName	Όνομα χρήστη που έκανε την πιο πρόσφατη τροποποίηση.
OwnerUserId	Ο κωδικός του Χρήστη που έκανε την ανάρτηση.
ParentId	Το πεδίο έχει τιμή μόνο για αναρτήσεις οι οποίες αποτελούν απάντηση κάποιας ερώτησης (Answer). Περιέχει τον κωδικό της ερώτησης στην οποία αναφέρεται η απάντηση και ουσιαστικά συνδέει κάθε ερώτηση με τις απαντήσεις της.
PostTypeId	Κωδικός που δηλώνει τον τύπο της ανάρτησης.
Score	Το Score της ανάρτησης το οποίο υπολογίζεται ως η διαφορά ανάμεσα στις θετικές και τις αρνητικές ψήφους που έχει λάβει η ανάρτηση.
Title	Ο τίτλος της ανάρτησης.
ViewCount	Ο αριθμός επισκέψεων/εμφανίσεων της ανάρτησης.

Comments: Πίνακας με τα σχόλια των χρηστών. Αριθμός Εγγραφών=3874841	
Cid	Μοναδικός κωδικός εγγραφής.
CreationDate	Ημερομηνία δημιουργίας σχολίου.
PostId	Ο κωδικός της ανάρτησης στην οποία αναφέρεται το σχόλιο.
Score	Το σύνολο των θετικών ψήφων που έχει λάβει το σχόλιο.
Text	Το κείμενο του σχολίου.
UserId	Ο κωδικός του χρήστη που δημιούργησε το σχόλιο.

postLinks: Πίνακας για την διασύνδεση σχετικών αναρτήσεων. Αριθμός Εγγραφών=135391	
id	Μοναδικός κωδικός εγγραφής.
CreationDate	Ημερομηνία δημιουργίας της σύνδεσης.
PostId	Κωδικός ανάρτησης.
RelatedPostId	Κωδικός σχετικής ανάρτησης.

Tags: Πίνακας με ετικέτες (λέξεις κλειδιά) που αποδίδουν οι χρήστες στις αναρτήσεις τους. Αριθμός Εγγραφών=24218.	
Tagid	Κωδικός ετικέτας.
tagName	Ετικέτα.

PostTags: Πίνακας που συσχετίζει τις ετικέτες με τις αναρτήσεις. Αριθμός Εγγραφών=3136519.	
PostId	Κωδικός ανάρτησης.
Tagid	Κωδικός ετικέτας.

Votes: Πίνακας με τις ψήφους των χρηστών. Αριθμός Εγγραφών=8079643	
Voteld	Μοναδικός κωδικός εγγραφής.
PostId	Κωδικός ανάρτησης στην οποία δίνεται η ψήφος.
UserId	Ο κωδικός του χρήστη που έδωσε την ψήφο.
BountyAmount	Μονάδες πριμοδότησης.
VoteTypeId	Κωδικός που δηλώνει το είδος της ψήφου.
CreationDate	Ημερομηνία απόδοσης της ψήφου.

3. Ζητούμενα εργασίας

Ακολουθούν τα ζητούμενα της εργασίας. Για την απάντηση των ζητημάτων **δεν επιτρέπεται καμία απολύτως τροποποίηση του σχήματος** εκτός φυσικά από την δημιουργία των ζητούμενων ευρετηρίων. Επίσης **απαγορεύεται** η δημιουργία και η χρήση όψεων (views). Αν για κάποιο ζήτημα **δεν ισχύουν** οι συγκεκριμένοι περιορισμοί τότε από θα δηλώνεται ρητά στην εκφώνηση του ζητήματος.

Σε κάθε ζήτημα δεν αρκεί μόνο να παραθέσετε τα ερωτήματα σε γλώσσα SQL ή/και τις εντολές δημιουργίας των ευρετηρίων που ζητούνται. Σε κάθε περίπτωση **πρέπει να τεκμηριώσετε τις απαντήσεις σας και να παραθέσετε στοιχεία που επιβεβαιώνουν τους ισχυρισμούς σας**. Για παράδειγμα:

- Σε περιπτώσεις που ζητείται να αποδείξετε ότι ένα ευρετήριο επιταχύνει ένα ερώτημα, εκτελέστε το ερωτήμα δίχως το ευρετήριο και εξετάστε το πλάνο εκτέλεσης. Αφού δημιουργήσετε το ευρετήριο εκτελέστε εκ νέου το ερωτήμα και επανεξετάστε το πλάνο εκτέλεσης. Συγκρίνοντας τα δύο πλάνα μπορείτε να καταλήξετε σε συμπεράσματα σχετικά με την καταλληλότητα του ευρετηρίου.
- Σε περιπτώσεις που πρέπει να συγκρίνετε ένα ή περισσότερα ερωτήματα, εκτελέστε τα όλα μαζί σε δέσμη και εξετάστε τα πλάνα εκτέλεσης. Ο SQL server δείχνει το κόστος κάθε ερωτήματος ως ποσοστό επί του συνολικού κόστους εκτέλεσης της δέσμης.
- Ενεργοποιείτε τα στατιστικά στοιχεία I/O με την εντολή: **set statistics io on**. Με τον τρόπο αυτό μπορείτε να βλέπετε κάθε φορά που εκτελείτε ένα ερωτήμα πόσες σελίδες διαβάζονται από τον δίσκο ή/και από την μνήμη (buffer).

- Μπορείτε να ενεργοποιήσετε τα στατιστικά στοιχεία σχετικά με τον χρόνο εκτέλεσης του επερωτήματος με την εντολή **set statistics time on**.
- Κάθε φορά πριν την εκτέλεση ενός επερωτήματος, εκτελέστε τις παρακάτω εντολές που "καθαρίζουν" τους buffers που χρησιμοποιεί ο SQL server για την αποθήκευση των δεδομένων και των πλάνων εκτέλεσης:

checkpoint
dbcc dropcleanbuffers

Με τον τρόπο αυτό διασφαλίζετε ότι, το επερώτημα που θα εκτελέσετε δεν θα χρησιμοποιήσει τυχόν σελίδες που υπάρχουν στην μνήμη από προηγούμενες εκτελέσεις του ιδίου ή/και άλλων επερωτημάτων. Σε αντίθετη περίπτωση μπορεί να οδηγηθείτε σε λάθος συμπεράσματα.

ΠΡΟΣΟΧΗ: Κάθε ζήτημα πρέπει να το αντιμετωπίσετε ανεξάρτητα από τα υπόλοιπα και να το υλοποιήσετε στο αρχικό στιγμιότυπο της βάσης. Για παράδειγμα αν θέλετε να εξετάσετε κατά πόσο ένα ευρετήριο κάνει πιο αποδοτικό ένα ερώτημα, βεβαιωθείτε ότι έχετε διαγράψει (drop index) τα ευρετήρια που έχετε δημιουργήσει για την βελτιστοποίηση άλλων επερωτημάτων.

Ζήτημα Πρώτο [20 μονάδες]

Καλείστε να βελτιστοποιήσετε την εκτέλεση ενός επερωτήματος το οποίο θα εμφανίζει τα ονόματα των χρηστών που ζουν σε συγκεκριμένη περιοχή και δημιούργησαν το προφίλ τους σε μια συγκεκριμένη ημερομηνία. Για παράδειγμα το επερώτημα **E1** που ακολουθεί εμφανίζει τα ονόματα των χρηστών που ζουν στην Αθήνα και δημιούργησαν το προφίλ τους στον ιστότοπο stackOverflow στις 15/09/2008.

```
E1
select DisplayName
  From users
 where CreationDate = '2008-09-15' and
        Userlocation='Athens, Greece'
```

1. Βάσει των γνώσεων που αποκομίσατε από τις διαλέξεις του μαθήματος (δίχως να συμβουλευτείτε τον query optimizer) ποιο από τα παρακάτω δύο ευρετήρια θα επιλέγατε να δημιουργήσετε; **Να αιτιολογήσετε την απάντησή σας.**

- α) **create index** idx_cdate **ON** users(creationDate)
- β) **create index** idx_location **ON** users(UserLocation)

2. Δημιουργήστε το ευρετήριο που επιλέξατε και **παραθέστε στοιχεία που να αποδεικνύουν** ότι επιταχύνει την εκτέλεση του επερωτήματος E1.
3. Προτείνετε ένα ευρετήριο το οποίο επιταχύνει ακόμη περισσότερο την εκτέλεση του επερωτήματος **E1** από ότι το ευρετήριο που επιλέξατε στο ερώτημα 2. **Να παραθέστε στοιχεία που να τεκμηριώνουν την πρότασή σας.**

Ζήτημα Δεύτερο [20 μονάδες]

Οι χρήστες του ιστότοπου αναζητούν συχνά τα στοιχεία των αναρτήσεων βάσει των ετικετών που αποδίδουν οι χρήστες. Για παράδειγμα το παρακάτω επερώτημα **E2** εμφανίζει τον τίτλο, το περιεχόμενο, το σκορ και τον αριθμό των εμφανίσεων όλων των αναρτήσεων στις οποίες οι χρήστες έχουν αποδώσει την ετικέτα (λέξη κλειδί) 'Informix' (πληροφοριακά το Informix είναι DBMS της IBM).

E2

```
select title, body, score, ViewCount
  from Posts, PostTags, Tags
 where Posts.postid=PostTags.postid and
        PostTags.tagid=tags.tagid and tagname = 'Informix'
```

1. Καλείστε να δημιουργήσετε **ένα ή περισσότερα** κατάλληλα ευρετήρια τα οποία θα βελτιστοποιούν την εκτέλεση του **E2**. Δεδομένου ότι καθημερινά καταχωρείται στην βάση ένας μεγάλος αριθμός σχολίων, πρέπει να δημιουργήσετε όσο το **δυνατόν λιγότερα** ευρετήρια ώστε να επιβαρύνονται στο ελάχιστο λειτουργίες όπως η καταχώρηση και η ενημέρωση των αναρτήσεων.
2. Εφόσον ολοκληρώσατε την απάντησή σας στο παραπάνω ερώτημα εκτελέστε το ακόλουθο επερώτημα:

E3

```
select title, body, score, ViewCount
  from Posts, PostTags, Tags
 where Posts.postid=PostTags.postid and
        PostTags.tagid=tags.tagid and tagname = 'SQL'
```

Συγκρίνετε την εκτέλεση των δύο επερωτημάτων. Τι παρατηρείτε; Περιγράψτε με συντομία.

ΣΗΜΕΙΩΣΗ: Για την απάντηση του συγκεκριμένου ζητήματος επιτρέπεται η τροποποίηση του λογικού σχήματος.

Ζήτημα 3 [20 μονάδες]

Θεωρήστε το παρακάτω ερώτημα σε φυσική γλώσσα:

"Εμφανίστε τον κωδικό (userid) και το όνομα (displayName) των χρηστών των οποίων όλες οι αναρτήσεις έχουν αρνητικό σκορ (score)."

Να γράψετε τουλάχιστον δύο διαφορετικά επερωτήματα σε γλώσσα SQL που να απαντούν στο παραπάνω ερώτημα και να δημιουργήσετε κατάλληλα ευρετήρια που επιταχύνουν την εκτέλεσή τους. Ποιό επερώτημα θα επιλέγατε και γιατί;

Ζήτημα Τέταρτο [40 μονάδες]

1. Να διατυπώσετε δύο ερωτήματα σε φυσική γλώσσα και στη συνέχεια να γράψετε εντολές σε γλώσσα SQL ώστε να απαντηθούν τα ερωτήματα που διατυπώσατε.
2. Να δημιουργήσετε κατάλληλα ευρετήρια που επιταχύνουν την εκτέλεση των επερωτημάτων σας. Να παραθέσετε τις εντολές δημιουργίας των ευρετηρίων, καθώς επίσης και στοιχεία που να αποδεικνύουν ότι τα ευρετήρια που δημιουργήσατε επιταχύνουν την εκτέλεση των επερωτημάτων.

Φροντίστε τα επερωτήματα που θα γράψετε να δίνουν χρήσιμες πληροφορίες, να μην είναι εντελώς απλοϊκά και να μην χρησιμοποιούν μόνο ευρετήρια που δημιουργήσατε για να απαντήσετε τα προηγούμενα ζητήματα.

Ζήτημα Πέμπτο [10 μονάδες]

Ζητήθηκε από ένα προγραμματιστή του ιστότοπου stackOverflow να γράψει ένα επερώτημα το οποίο να εμφανίζει τα ονόματα των χρηστών μιας συγκεκριμένης χώρας (Π.χ. Greece).

Ο προγραμματιστής παρατήρησε ότι δεν υπάρχει ξεχωριστό πεδίο για την χώρα και ότι στην συντριπτική πλειοψηφία των εγγραφών το όνομα της χώρας βρίσκεται στο τέλος του πεδίου **userLocation** όπως φαίνεται στον παρακάτω πίνακα:

UserLocation
Athens, Greece
Heraklion, Greece
Volos, Greece
Larisa, Larissa, Greece

Ο προγραμματιστής έγραψε το παρακάτω επερώτημα:

E4

```
select displayName, userlocation
  from users where userlocation like '%Greece'
```

και στη συνέχεια για να επιταχύνει την εκτέλεσή του ερωτήματος δημιούργησε το ακόλουθο ευρετήριο:

```
create index idx1 on users (userLocation) include (displayName)
```

1. Το ευρετήριο **idx1** επιταχύνει την εκτέλεση του παραπάνω επερωτήματος; Να αιτιολογήσετε την απάντησή σας.
2. Ζητείται να υλοποιήσετε κατάλληλη λύση για την βελτιστοποίηση του επερωτήματος **E4**. Να παραθέσετε στοιχεία που αποδεικνύουν την αποτελεσματικότητα της προτεινόμενης λύσης.

ΣΗΜΕΙΩΣΗ: Για την απάντηση του συγκεκριμένου ζητήματος επιτρέπεται η τροποποίηση του λογικού σχήματος.