Name: Tan Han Nguyen

NetID: TXN200004

Week 14 Lab 1

0. Login

Pre-authentication banner message from server:

| University of Texas at Dallas

| Department of Computer Science

|

| Use of UTD Information Systems is subject to

| the UTD Information Security and Acceptable Use Policy.

|

| Pursuant to Texas Administrative Code 202:

| (1) Unauthorized use is prohibited;

| (2) Usage may be subject to security testing and monitoring;

| (3) Misuse is subject to criminal prosecution; and

| (4) No expectation of privacy except as otherwise provided by applicable

| privacy laws.

|

| ATTENTION: utdnetid != utdnetid@utdallas.edu (UTD != Google!)

|

| \*\*\*\*\* This system will require a connection to the GlobalProtect VPN startin

> g

| on the following dates:

|

| cslinux1.utdallas.edu - June 15, 2020

| cslinux2.utdallas.edu - June 22, 2020

|

| \*\*\*\*\* GlobalProtect VPN Instructions: https://www.utdallas.edu/oit/howto/vpn

> /

|

End of banner message from server

Keyboard-interactive authentication prompts from server:

| Password:

End of keyboard-interactive prompts from server

Access denied

Keyboard-interactive authentication prompts from server:

| Password:

End of keyboard-interactive prompts from server

┌──────────────────────────────────────────────────────────────────────┐

│ • MobaXterm Personal Edition v24.2 • │

│ (SSH client, X server and network tools) │

│ │

│ ⮞ SSH session to txn200004@cslinux2.utdallas.edu │

│ • Direct SSH : ✓ │

│ • SSH compression : ✓ │

│ • SSH-browser : ✓ │

│ • X11-forwarding : ✓ (remote display is forwarded through SSH) │

│ │

│ ⮞ For more info, ctrl+click on help or visit our website. │

└──────────────────────────────────────────────────────────────────────┘

Last failed login: Tue Nov 19 17:36:56 CST 2024 from 10.50.241.82 on ssh:notty

There was 1 failed login attempt since the last successful login.

Last login: Tue Nov 12 12:51:30 2024 from 10.176.205.105

\*\*\*---\*\*\*---\*\*\*---\*\*\*---\*\*\*---\*\*\*

csgrads1.utdallas.edu - CentOS Linux 7.9

--All CS Graduate Students should use csgrads1--

cs1.utdallas.edu - CentOS Linux 7.9

cs2.utdallas.edu - CentOS Linux 7.9

\*\*\*---\*\*\*---\*\*\*---\*\*\*---\*\*\*---\*\*\*

This system is for use by CS students who need a general purpose Linux system

to complete homework assignments. Computationally or resource intensive

simulations will be throttled automatically.

Thank you,

CS Lab Manager

cs-labs@utdallas.edu

/scratch disk space can be used for temporary files.

All files within /scratch will be erased on a regular basis (Sunday 0300).

1. Download files into cs2

{cslinux2:~} mkdir week14Lab1; cd week14Lab1

{cslinux2:~/week14Lab1} cd sqlite/

{cslinux2:~/week14Lab1/sqlite} touch test.db

2. Compile with g++

{cslinux2:~/week14Lab1/sqlite} g++ test0.c -o test0 -l sqlite3

{cslinux2:~/week14Lab1/sqlite} cp ../test6.c .

{cslinux2:~/week14Lab1/sqlite} g++ test6.c -o test6 -l sqlite3

**test6.c:** In function ‘**int main(int, char\*\*)**’:

**test6.c:28:18:** warning: deprecated conversion from string constant to ‘**char\***’ [-Wwrite-strings]

char \*query = "select \* from COMPANY;";

^

3. Run each program and check with sqlite3

./test0

{cslinux2:~/week14Lab1/sqlite} ./test0

Opened database successfully

{cslinux2:~/week14Lab1/sqlite} sqlite3 test.db

SQLite version 3.14.1 2016-08-11 18:53:32

Enter ".help" for usage hints.

sqlite> .help

.auth ON|OFF Show authorizer callbacks

.backup ?DB? FILE Backup DB (default "main") to FILE

.bail on|off Stop after hitting an error. Default OFF

.binary on|off Turn binary output on or off. Default OFF

.changes on|off Show number of rows changed by SQL

.clone NEWDB Clone data into NEWDB from the existing database

.databases List names and files of attached databases

.dbinfo ?DB? Show status information about the database

.dump ?TABLE? ... Dump the database in an SQL text format

If TABLE specified, only dump tables matching

LIKE pattern TABLE.

.echo on|off Turn command echo on or off

.eqp on|off|full Enable or disable automatic EXPLAIN QUERY PLAN

.exit Exit this program

.explain ?on|off|auto? Turn EXPLAIN output mode on or off or to automatic

.fullschema ?--indent? Show schema and the content of sqlite\_stat tables

.headers on|off Turn display of headers on or off

.help Show this message

.import FILE TABLE Import data from FILE into TABLE

.indexes ?TABLE? Show names of all indexes

If TABLE specified, only show indexes for tables

matching LIKE pattern TABLE.

.limit ?LIMIT? ?VAL? Display or change the value of an SQLITE\_LIMIT

.load FILE ?ENTRY? Load an extension library

.log FILE|off Turn logging on or off. FILE can be stderr/stdout

.mode MODE ?TABLE? Set output mode where MODE is one of:

ascii Columns/rows delimited by 0x1F and 0x1E

csv Comma-separated values

column Left-aligned columns. (See .width)

html HTML <table> code

insert SQL insert statements for TABLE

line One value per line

list Values delimited by .separator strings

tabs Tab-separated values

tcl TCL list elements

.nullvalue STRING Use STRING in place of NULL values

.once FILENAME Output for the next SQL command only to FILENAME

.open ?FILENAME? Close existing database and reopen FILENAME

.output ?FILENAME? Send output to FILENAME or stdout

.print STRING... Print literal STRING

.prompt MAIN CONTINUE Replace the standard prompts

.quit Exit this program

.read FILENAME Execute SQL in FILENAME

.restore ?DB? FILE Restore content of DB (default "main") from FILE

.save FILE Write in-memory database into FILE

.scanstats on|off Turn sqlite3\_stmt\_scanstatus() metrics on or off

.schema ?PATTERN? Show the CREATE statements matching PATTERN

Add --indent for pretty-printing

.separator COL ?ROW? Change the column separator and optionally the row

separator for both the output mode and .import

.shell CMD ARGS... Run CMD ARGS... in a system shell

.show Show the current values for various settings

.stats ?on|off? Show stats or turn stats on or off

.system CMD ARGS... Run CMD ARGS... in a system shell

.tables ?TABLE? List names of tables

If TABLE specified, only list tables matching

LIKE pattern TABLE.

.timeout MS Try opening locked tables for MS milliseconds

.timer on|off Turn SQL timer on or off

.trace FILE|off Output each SQL statement as it is run

.vfsinfo ?AUX? Information about the top-level VFS

.vfslist List all available VFSes

.vfsname ?AUX? Print the name of the VFS stack

.width NUM1 NUM2 ... Set column widths for "column" mode

Negative values right-justify

sqlite> .databases

seq name file

--- --------------- ----------------------------------------------------------

0 main /home/010/t/tx/txn200004/week14Lab1/sqlite/test.db

sqlite> .quit

./test1

{cslinux2:~/week14Lab1/sqlite} ./test1

-bash: ./test1: Permission denied

{cslinux2:~/week14Lab1/sqlite} chmod 777 test\*

{cslinux2:~/week14Lab1/sqlite} ./test1

Opened database successfully

Table created successfully

{cslinux2:~/week14Lab1/sqlite} sqlite3 test.db

SQLite version 3.14.1 2016-08-11 18:53:32

Enter ".help" for usage hints.

sqlite> .databases

seq name file

--- --------------- ----------------------------------------------------------

0 main /home/010/t/tx/txn200004/week14Lab1/sqlite/test.db

sqlite> .table

COMPANY

sqlite> .schema COMPANY

CREATE TABLE COMPANY(ID INT PRIMARY KEY NOT NULL,NAME TEXT NOT NULL,AGE INT NOT NULL,ADDRESS CHAR(50),SALARY REAL );

sqlite> .dump COMPANY

PRAGMA foreign\_keys=OFF;

BEGIN TRANSACTION;

CREATE TABLE COMPANY(ID INT PRIMARY KEY NOT NULL,NAME TEXT NOT NULL,AGE INT NOT NULL,ADDRESS CHAR(50),SALARY REAL );

COMMIT;

sqlite> .exit

./test2

{cslinux1:~/week14Lab1/sqlite} ./test2

Opened database successfully

Records created successfully

{cslinux1:~/week14Lab1/sqlite} sqlite3 test.db

SQLite version 3.14.1 2016-08-11 18:53:32

Enter ".help" for usage hints.

sqlite> .tables

COMPANY

sqlite> .dump COMPANY

PRAGMA foreign\_keys=OFF;

BEGIN TRANSACTION;

CREATE TABLE COMPANY(ID INT PRIMARY KEY NOT NULL,NAME TEXT NOT NULL,AGE INT NOT NULL,ADDRESS CHAR(50),SALARY REAL );

INSERT INTO "COMPANY" VALUES(1,'Paul',32,'California',20000.0);

INSERT INTO "COMPANY" VALUES(2,'Allen',25,'Texas',15000.0);

INSERT INTO "COMPANY" VALUES(3,'Teddy',23,'Norway',20000.0);

INSERT INTO "COMPANY" VALUES(4,'Mark',25,'Rich-Mond ',65000.0);

COMMIT;

sqlite> .exit

./test3

{cslinux2:~/week14Lab1/sqlite} ./test3

Opened database successfully

Callback function called: ID = 1

NAME = Paul

AGE = 32

ADDRESS = California

SALARY = 20000.0

Callback function called: ID = 2

NAME = Allen

AGE = 25

ADDRESS = Texas

SALARY = 15000.0

Callback function called: ID = 3

NAME = Teddy

AGE = 23

ADDRESS = Norway

SALARY = 20000.0

Callback function called: ID = 4

NAME = Mark

AGE = 25

ADDRESS = Rich-Mond

SALARY = 65000.0

Operation done successfully

{cslinux2:~/week14Lab1/sqlite} sqlite3 test.db

SQLite version 3.14.1 2016-08-11 18:53:32

Enter ".help" for usage hints.

sqlite> .dump COMPANY

PRAGMA foreign\_keys=OFF;

BEGIN TRANSACTION;

CREATE TABLE COMPANY(ID INT PRIMARY KEY NOT NULL,NAME TEXT NOT NULL,AGE INT NOT NULL,ADDRESS CHAR(50),SALARY REAL );

INSERT INTO "COMPANY" VALUES(1,'Paul',32,'California',20000.0);

INSERT INTO "COMPANY" VALUES(2,'Allen',25,'Texas',15000.0);

INSERT INTO "COMPANY" VALUES(3,'Teddy',23,'Norway',20000.0);

INSERT INTO "COMPANY" VALUES(4,'Mark',25,'Rich-Mond ',65000.0);

COMMIT;

sqlite> .output COMPANY.sql

sqlite> .dump COMPANY

sqlite> .exit

{cslinux2:~/week14Lab1/sqlite} cat COMPANY.sql

PRAGMA foreign\_keys=OFF;

BEGIN TRANSACTION;

CREATE TABLE COMPANY(ID INT PRIMARY KEY NOT NULL,NAME TEXT NOT NULL,AGE INT NOT NULL,ADDRESS CHAR(50),SALARY REAL );

INSERT INTO "COMPANY" VALUES(1,'Paul',32,'California',20000.0);

INSERT INTO "COMPANY" VALUES(2,'Allen',25,'Texas',15000.0);

INSERT INTO "COMPANY" VALUES(3,'Teddy',23,'Norway',20000.0);

INSERT INTO "COMPANY" VALUES(4,'Mark',25,'Rich-Mond ',65000.0);

COMMIT;

./test4

{cslinux2:~/week14Lab1/sqlite} ./test4

Opened database successfully

Callback function called: ID = 1

NAME = Paul

AGE = 32

ADDRESS = California

SALARY = 25000.0

Callback function called: ID = 2

NAME = Allen

AGE = 25

ADDRESS = Texas

SALARY = 15000.0

Callback function called: ID = 3

NAME = Teddy

AGE = 23

ADDRESS = Norway

SALARY = 20000.0

Callback function called: ID = 4

NAME = Mark

AGE = 25

ADDRESS = Rich-Mond

SALARY = 65000.0

Operation done successfully

{cslinux2:~/week14Lab1/sqlite} sqlite3 test.db

SQLite version 3.14.1 2016-08-11 18:53:32

Enter ".help" for usage hints.

sqlite> .dump COMPANY

PRAGMA foreign\_keys=OFF;

BEGIN TRANSACTION;

CREATE TABLE COMPANY(ID INT PRIMARY KEY NOT NULL,NAME TEXT NOT NULL,AGE INT NOT NULL,ADDRESS CHAR(50),SALARY REAL );

INSERT INTO "COMPANY" VALUES(1,'Paul',32,'California',25000.0);

INSERT INTO "COMPANY" VALUES(2,'Allen',25,'Texas',15000.0);

INSERT INTO "COMPANY" VALUES(3,'Teddy',23,'Norway',20000.0);

INSERT INTO "COMPANY" VALUES(4,'Mark',25,'Rich-Mond ',65000.0);

COMMIT;

sqlite> .exit

./test5

{cslinux2:~/week14Lab1/sqlite} ./test5

Opened database successfully

Callback function called: ID = 1

NAME = Paul

AGE = 32

ADDRESS = California

SALARY = 25000.0

\*Allen removed\*

Callback function called: ID = 3

NAME = Teddy

AGE = 23

ADDRESS = Norway

SALARY = 20000.0

Callback function called: ID = 4

NAME = Mark

AGE = 25

ADDRESS = Rich-Mond

SALARY = 65000.0

Operation done successfully

{cslinux2:~/week14Lab1/sqlite} sqlite3 test.db

SQLite version 3.14.1 2016-08-11 18:53:32

Enter ".help" for usage hints.

sqlite> .dump COMPANY

PRAGMA foreign\_keys=OFF;

BEGIN TRANSACTION;

CREATE TABLE COMPANY(ID INT PRIMARY KEY NOT NULL,NAME TEXT NOT NULL,AGE INT NOT NULL,ADDRESS CHAR(50),SALARY REAL );

INSERT INTO "COMPANY" VALUES(1,'Paul',32,'California',25000.0);

INSERT INTO "COMPANY" VALUES(3,'Teddy',23,'Norway',20000.0);

INSERT INTO "COMPANY" VALUES(4,'Mark',25,'Rich-Mond ',65000.0);

COMMIT;

sqlite> .exit

./test6

{cslinux2:~/week14Lab1/sqlite} ./test6

Opened database successfully

1 Paul 32 California 25000.0

3 Teddy 23 Norway 20000.0

4 Mark 25 Rich-Mond 65000.0

done

{cslinux2:~/week14Lab1/sqlite} sqlite3 test.db

SQLite version 3.14.1 2016-08-11 18:53:32

Enter ".help" for usage hints.

.Compare COMPANY.sql and ‘select from’ command

sqlite> .output COMPANY.sql

sqlite> .dump COMPANY

PRAGMA foreign\_keys=OFF;

BEGIN TRANSACTION;

CREATE TABLE COMPANY(ID INT PRIMARY KEY NOT NULL,NAME TEXT NOT NULL,AGE INT NOT NULL,ADDRESS CHAR(50),SALARY REAL );

INSERT INTO "COMPANY" VALUES(1,'Paul',32,'California',25000.0);

INSERT INTO "COMPANY" VALUES(3,'Teddy',23,'Norway',20000.0);

INSERT INTO "COMPANY" VALUES(4,'Mark',25,'Rich-Mond ',65000.0);

COMMIT;

sqlite> select \* from COMPANY;

1|Paul|32|California|25000.0

3|Teddy|23|Norway|20000.0

4|Mark|25|Rich-Mond |65000.0

sqlite> .exit

{cslinux2:~/week14Lab1/sqlite} cat COMPANY.sql

PRAGMA foreign\_keys=OFF;

BEGIN TRANSACTION;

CREATE TABLE COMPANY(ID INT PRIMARY KEY NOT NULL,NAME TEXT NOT NULL,AGE INT NOT NULL,ADDRESS CHAR(50),SALARY REAL );

INSERT INTO "COMPANY" VALUES(1,'Paul',32,'California',25000.0);

INSERT INTO "COMPANY" VALUES(3,'Teddy',23,'Norway',20000.0);

INSERT INTO "COMPANY" VALUES(4,'Mark',25,'Rich-Mond ',65000.0);

COMMIT;

End Lab and EXIT

{cslinux2:~/week14Lab1/sqlite} ls -l

total 1800

-rw------- 1 txn200004 se 405 Nov 19 18:06 COMPANY.sql

-rw------- 1 txn200004 se 151981 Nov 19 17:41 shell.c

-rw------- 1 txn200004 se 623632 Nov 19 17:41 sqlite3

-rw------- 1 txn200004 se 221726 Nov 19 17:41 Sqlite3 tutorial.pptx

-rw------- 1 txn200004 se 207 Nov 19 17:41 # SQLite C-C++ Tutorial.url

-rw------- 1 txn200004 se 184 Nov 19 17:41 # SQLite Download Page.url

-rwxrwxrwx 1 txn200004 se 8592 Nov 19 17:49 test0

-rwxrwxrwx 1 txn200004 se 393 Nov 19 17:41 test0.c

-rwxrwxrwx 1 txn200004 se 9542 Nov 19 17:41 test1

-rwxrwxrwx 1 txn200004 se 1237 Nov 19 17:41 test1.c

-rwxrwxrwx 1 txn200004 se 9542 Nov 19 17:41 test2

-rwxrwxrwx 1 txn200004 se 1494 Nov 19 17:41 test2.c

-rwx--x--x 1 txn200004 se 8912 Nov 19 17:55 test3

-rwxrwxrwx 1 txn200004 se 1128 Nov 19 17:41 test3.c

-rwxrwxrwx 1 txn200004 se 9542 Nov 19 17:41 test4

-rwxrwxrwx 1 txn200004 se 1199 Nov 19 17:41 test4.c

-rwxrwxrwx 1 txn200004 se 9542 Nov 19 17:41 test5

-rwxrwxrwx 1 txn200004 se 1182 Nov 19 17:41 test5.c

-rwxrwxrwx 1 txn200004 se 14304 Nov 19 17:50 test6

-rwxrwxrwx 1 txn200004 se 1481 Nov 19 17:49 test6.c

-rw------- 1 txn200004 se 3072 Nov 19 18:01 test.db

{cslinux2:~/week14Lab1/sqlite} date

Tue Nov 19 18:07:25 CST 2024

{cslinux2:~/week14Lab1/sqlite} uname -a

Linux cslinux2.utdallas.edu 3.10.0-1160.119.1.el7.x86\_64 #1 SMP Tue Jun 4 14:43:51 UTC 2024 x86\_64 x86\_64 x86\_64 GNU/Linux

{cslinux2:~/week14Lab1/sqlite} exit

logout

───────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────

Session stopped

- Press <Return> to exit tab

- Press R to restart session

- Press S to save terminal output to file