



AppStore μελέτη περίπτωσης

Το AppStore είναι ένα τυπικό ηλεκτρονικό κατάστημα για εφαρμογές λογισμικού κινητών συσκευών (Apps). Σ' αυτό διακινούνται Apps που ανήκουν που συνήθως κατατάσσονται σε μια από τις διαθέσιμες κατηγορίες στις οποίες εξειδικεύεται το AppStore. Ειδικότερα, οι διαθέσιμες κατηγορίες διακρίνονται από ένα ενδεικτικό κωδικό (Cat_Name), μια περιγραφή της κατηγορίας (Cat_Description) και την ένδειξη της ομάδας του πληθυσμού για την οποία ενδείκνυται η κατηγορία (Cat_Age_Group). Τα Apps δηλώνονται στο AppStore με ένα μοναδικό αλφαριθμητικό αναγνωριστικό (App_ID), το όνομα του App (App_Name), μια περιγραφή του (App_Description), την ημερομηνία κατασκευής του (App_Date), την τρέχουσα έκδοση του (App_Version) και το πλήθος των εγκαταστάσεων του App (App_Installations) που έχουν επιτευχθεί μέσω του ηλεκτρονικού καταστήματος. Όταν ένα App καταγράφεται στο ηλεκτρονικό κατάστημα δηλώνεται ότι ανήκει / κατατάσσεται σε μία μόνο κατηγορία από αυτές που έχουν οριστεί στο ηλεκτρονικό κατάστημα.

Στο AppStore καταγράφονται επίσης:

- δημοφιλή λειτουργικά συστήματα / πλατφόρμες, που για καθένα υπάρχει ένα ενδεικτικό (μοναδικό) όνομα (OS_Name), ένδειξη μοντέλου (Source_Model) και μια ιστοσελίδα (Website)
- κατασκευαστές κινητών συσκευών (Vendor) που έχουν ένα μοναδικό αναγνωριστικό (Vendor_Name), ένδειξη γεωγραφικής θέσης (Vendor_Location) και αναφορά του ιδρυτή του (Vendor_Founder).
- φορείς (Developer) που αναπτύσσουν Apps για τους οποίους καταγράφονται ο μοναδικός κωδικός του (Dev_ID), το όνομα του (Dev_Name), το email επικοινωνίας του (Dev_email) και η διεύθυνσή του (Dev_Address).

Ας σημειωθεί ότι ένα app σχεδιάζεται και αναπτύσσεται από ένα φορέα ο οποίος έχει και τη συνολική ευθύνη του έναντι του ηλεκτρονικού καταστήματος. Τέλος, ένα λειτουργικό σύστημα / πλατφόρμα μπορεί να χρησιμοποιείται από πολλά Apps και κατασκευαστές κινητών συσκευών ενώ ταυτόχρονα ένα App αλλά και ένας κατασκευαστής κινητών συσκευών να υποστηρίζουν πολλά λειτουργικά συστήματα / πλατφόρμες.

Άσκηση 1 (Αναγνώριση συναρτησιακών εξαρτήσεων και σχεσιακό σχήμα)

Ερώτημα 1Α (Συναρτησιακές εξαρτήσεις και σχεσιακό σχήμα): Η πρώτη άσκηση αποσκοπεί στην εξοικείωση των φοιτητών με το σχεδιασμό σχεσιακών σχημάτων με τη χρήση συναρτησιακών εξαρτήσεων. Συγκεκριμένα, σας ζητούν να φιλτράρετε την εκφώνηση έτσι ώστε να εντοπίσετε (και να καταγράψετε) τις συναρτησιακές εξαρτήσεις που εντοπίζετε. Στη συνέχεια, με βάση τις συναρτησιακές εξαρτήσεις, τα διαθέσιμα δεδομένα και τα scripts (βλ. υλικό eclass) θα πρέπει να οριστικοποιήσετε το σχεσιακό σχήμα.



Ερώτημα 1B (Τροποποιήσεις σχήματος): Σε συνέχεια της εκφώνησης της μελέτης περίπτωσης AppStore σας ενημερώνουν ότι πρέπει να δρομολογηθούν μια σειρά από τροποποιήσεις. Η πρώτη αφορά την υποστήριξη αξιολογήσεων (ratings) για τα Apps του ηλεκτρονικού καταστήματος. Τα δεδομένα των αξιολογήσεων που πρέπει να καταγράφονται έχουν ως εξής. Μια αξιολόγηση αφορά ένα και μόνο app και έχει ένα μοναδικό κωδικό (Rating_ID), μια τιμή (App_Rating_Value) σε μια κλίμακα (0 έως 5) και την ημερομηνία της αξιολόγησης (App_Rating_Date).

Η δεύτερη απαίτηση αφορά την καταγραφή των καταναλωτών που χρησιμοποιούν το AppStore για να αγοράσουν Apps. Στη τρέχουσα έκδοση του AppStore ως καταναλωτές θεωρούνται μόνο οι πιστοποιημένοι χρήστες - πελάτες των κατασκευαστών (Vendor) που υπάρχουν στη βάση. Για αυτούς, το AppStore πρέπει να καταγράφει ένα μοναδικό username, όνομα, επώνυμο, τηλέφωνο επικοινωνίας και διεύθυνση κατοικίας. Τα δεδομένα αυτά αντλούνται από υπηρεσίες που προσφέρουν οι κατασκευαστές (τύπου APIs) και οι οποίες λειτουργούν ως μια μορφή third-party log in. Οι επιμέρους λεπτομέρειες δεν πρέπει να σας απασχολήσουν πέρα του ότι η βάση σας πρέπει να καταγράφει αυτά που προ αναφέραμε ανεξαρτήτως της προέλευσης τους.

Με βάση τα παραπάνω θα πρέπει πρώτα να αναγνωρίσετε τις νέες συναρτησιακές εξαρτήσεις που προκύπτουν από τις νέες απαιτήσεις και στη συνέχεια θα πρέπει να ενημερώσετε τη βάση σας κατάλληλα. Έχουν αναρτηθεί ενδεικτικά δεδομένα με Ratings κάποιων Apps και ένα ερώτημα που μπορείτε να διατυπώσετε σε SQL προς επιβεβαίωση. Για χρήστες - καταναλωτές εισάγετε τουλάχιστον δύο πλειάδες της αρεσκείας σας.

Άσκηση 2 (Βελτιώσεις σχεσιακού σχήματος & νέοι τύποι δεδομένων)

Ερώτημα 2A (Νέοι τύποι δεδομένων): Στο script της βάσης δεδομένων AppStore (στο οποίο εργάζεστε) πρέπει να υλοποιήσετε τις ακόλουθες τροποποιήσεις:

- Δημιουργήστε ένα νέο τύπο δεδομένων S_Model_Type και χρησιμοποιήστε τον κατάλληλα για αποφυγή της δήλωσης CHECK στον πίνακα OPERATING_SYSTEM
- Δημιουργήστε ένα νέο τύπο δεδομένων 'Affiliation' και χρησιμοποιήστε τον κατάλληλα για διαχωρισμό των DEVELOPER σε εξωτερικούς (E) και εσωτερικούς (I)
- Δημιουργήστε ένα νέο σύνθετο τύπο δεδομένων Address με παραμέτρους Line VARCHAR(90) και Zip_Code VARCHAR(30) για τις διευθύνσεις των DEVELOPER
- Μετατρέψτε το email στον πίνακα DEVELOPER σε multidimensional array έτσι ώστε να είναι δυνατή η εισαγωγή δύο τύπων email (π.χ. ιδρυματικό και προσωπικό) με κατάλληλη κωδικοποίηση στο array
- Υποθέστε ότι κάθε App έχει ένα Blog στο οποίο αναρτώνται μηνύματα σχετικά με το App (από DEVELOPERs ή VENDORs του AppStore). Ακολουθώς εξετάστε τη χρήση ενός νέου σύνθετου τύπου δεδομένων MESSAGETYPE με παραμέτρους PERIEXOMENO VARCHAR(4000), HMEROMHΝΙΑ DATE και TAGS TEXT [] που να επιτρέπει αναρτήσεις μηνυμάτων σε Blogs των Apps του AppStore



- Δημιουργήστε νέο σύνθετο τύπο δεδομένων RATING με παραμέτρους Rating_ID INT, App_ID TEXT, App_Rating_Value TEXT, App_Rating_Date TEXT για την ανάρτηση των αξιολογήσεων που ήδη υπάρχουν στη βάση σας

Ερώτημα 2B (Querying): Αφού ολοκληρώσετε τις τροποποιήσεις θα πρέπει να απαντήσετε τα ακόλουθα ερωτήματα σε SQL (Σημ: Πρέπει να προσέξετε διότι ανάλογα με τις τροποποιήσεις που θα κάνετε κάποια από τα παρακάτω ερωτήματα ίσως πρέπει να αλλάξουν. Συνιστάται, με το που ολοκληρώνονται κάποιες τροποποιήσεις στο σχήμα, να διατυπώνονται τα αντίστοιχα ερωτήματα και μετά να συνεχίζετε):

1. Για όλους τους DEVELOPER τυπώστε όλα τα προσωπικά emails που δηλώνουν
2. Να εκτυπωθούν τα στοιχεία των DEVELOPER που δηλώνουν ZIP code που περιλαμβάνει τους χαρακτήρες 'CA' και έχουν affiliation 'E'
3. Να εκτυπωθούν τα στοιχεία των DEVELOPER που δεν δηλώνουν ZIP code
4. Να εκτυπωθούν οι συγγραφείς των μηνυμάτων και τα περιεχόμενα των μηνυμάτων που έχουν tag της αρεσκείας σας (π.χ., 'Messaging')
5. Να υπολογιστεί το πλήθος των μηνυμάτων που περιλαμβάνουν ένα tag της αρεσκείας σας (π.χ. 'Messaging')
6. Να εκτυπωθούν οι συγγραφείς των μηνυμάτων και τα περιεχόμενα των μηνυμάτων που περιλαμβάνουν ένα απόσπασμα της αρεσκείας σας (π.χ. 'Sales')
7. Εκτυπώστε το περιεχόμενο των μηνυμάτων που δεν περιλαμβάνουν ένα tag της επιλογής σας, τον συγγραφέα τους και το Blog
8. Να εκτυπωθούν τα στοιχεία των APP που έχουν τιμή αξιολόγησης 2 (το ερώτημα πρέπει να απαντηθεί στο τροποποιημένο script που το RATING είναι σύνθετος τύπος)
9. Να εκτυπωθούν τα στοιχεία των APP που δεν έχουν αξιολογηθεί (το ερώτημα πρέπει να απαντηθεί στο τροποποιημένο script που το RATING είναι σύνθετος τύπος)
10. Να εκτυπωθεί το APP_ID και η τιμή αξιολόγησης όλων των APP εκτός αυτών που δεν έχουν αξιολογηθεί (το ερώτημα πρέπει να απαντηθεί στο τροποποιημένο script που το RATING είναι σύνθετος τύπος)

Άσκηση 3 (JSON/XML και ιεραρχίες εξειδίκευσης)

Ακολούθως θα πρέπει να τροποποιήσετε το script του AppStore έτσι ώστε να ενσωματωθούν δεδομένα τύπου JSON και να δημιουργηθεί μια ιεραρχία εξειδίκευσης. Συγκεκριμένα θα πρέπει να γίνουν τα ακόλουθα:

Ερώτημα 3A (JSON data type): Τροποποιήστε το script έτσι ώστε ο πίνακας RATING να ορίζεται ως ακολούθως:

```
CREATE TABLE RATING (  
Rating_ID serial NOT NULL PRIMARY KEY,  
info JSON NOT NULL);
```



Αφού ενημερώσετε τον πίνακα με ενδεικτικά δεδομένα (από τα ήδη διαθέσιμα) έτσι ώστε να υπάρχει συμβατότητα με τον νέο ορισμό θα πρέπει να διατυπώσετε δηλώσεις SQL για τα ερωτήματα:

- Ποια είναι τα Apps που έχουν έστω και μία αξιολόγηση
- Για κάθε App που έχει δεχθεί αξιολόγηση βρείτε τον κωδικό αξιολόγησης, την ημερομηνία και την τιμή της αξιολόγησης
- Βρείτε τα Apps που έχουν αξιολογήσεις με τιμή > 3
- Για τα Apps που έχουν αξιολογήσεις με τιμή > 3 να βρείτε τα installations που έχουν
- Βρείτε τα λειτουργικά συστήματα που τρέχουν τα Apps που έχουν αξιολογήσεις > 3

Ερώτημα 3B (Κληρονομικότητα): Χρησιμοποιήστε το script που ακολουθεί (ως οδηγό) για να τροποποιήσετε την έκδοση του AppStotre που υλοποιήσατε προηγούμενα έτσι ώστε να υποστηρίζεται η συγκεκριμένη ιεραρχία εξειδίκευσης

```
CREATE TABLE CATEGORY (  
Name TEXT NOT NULL,  
PRIMARY KEY (Name));
```

```
CREATE TABLE HEALTH_AND_FITNESS (  
Description TEXT ) INHERITS (CATEGORY);
```

```
CREATE TABLE PRODUCTIVITY (  
Description TEXT ) INHERITS (CATEGORY);
```

```
CREATE TABLE TRAVEL_AND_LOCAL (  
Description TEXT ) INHERITS (CATEGORY);
```

```
CREATE TABLE COMMUNICATION (  
Description TEXT ) INHERITS (CATEGORY);
```

```
CREATE TABLE WEATHER (  
Description TEXT ) INHERITS (CATEGORY);
```

```
CREATE TABLE PHOTOGRAPHY (  
Description TEXT ) INHERITS (CATEGORY);
```

```
CREATE TABLE GAMES (  
Description TEXT,  
SUB_CATEGORIES TEXT [][[]] ) INHERITS (CATEGORY);
```



```
CREATE TABLE GAME_SIMULATION (  
Description TEXT,  
SUB_CATEGORIES TEXT [[]] ) INHERITS (CATEGORY);
```

```
CREATE TABLE GAME_ACTION (  
Description TEXT ) INHERITS (CATEGORY);
```

```
CREATE TABLE SOCIAL (  
Description TEXT ) INHERITS (CATEGORY);
```

```
CREATE TABLE ENTERTAINMENT (  
Description TEXT ) INHERITS (CATEGORY);
```

```
CREATE TABLE ART_AND_DESIGN (  
Description TEXT ) INHERITS (CATEGORY);
```

Όπου συναντάτε το γνώρισμα SUB_CATEGORIES που είναι τύπου array (απλό ή multi-dimensional) να θεωρήσετε ότι χρησιμοποιείται για να καταχωρηθούν χαρακτηριστικά του εκάστοτε App υπό μορφή ζευγών όπως π.χ. {"Field", "War affair"} ή / και {"Period", "Second World War"}. Αφού τροποποιήσετε κατάλληλα κάποια από τα δεδομένα σας θα πρέπει να διατυπώσετε τη δήλωση της SQL που υπολογίζει το App_ID, το όνομα του App, το όνομα του κατασκευαστή του App και τον αριθμό των εγκαταστάσεων του για όλα τα Apps μιας υποκατηγορίας (π.χ. GAME_SIMULATION) τα οποία σχετίζονται με κάποιο χαρακτηριστικό της αρεσκείας σας (π.χ. έχουν ως περίοδο αναφοράς το δεύτερο παγκόσμιο πόλεμο, κάτι που διατυπώνεται ως {"Period", "Second World War"}.

Άσκηση 4 (Γράφοι σε Postgres και ερωτήματα τροχιάς)

Η άσκηση 4 έχει ως διττό στόχο αφενός να εξοικειώσει τους φοιτητές με την ανάπτυξη γράφων και αφετέρου να επιτρέψει την ανάλυση εναλλακτικών τρόπων αναπαράστασης γράφων με τη χρήση της σχεσιακής τεχνολογίας. Προοδευτικά η ανάλυση αυτή θα επιτρέψει την κατανόηση και εξάσκηση ορισμένων προχωρημένων τεχνικών της PostgreSQL όπως η αναδρομική λειτουργία για τη διάσχιση (traversal) και το μεταβατικό κλείσιμο (transitive closure) γράφων. Οι φοιτητές εξακολουθούν και εργάζονται στην τρέχουσα υλοποίηση της βάσης δεδομένων AppStore όπου θα χρειαστεί να υλοποιηθούν τα ακόλουθα:

Ερώτημα 4A (Δημιουργία ενός property graph και υλοποίηση σε PostgreSQL): Θα ξεκινήσετε με την ανάπτυξη (καταρχήν στο χαρτί ή σε κάποιο εργαλείο) ενός property graph (βλέπε θεωρία) που θα αποτυπώνει τα ακόλουθα:



- Σχέσεις ακολουθίας (δηλ. follows) μεταξύ των φορέων (developers) που αναπτύσσουν Apps (δηλαδή ποιος ακολουθεί ποιόν, κατά την έννοια που συναντάται η ακολουθία στην υπηρεσία Twitter)
- Σχέσεις ενδιαφέροντος (δηλ. interest) των φορέων (developers) που αναπτύσσουν Apps για λειτουργικά συστήματα (σημειώνεται ότι η σχέση αυτή θα μπορούσε να αποτυπώνεται στην τρέχουσα έκδοση της υλοποίησής σας με τα tags στο περιεχόμενο των μηνυμάτων που αναρτούν οι developers σε κάποιο τοίχο).
- Σχέση ιδιοκτησίας (δηλ. belongs_to) για τους φορείς (developers) που αναρτούν μηνύματα σε κάποιο τοίχο.

Η τελική έκδοση του property graph που θα αναπτυχθεί θα πρέπει να αποτυπώνει οπωσδήποτε δεδομένα που υπάρχουν ήδη στη βάση σας και τυχόν νέα που θα θελήσετε να προσθέσετε

Έχοντας σχεδιάσει το ενδεικτικό property graph (με μικρό αριθμό κόμβων έτσι ώστε να καταγράφονται τουλάχιστον δύο παραστάσεις κάθε κόμβου) θα πρέπει στην συνέχεια να αναπαραστήσετε το property graph με τη χρήση σχεσιακής τεχνολογίας. Είναι στη διακριτική σας ευχέρεια να επιλέξετε τεχνική από αυτές που έχετε διδαχθεί μέχρι τώρα δηλαδή χρήση κανονικοποιημένων σχέσεων (normalized relation), μη-κανονικοποιημένων σχέσεων (unnormalized relation), λίστες γειτνίασης (adjacency lists), κοκ, ενώ θα πρέπει να μελετήσετε τον τρόπο που θα υποστηριχθούν οι εναλλακτικοί κόμβοι και ακμές που θα υπάρχουν στο γράφο σας (βλέπε μηχανισμό κληρονομικότητας της PostgreSQL).

Ερώτημα 4B (Ερωτήματα τροχιάς με αναδρομή): Έχοντας αναπαραστήσει το property graph με σχέσεις θα πρέπει να εξοικειωθείτε με σύγχρονες τεχνικές διάσχισης γράφων χρησιμοποιώντας αναδρομικές κλήσεις της SQL. Συγκεκριμένα, στην τρέχουσα έκδοση της βάσης δεδομένων AppStore που έχετε υλοποιήσει θα πρέπει να απαντήσετε στα παρακάτω ερωτήματα:

1. Για κάθε / ένα κόμβο υπολογίστε τους επόμενους (σε μια κατεύθυνση της επιλογής σας)
2. Για κάθε / ένα κόμβο υπολογίστε τους προηγούμενους (σε μια κατεύθυνση επιλογής σας)
3. Για κάθε / ένα κόμβο υπολογίστε τους επόμενους ανεξαρτήτως κατεύθυνσης
4. Για κάθε / ένα κόμβο υπολογίστε τους προηγούμενους ανεξαρτήτως κατεύθυνσης
5. Υπολογίστε όλες τις πιθανές διαδρομές που συνδέουν τους καταγεγραμμένους developers στο property graph που υλοποιήσατε

Άσκηση 5 (NoSQL)

Η τελευταία άσκηση απαιτεί να επιλέξετε ένα σύστημα NoSQL της προτίμησής σας και να επανεξετάσετε (ένα υποσύνολο) της βάσης δεδομένων AppStore έτσι ώστε να αποκτήσετε hands-on experience με ένα σύστημα διαφορετικού τύπου και φιλοσοφίας από αυτό που χρησιμοποιήσατε για να αναπτύξετε τη βάση δεδομένων. Σκοπός σας δεν πρέπει να είναι να δημιουργήσετε μια ακόμα διαφορετική υλοποίηση της βάσης δεδομένων σε ένα άλλο υπολογιστικό περιβάλλον (γι' αυτό και η αναφορά στο υποσύνολο παραπάνω) αλλά να



μεταφέρετε ένα τμήμα των δεδομένων σας σε NoSQL περιβάλλον έτσι ώστε να δημιουργηθεί μια νέα πηγή δεδομένων (data source) συμπληρωματικά προς την PostgreSQL. Το τμήμα των δεδομένων που θα επιλέξετε θα πρέπει να είναι ρεαλιστικό και να ανταποκρίνεται σε πραγματικές συνθήκες. Παραδείγματος χάριν θα μπορούσε να είναι ένα σύνολο από κριτικές και ratings για Apps από ένα ανοικτό σύνολο δεδομένων. Φοιτητές που θα καταφέρουν να παρουσιάσουν ένα ενδεικτικό σενάριο αξιοποίησης των δεδομένων από όλες τις πηγές που θα δημιουργήσουν θα έχουν πρόσθετο bonus κατά τη τελική βαθμολόγηση.