

Adatbázis rendszerek I

Gyakorlati feladatok

11. hét

Témakör: SQL API (SQLite, C)

Miskolc, 2020

Készítette:
dr. Kovács László

1. Feladat

Mutassa be, hogy miben különbözik egymástól a CLI és E-SQL interfészek.

Útmutató:

szempont	E-SQL	CLI
SQL parancs megadása	változatlan alakban, de egy megadott prefix megadása után	függvény hívás argumentumában
adatbázis kapcsolatok	alapesetben egy adatbázissal kommunikál	több adatbázissal, DB azonosító változó használata
parancsok jellege	csak SQL	SQL és nem-SQL jellegű parancsok is kiadhatóak
elterjedtség	ritkán használt	elterjedtebb

2. Feladat

Mutassa be, milyen lépésekben lehet a Codeblocks-ból az SQLite adatbázis elérni.

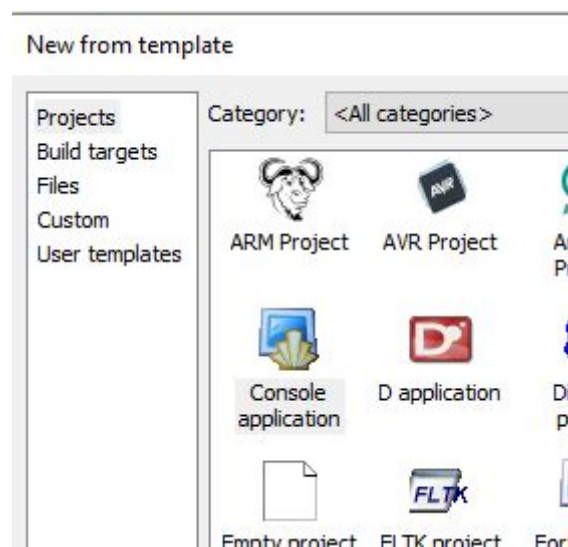
Útmutató:

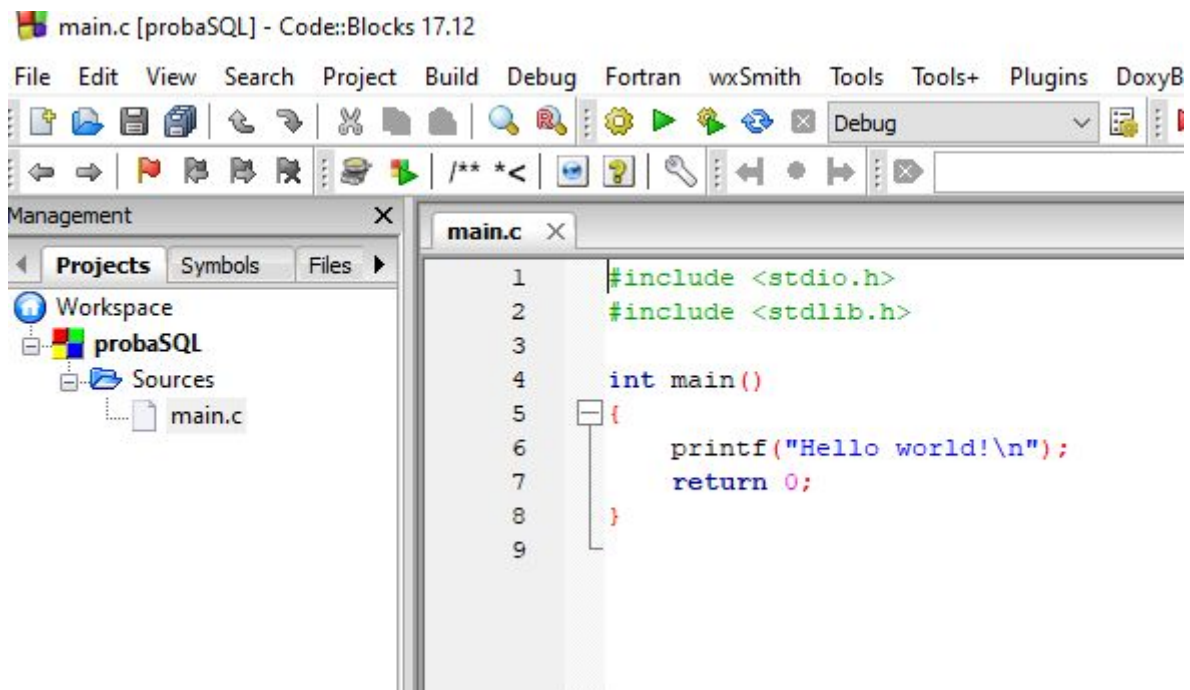
1. SQLite letöltése
2. SQLite adatbázis létrehozása

```
2 Dir(s) 601,716,035,584 bytes free

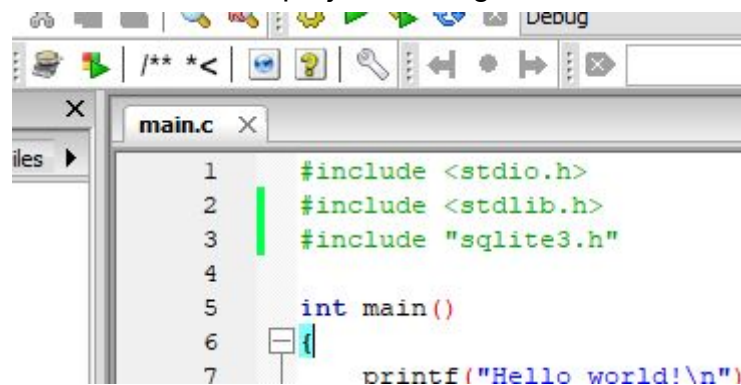
D:\sqlite>sqlite3 ujdbc
SQLite version 3.31.1 2020-01-27 19:55:54
Enter ".help" for usage hints.
sqlite> create table autok (rsz char(6), tipus char(30), ar int);
sqlite> insert into autok values (1,'Opel',56666);
sqlite> insert into autok values (2,'Fiat',66);
sqlite> select * from autok;
1|Opel|56666
2|Fiat|66
sqlite>
```

3. CodeBlocks C projekt létrehozatala

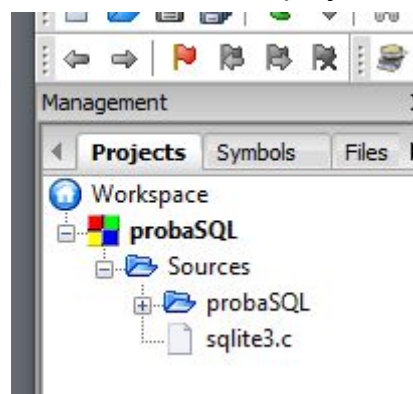




4. sqlite3.h header file átmásolása projekt katalógusba



5. A sq.lite3.c letöltött kódfájl beillesztése a projektbe



3. Feladat

Kérdezze le az SQLite rendszer verzióját C programból!

Útmutató:

```
#include <sqlite3.h>
#include <stdio.h>

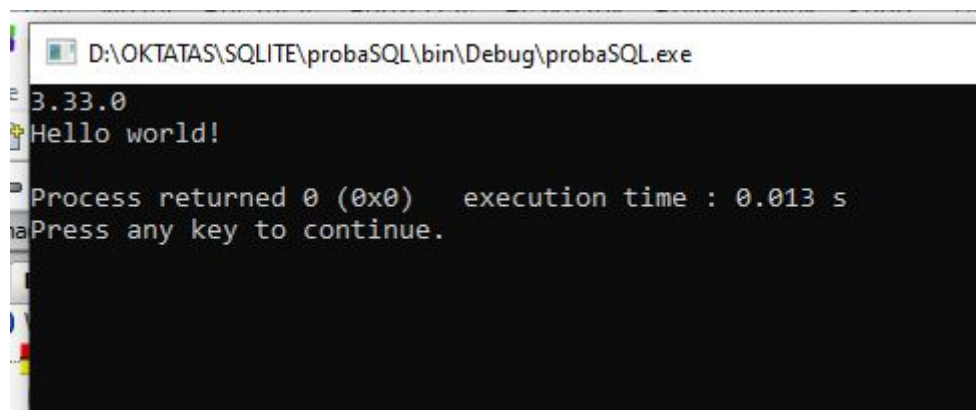
int main(void) {

    printf("%s\n", sqlite3_libversion());

    return 0;
}
```

A screenshot of a code editor window titled 'main.c'. The code is as follows:

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include "sqlite3.h"
4
5  int main()
6  {
7      printf("%s\n", sqlite3_libversion());
8
9      printf("Hello world!\n");
10     return 0;
11 }
12
```

A screenshot of a Windows command prompt window. The title bar shows the path 'D:\OKTATAS\SQLITE\probaSQL\bin\Debug\probaSQL.exe'. The output of the program is displayed as follows:

```
3.33.0
Hello world!

Process returned 0 (0x0)   execution time : 0.013 s
Press any key to continue.
```

4. Feladat

Csatlakozzon egy meglévő SQLite adatbázishoz


Útmutató:

```
#include <stdio.h>
#include <sqlite3.h>

int main(int argc, char* argv[]) {
    sqlite3 *db;
    int rc;

    rc = sqlite3_open("test.db", &db);

    if( rc ) {
        fprintf(stderr, "Hiba a megnyitáskor: %s\n", sqlite3_errmsg(db));
        return(0);
    } else {
        fprintf(stderr, "Sikeresen megnyitva\n");
    }
    sqlite3_close(db);
}
```



```
int main(int argc, char* argv[]) {
    sqlite3 *db;
    int rc;

    rc = sqlite3_open("D:/sqlite/ujdb", &db);

    if( rc ) {
        fprintf(stderr, "Hiba a megnyitáskor: %s\n", sqlite3_errmsg(db));
        return(0);
    } else {
        fprintf(stderr, "Sikeresen megnyitva\n");
    }
    sqlite3_close(db);
}
```

5. Feladat

Hozzon létre egy *Cars*(*Id* INT, *Name* TEXT, *Price* INT) adattáblát és vigye fel több új rekordot a táblába, a hol az adatok fixen adottak a programban.

Útmutató

```
#include <sqlite3.h>
#include <stdio.h>

int main(void) {

    sqlite3 *db;
    char *hibaUzenet = 0;

    int rc = sqlite3_open("test.db", &db);

    if (rc != SQLITE_OK) {

        fprintf(stderr, "Hiba a megnyitáskor: %s\n", sqlite3_errmsg(db));
        sqlite3_close(db);

        return 1;
    }

    char *sql = "DROP TABLE IF EXISTS Cars;"
               "CREATE TABLE Cars(Id INT, Name TEXT, Price INT);"
               "INSERT INTO Cars VALUES(1, 'Audi', 52642);"
               "INSERT INTO Cars VALUES(2, 'Mercedes', 57127);"
               "INSERT INTO Cars VALUES(3, 'Skoda', 9000);"
               "INSERT INTO Cars VALUES(4, 'Volvo', 29000);"
               "INSERT INTO Cars VALUES(5, 'Bentley', 350000);"
               "INSERT INTO Cars VALUES(6, 'Citroen', 21000);"
               "INSERT INTO Cars VALUES(7, 'Hummer', 41400);"
               "INSERT INTO Cars VALUES(8, 'Volkswagen', 21600);";

    rc = sqlite3_exec(db, sql, 0, 0, &hibaUzenet);
```

```

    if (rc != SQLITE_OK ) {

        fprintf(stderr, "SQL hiba: %s\n", hibaUzenet);

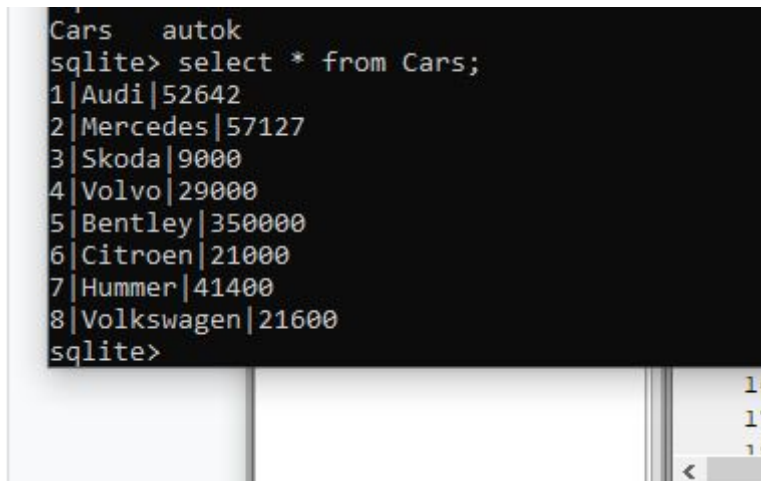
        sqlite3_free(hibaUzenet);
        sqlite3_close(db);

        return 1;
    }

    sqlite3_close(db);

    return 0;
}

```



The screenshot shows a terminal window with a SQLite prompt. The user has entered the command 'select * from Cars;' and the output displays a list of cars with their IDs, names, and prices. The data is as follows:

ID	Tipus	Ar
1	Audi	52642
2	Mercedes	57127
3	Skoda	9000
4	Volvo	29000
5	Bentley	350000
6	Citroen	21000
7	Hummer	41400
8	Volkswagen	21600

6. Feladat

Olvassa be egy új autó adatait a képernyőről és vigye fel egy új rekordként a Cars(rsz, típus, ar) táblába.

Útmutató:

```

#include <stdio.h>
#include <stdlib.h>
#include "sqlite3.h"

```



```

int main(int argc, char* argv[]) {
    sqlite3 *db;
    int rc;
    char *hibaUzenet = 0;
    char rsz[30];
    char tipus[30];
    int ar;
    char sar[30];

    rc = sqlite3_open("D:/sqlite/ujdb", &db);

    if( rc ) {
        fprintf(stderr, "Hiba a megnyitáskor: %s\n", sqlite3_errmsg(db));
        return(0);
    } else {
        fprintf(stderr, "Sikeresen megnyitva\n");
    }

    printf ("rsz=");
    scanf ("%s",&rsz);
    printf ("tipus=");
    scanf ("%s",&tipus);
    printf ("ar=");
    scanf ("%d",&ar);
    sprintf(sar, "%d", ar);

    char sql[200];
    strcpy(sql, "INSERT INTO Cars VALUES('");
    strcat(sql,rsz);
    strcat(sql,"", "");
    strcat(sql, tipus);
    strcat(sql,"", "");
    strcat(sql,sar);
    strcat(sql,"");

    printf ("%s\n", sql);

    rc = sqlite3_exec(db, sql, 0, 0, &hibaUzenet);

    if (rc != SQLITE_OK ) {

```

```

        fprintf(stderr, "SQL hiba: %s\n", hibaUzenet);

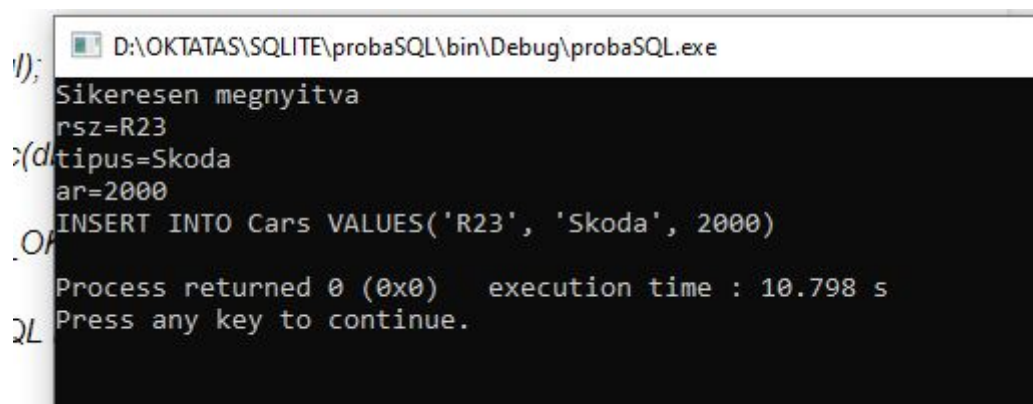
        sqlite3_free(hibaUzenet);
        sqlite3_close(db);

        return 1;
    }

    sqlite3_close(db);

    return 0;
}

```



```

D:\OKTATAS\SQLITE\probaSQL\bin\Debug\probaSQL.exe
Sikeresen megnyitva
rsz=R23
tipus=Skoda
INSERT INTO Cars VALUES('R23', 'Skoda', 2000)
Process returned 0 (0x0)   execution time : 10.798 s
Press any key to continue.

```