

Τεχνολογικό Εκπαιδευτικό Ίδρυμα Σερρών
Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Πληροφορικής και Επικοινωνιών

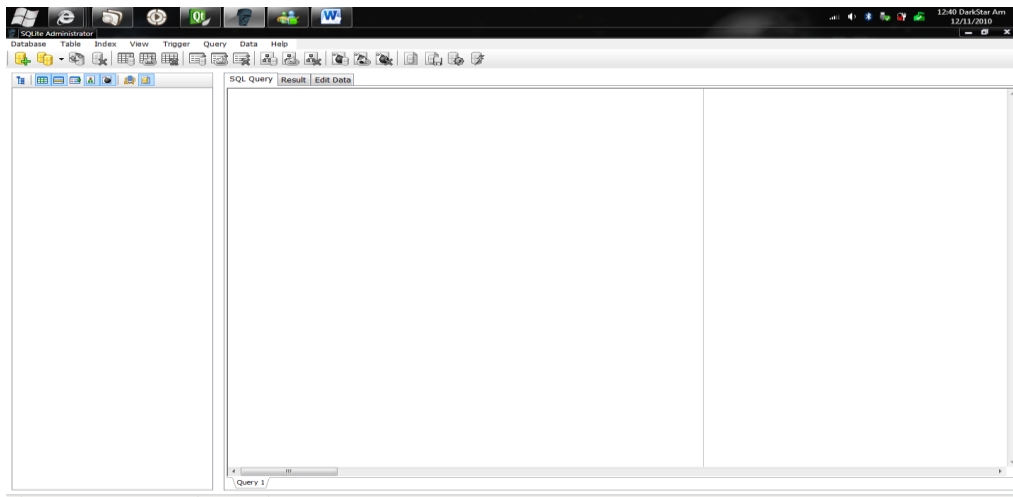
Πλήρες παράδειγμα σύνδεσης σε βάση
δεδομένων με QT + SQLite

Γεωργουδάκης Ιωάννης

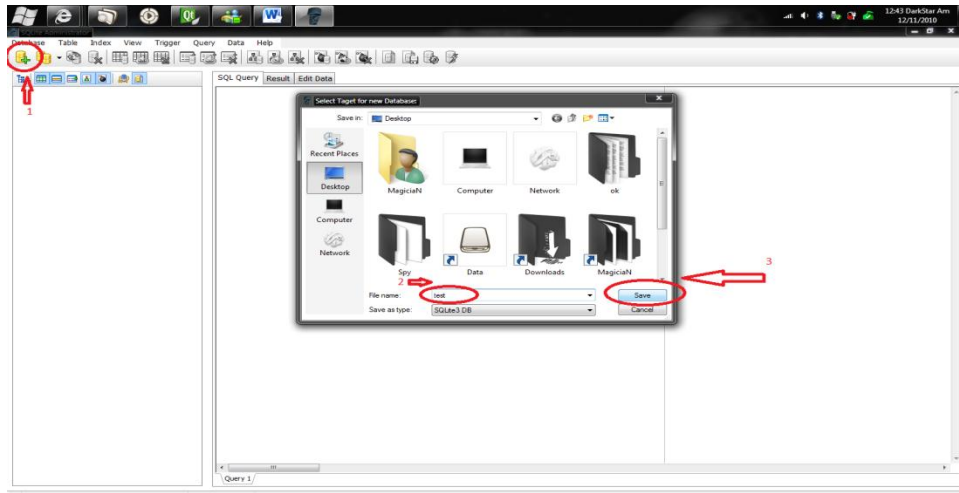
Το παράδειγμα παρακάτω θα σας βοηθήσει να κατανοήσετε πως μπορείτε να συνδεθείτε σε μία βάση δεδομένων SQLite με QT . Επίτηδες έχει κατασκευαστεί σε Console Application , διότι η μεταφορά αυτής της λογικής σε φόρμα είναι πιο απλή από 'τι είναι η αντίστροφη διαδικασία . Τέλος το παράδειγμα έγινε κάτω από περιβάλλον Windows .Αυτό όμως δεν σημαίνει κάτι διαφορετικό σε φάση υλοποίησης παρά μόνο του γεγονότος ότι θα πρέπει να χρησιμοποιηθεί μία εναλλακτική εφαρμογή για να παράγει SQLite datasources αρχεία . Θα το ξεκαθαρίσουμε και παρακάτω .

Αρχικά κατεβάστε το SQLite Administrator από την διεύθυνση <http://sqliteadmin.orbmu2k.de/> . Εναλλακτικά σε περιβάλλον Unix ή Linux κατεβάστε κάποιο αντίστοιχο . Κοιτάξτε εδώ θα βρείτε αρκετά : <http://www.sqlite.org/cvstrac/wiki?p=ManagementTools>

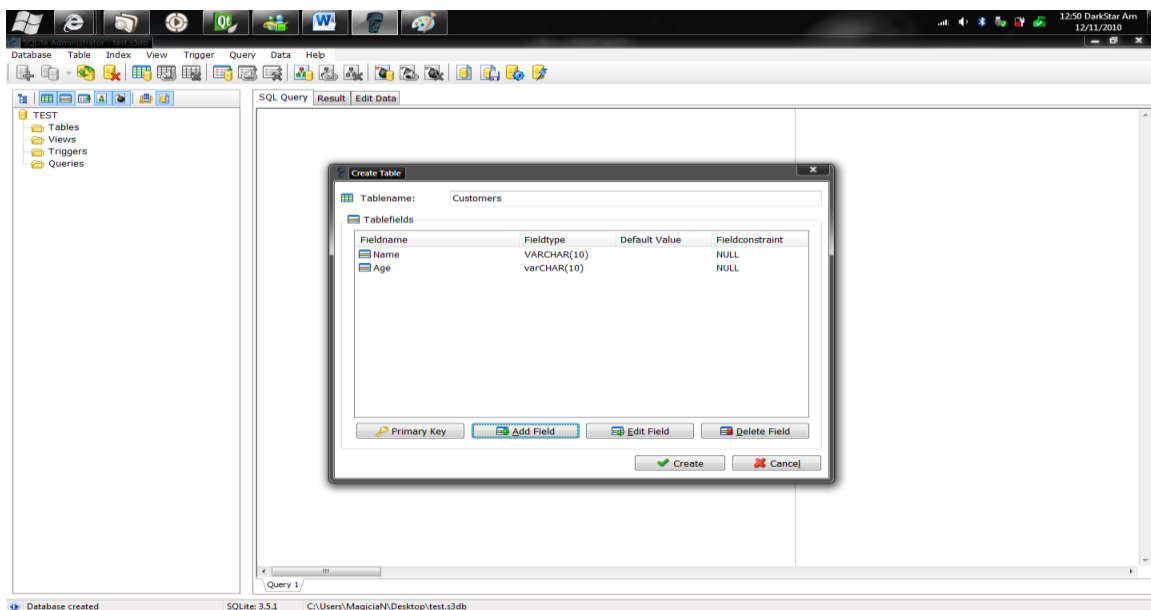
Εμείς θα συνεχίσουμε με αυτό το εργαλείο απλώς για να παράξουμε το αρχείο βάσης . Να σημειώσω ότι δεν χρειάζεται να τρέχει κάποιος SQL Server . Αφού κάνουμε την εγκατάσταση του SQLite Administrator τρέχουμε την εφαρμογή μας και βλέπουμε την αρχική οθόνη όπως φαίνεται παρακάτω.



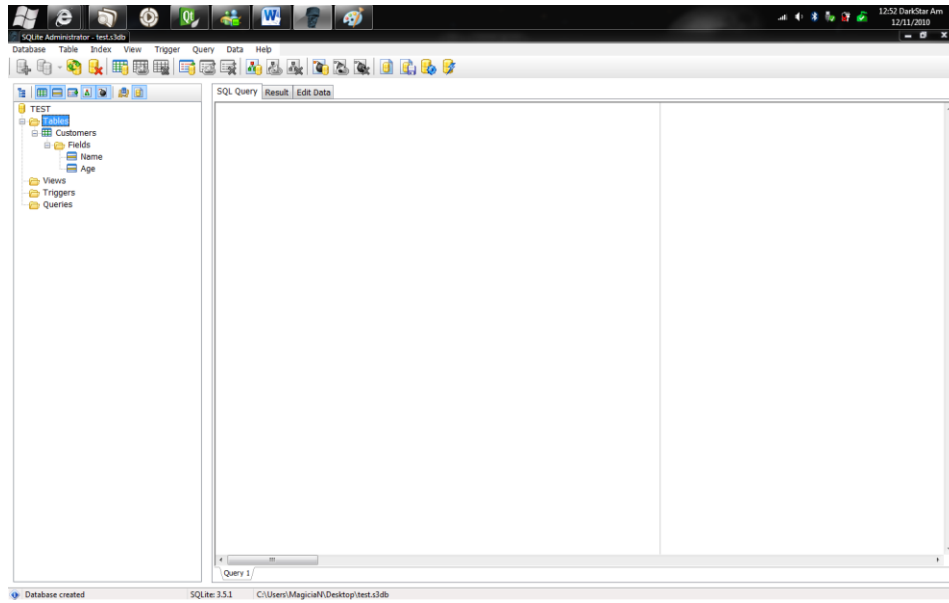
Έπειτα πατάμε το πρώτο εικονίδιο πάνω αριστερά ώστε να δημιουργήσουμε μία καινούρια βάση δεδομένων. Στον διάλογο που θα ανοίξει συμπληρώνουμε το όνομα της βάσης. Σε αυτό το παράδειγμα θα χρησιμοποιηθεί το όνομα 'test' οπότε για λόγους ευκολίας ονομάστε και εσείς την βάση σας έτσι.



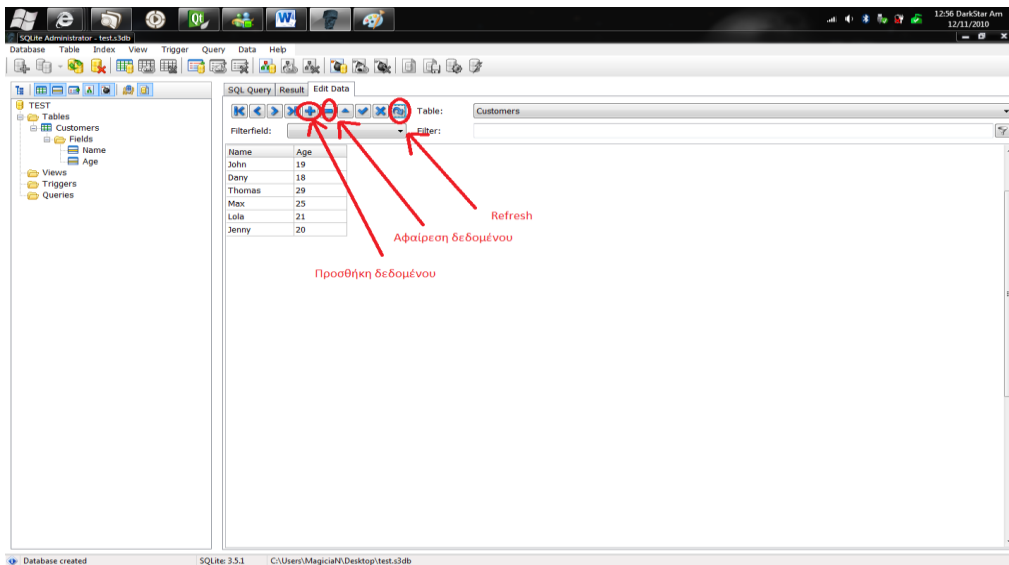
Έπειτα αριστερά θα δούμε να δημιουργούνται υποφάκελοι και άλλα που αφορούν την βάση μας .Εμείς θα κάνουμε δεξί κλικ στο 'tables' και θα επιλέξουμε 'create table'. Στο 'table name' τοποθετούμε 'Customers' και πατάμε στο ίδιο παράθυρο το κουμπί 'add field' . Στο νέο παράθυρο λοιπόν τοποθετούμε στο 'field name' => 'Name' και 'field type' => 'VARCHAR(10)' έπειτα πατάμε το 'add'. Ξανακάνουμε την ίδια διαδικασία ώστε να προσθέσουμε ένα πεδίο 'age' στην βάση μας . Όταν τελειώσουμε θα πρέπει να φαίνεται κάπως έτσι :



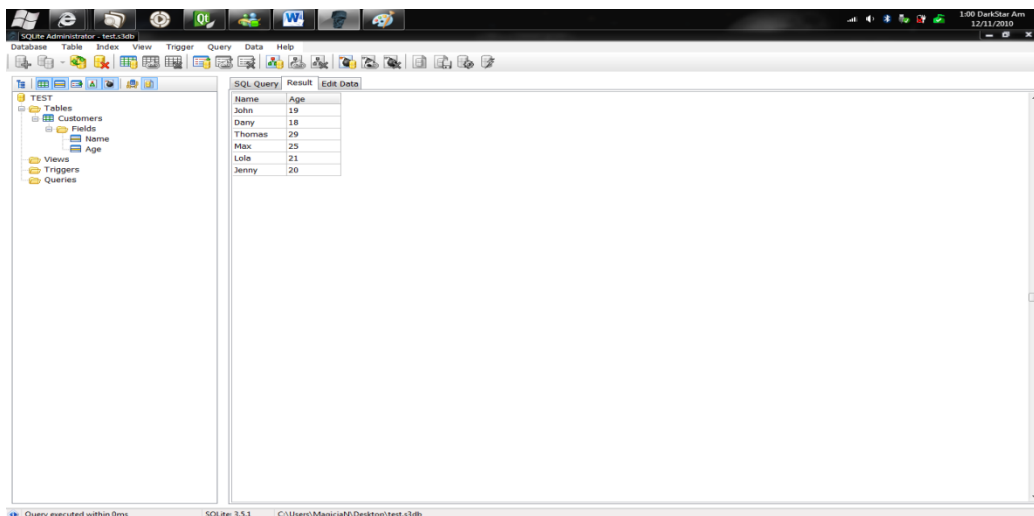
Συνεχίζοντας και για να βεβαιωθούμε ότι όλα έχουν πάει καλά πρέπει να διαπιστώσουμε πως έχουν δημιουργηθεί τα πεδία μας ώστε να τοποθετήσουμε τιμές. Αυτό το κάνουμε από αριστερά εκεί που φαίνεται το 'tables' πατάμε στο σημείο '+' και γίνεται expand. Τελικά και έπειτα από όλα αυτά πρέπει να φαίνεται κάτι τέτοιο:



Ήρθε λοιπόν η στιγμή να τοποθετήσουμε τιμές στο νέο μας πίνακα της νέας μας βάσης. Αυτό το κάνουμε πατώντας στην μέση του παραθύρου το tab 'edit data'. Για 'table' επιλέγουμε το 'Customers' και πατάμε μέσα στα κελιά ώστε να γράψουμε τιμές. Σε αυτό το σημείο τοποθετήστε ότι τιμές θέλετε. Για να προσθέσετε και άλλες πατήστε πάνω το κουμπι '+'. Στο τέλος κάντε ένα 'refresh' για να γίνουν sort τα δεδομένα:

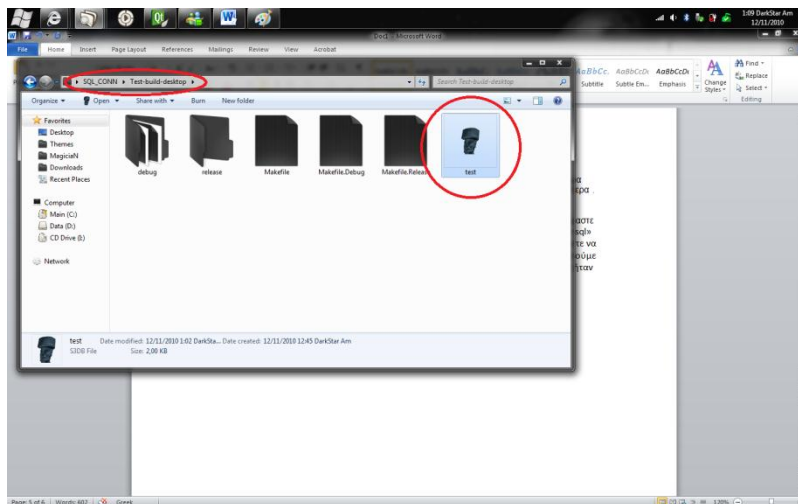


Έχουμε τελειώσει πλέον με το κομμάτι της βάσης . Εάν θέλετε για επαλήθευση εκτελέστε και ένα ερώτημα. Αυτό γίνεται πατώντας το tab SQL Query και γράφοντας ένα ερώτημα πχ `SELECT * FROM Customers` και πατάμε το προτελευταίο κουμπί για να εκτελεστεί. Έπειτα στο tab Result βλέπουμε τα αποτελέσματα του ερωτήματός μας . Εάν όλα έχουν πάει καλά μέχρι στιγμής θα πρέπει να δείτε κάτι τέτοιο:



Το αρχείο τελικά της βάσης βρίσκεται εκεί όπου το αποθηκεύσαμε . Κλείστε το πρόγραμμα SQLite Adminsitrator και κρατήστε την τοποθεσία του αρχείου γιατί θα το χρειαστούμε αργότερα .

Ανοίγουμε έπειτα τον QT Creator και δημιουργούμε ένα Console Application. Εκεί όπως είμαστε πηγαίνουμε πάνω στο αρχείο <ProjectName>.pro και τοποθετούμε την γραμμή «QT +=sql» για να έχουμε SQL Support στο project μας .Έπειτα κάνουμε build το project μία φορά ώστε να δημιουργήσει φακέλους debug κτλ.Έπειτα ανοίγουμε το φάκελο του project και τοποθετούμε το αρχείο βάσης δεδομένων “test.s3db” μέσα στο φάκελο <Project Name>-build-desktop .Στην περίπτωση μου ήταν κάπως έτσι:



Πηγαίνουμε μέσα στο αρχείο .cpp και γράφουμε τον παρακάτω κώδικα . Θα τον εξηγήσουμε μετέπειτα .Προτείνεται να σβήσετε ότι υπάρχει μέσα στο .cpp και να κάνετε αντιγραφή τον κώδικα που παραθέτω.

```

1  #include <QCoreApplication>
2  #include <direct.h>
3  #include <QtSql>
4
5  int main()
6  {
7      QSqlDatabase dataBase = QSqlDatabase::addDatabase("QSQLITE");
8      QString databasePath = getcwd(NULL, 0);
9
10     databasePath += "\\test.s3db" ;
11
12     dataBase.setDatabaseName(databasePath);
13
14     if ( dataBase.open() )
15     {
16         QSqlQuery query(dataBase);
17         query.exec("SELECT * from Customers " );
18
19         while ( query.next() )
20         {
21             qDebug() << query.value(0).toString() ;
22             qDebug() << query.value(1).toString() ;
23         }
24     }
25
26     dataBase.close();
27     getchar() ;
28
29     return 0;
30 }
31

```

Στην γραμμή 2 δηλώσαμε το <direct.h> η οποία βρίσκεται στην std βιβλιοθήκη της c και μας επιτρέπει να χρησιμοποιούμε την συνάρτηση getcwd() ώστε να παίρνουμε το path του προγράμματός μας δυναμικά.

Στην γραμμή 3 το <QtSql> μας βοηθά να χρησιμοποιήσουμε αντικείμενα ώστε θα μας φανούν χρήσιμα αργότερα ώστε να μιλήσουμε με τη βάση μας.

Στην 7 γραμμή δηλώνουμε ένα αντικείμενο QSqlDatabase και το καλούμε να φορτώσει τον driver για SQLITE όπως φαίνεται και παραπάνω.

Στην γραμμή 8 το QString αντικείμενο databasePath θα κρατήσει μέσα το path που βρίσκεται το datasource της βάσης .

Στην γραμμή 10 συμπληρώνουμε το databasePath με το όνομα της βάσης μας διότι χρειαζόμαστε την πλήρη διαδρομή για αυτή .

Έπειτα στη γραμμή 12 ρυθμίζουμε την βάση μας να κοιτά στο αρχείο που δημιουργήσαμε πιο πριν .

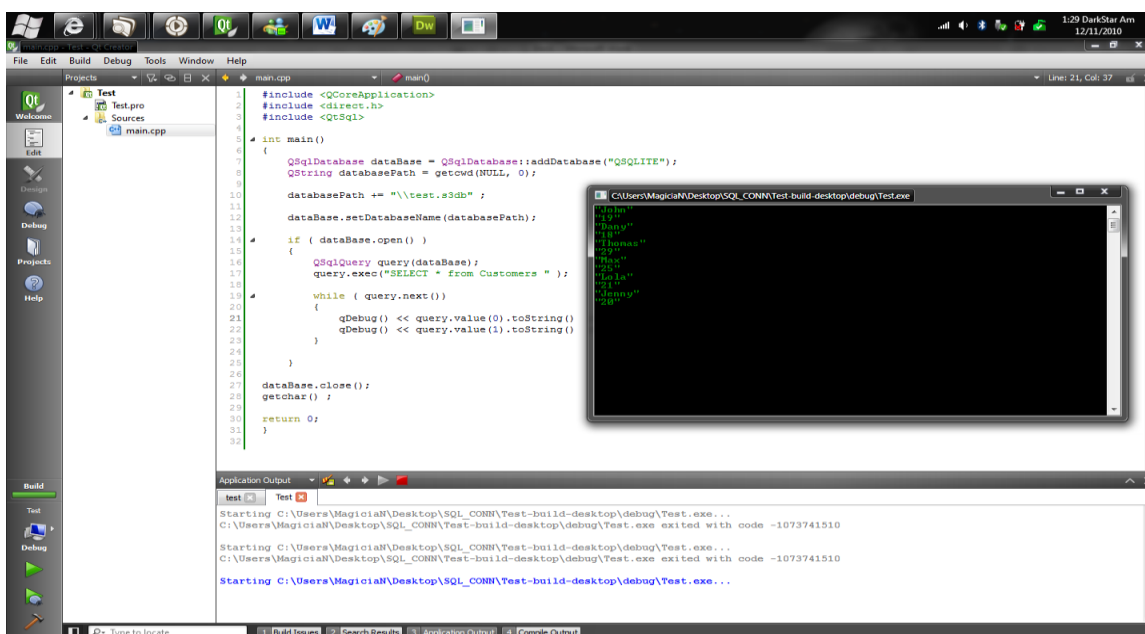
Στην γραμμή 14 'ανοίγουμε' πλέον την σύνδεση με τη βάση και αν επιστρέψει true προχωράμε παρακάτω . Στην γραμμή 16 δηλώνουμε ένα αντικείμενο τύπου QSqlQuery και το βάζουμε να κοιτάει στη βάση ώστε να κάνει ερωτήματα σε αυτήν.

Στην 17 γραμμή με αυτή τη συνάρτηση εκτελούμε το ερώτημα SELECT .

Τώρα πλέον μέσα στο query έχουν επισταφεί τα αποτελέσματα του ερωτήματος . Να σημειώσω ότι σε αυτό το σημείο θα μπορούσαμε να έχουμε πεί πολλά και να κάνουμε πάμπολους ελέγχους . Εάν υπάρχει η βάση, εάν έχουν όντως επιστραφεί κάτι από το ερώτημα και άλλα. Το κεντρικό σημείο και η δυσκολία μας όμως εδώ είναι να κάνουμε ένα απλό ερώτημα και προς στιγμή επικεντρώναστε μόνο σε αυτό .

Συνεχίζοντας στην γραμμή 19 έως το τέλος λέμε ότι 'όσο έχεις επόμενα δεδομένα να τα εκτυπώνεις' . Έτσι κάνουμε 21 και 22 . Τα βούμερα μέσα στις συναρτήσεις(value(0) και value(1)) υποδηλώνουν τις στήλες του ερωτήματός μας δηλαδή name και age αντίστοιχα . Τα στέλνουμε στην qDebug() η οποία έχει την δυνατότητα να εκτυπώνει QObjects και τα μετετρέπουμε απευθείας σε strings . Να σημειώσω κάτι τέτοιο δεν θα γινόταν με την γνωστή cout << κτλ γιατί θέλει και ένα τέχνασμα για να το επιτύχουμε αυτό.

Το συνολικό αποτέλεσμα εάν τρέξουμε το πρόγραμμα θα είναι αυτό :



```
1 #include <QCoreApplication>
2 #include <direct.h>
3 #include <QtSql>
4
5 int main()
6 {
7     QSqlDatabase database = QSqlDatabase::addDatabase("QSQLITE");
8     QString databasePath = getcwd(NULL, 0);
9
10    databasePath += "\\test.s3db" ;
11    database.setDatabaseName(databasePath);
12
13    if ( database.open() )
14    {
15        QSqlQuery query(database);
16        query.exec("SELECT * from Customers " );
17
18        while ( query.next() )
19        {
20            qDebug() << query.value(0).toString()
21                << query.value(1).toString()
22                << endl;
23        }
24    }
25
26    database.close();
27    getchar();
28
29    return 0;
30 }
```

Application Output

```
Starting C:\Users\MagiciaN\Desktop\SQL_CONN\Test-build-desktop\debug\Test.exe...
C:\Users\MagiciaN\Desktop\SQL_CONN\Test-build-desktop\debug\Test.exe exited with code -1073741510
Starting C:\Users\MagiciaN\Desktop\SQL_CONN\Test-build-desktop\debug\Test.exe...
C:\Users\MagiciaN\Desktop\SQL_CONN\Test-build-desktop\debug\Test.exe exited with code -1073741510
Starting C:\Users\MagiciaN\Desktop\SQL_CONN\Test-build-desktop\debug\Test.exe...
```

Test Output

```
John
Denny
Tom
Thomas
Wes
Wesley
Lola
Lola
Denny
20
```

Περιμένω σχόλια σας .

Happy Coding !

