**CSCI 4333.1**

8/26/2024

Toyu

**2.2 Simple schema and keys**

**Relations/tables (primary key underscored):**

Grade(grade, gradePoint)  
School(schoolCode, schoolName)  
Department(deptCode, deptName, schoolCode, numStaff)  
Faculty(facId, fname, lname, deptCode, rank)  
Course(courseId, rubric, number, title, credits)  
Class(classId, courseId, semester, year, facId, room)  
Student(StuId, fname, lname, major, minor, ach, advisor)  
Enroll(stuId, classId, grade, n\_alerts)

CREATE TABLE IF NOT EXISTS Student (

stuId INT NOT NULL,

fname VARCHAR(30) NOT NULL,

lname VARCHAR(30) NOT NULL,

major CHAR(4) NULL,

minor CHAR(4) NULL,

-- ach: accumulated credit hours, including transferred credits.

ach INTEGER(3) UNSIGNED NULL DEFAULT 0,

advisor INT NULL,

CONSTRAINT Student\_stuId\_pk PRIMARY KEY(stuId),

-- an artificial example of a CHECK constraint.

CONSTRAINT Student\_ach\_cc CHECK ((ach>=0) AND (ach < 250)),

CONSTRAINT Student\_major\_fk FOREIGN KEY (major)

REFERENCES Department(deptCode) ON DELETE CASCADE,

CONSTRAINT Student\_minor\_fk FOREIGN KEY (minor)

REFERENCES Department(deptCode) ON DELETE CASCADE,

CONSTRAINT Student\_advisor\_fk FOREIGN KEY (advisor)

REFERENCES Faculty(facId)

);

A primary key of a table uniquely identify a row.

**Foreign keys:**  
  
1. Student(advisor) references Faculty(facId)  
2. Student(major) references Department(deptCode)  
3. Student(minor) references Department(deptCode)  
4. Faculty(deptCode) references Department(deptCode)  
5. Department(schoolCode) references School(schoolCode)  
6. Enroll(stuId) references Student(stuId)  
7. Enroll(classId) references Class(classId)  
8. Enroll(grade) references Grade(grade)  
9. Class(courseId) references Course(courseId)  
10. Class(facId) references Faculty(facId)  
11. Course(Rubric) references Department(deptCode)

8/21/2024

HW #1 software

**3.1 MySQL Server Setup**

We will use MariaDB that is a part of XAMPP. Do not recommend installing standalone MySQL.

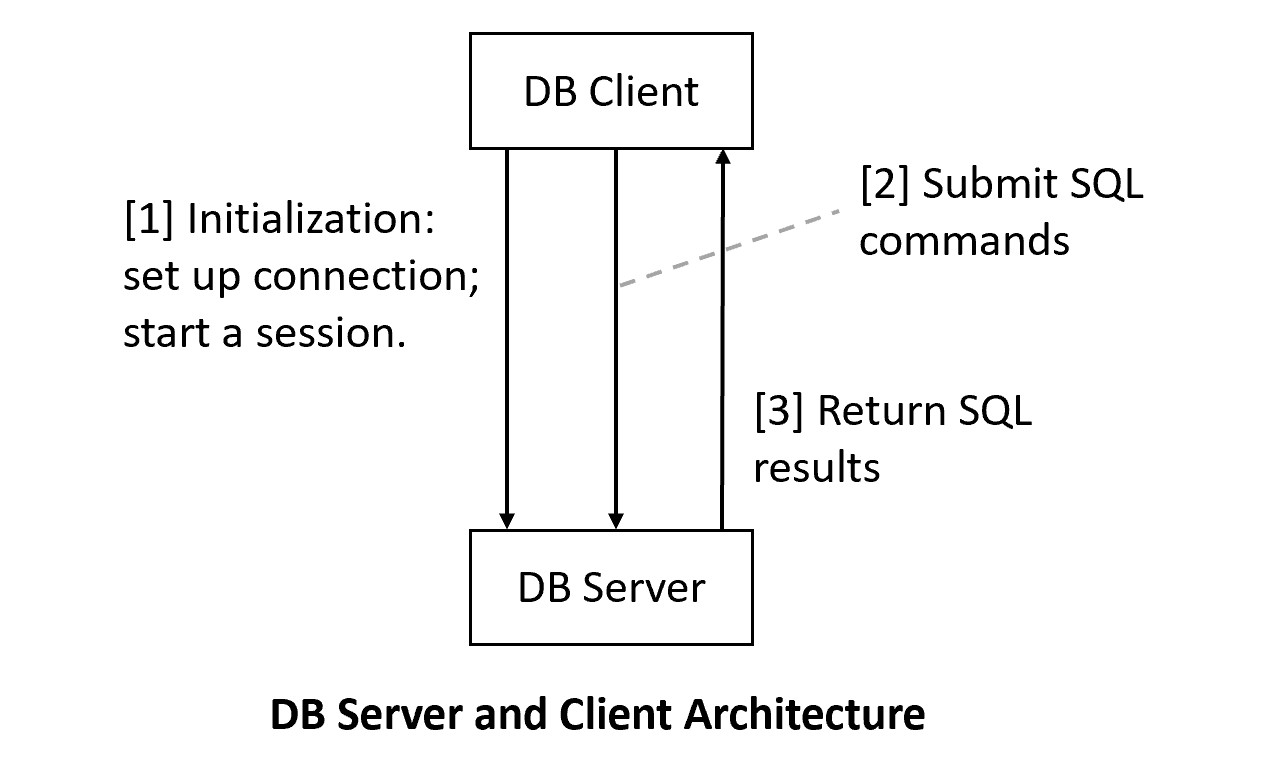
[1] Install XAMPP, which contains many server software configured to work together for development purposes. For XAMPP, we will use MySQL/Maria DB and Apache (Web server).

1. Recommended to install XAMPP in the*top* level: c:\xampp (likely the default).
2. Set up development accounts immediately using *phpMyAdmin* after installation.
3. Change the root password (optional but recommended): a secure step that requires tinkering.
4. To ensure that PHPMyAdmin will work on a new admin account (optional):
   1. Use PHPMyAdmin to create a new admin account, e.g., "frog\_ad", with the password "a\_new\_prince" for both hostname '%' and 'localhost'
   2. PhpMyAdmin uses the default root account (with no initial password) via localhost.
   3. Thus, you will need to supply the new username and password to start up PhpMyAdmin by editing the file c:\xampp\phpMyAdmin\config.inc.php, search change the line to, for example:
      1. $cfg['Servers'][$i]['user'] = 'frog\_ad';
      2. $cfg['Servers'][$i]['password'] = 'a\_new\_prince';

**3. MySQL**

* The standard query language for RDBMS is Structured Query Language (SQL).
* We use MySQL (or MariaDB) in this class.

DBMS mostly uses a client-server architecture.



DB Server: XAMPP

A screenshot of a computer

Description automatically generated

**3.1 MySQL Server Setup**

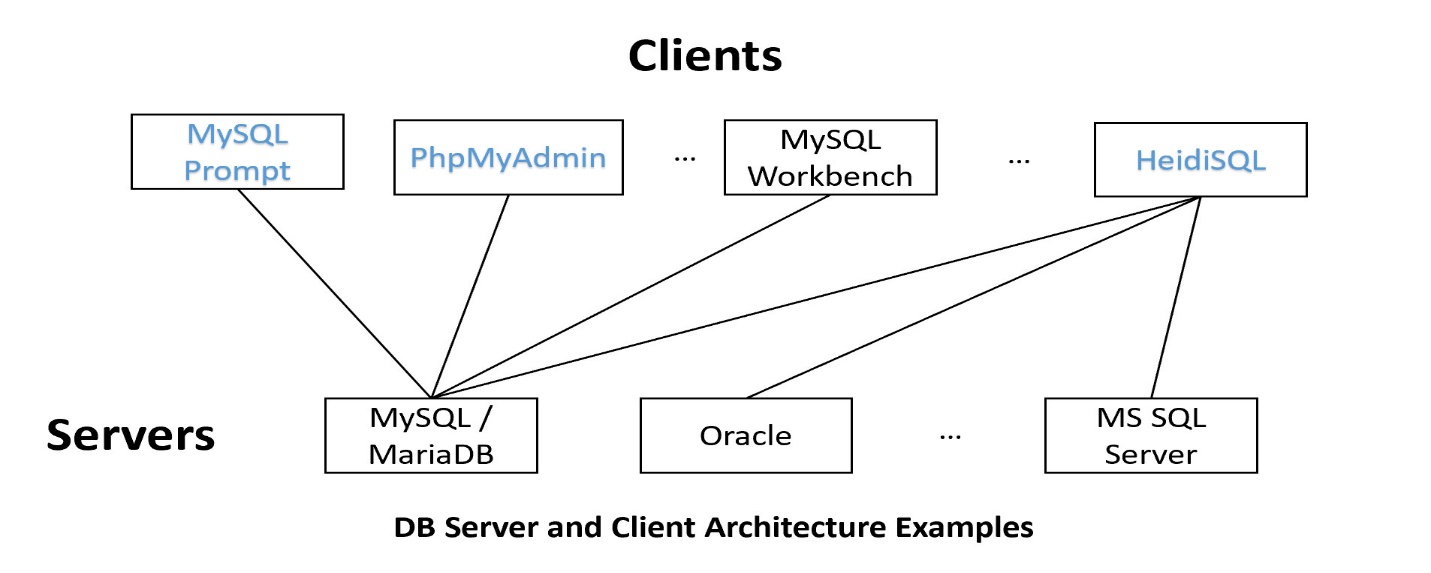
We will use MariaDB that is a part of XAMPP. Do not recommend installing standalone MySQL.

[1] Install XAMPP, which contains many server software configured to work together for development purposes. For XAMPP, we will use MySQL/Maria DB and Apache (Web server).

1. Recommended to install XAMPP in the*top* level: c:\xampp (likely the default).
2. Set up development accounts immediately using *phpMyAdmin* after installation.
3. Change the root password (optional but recommended): a secure step that requires tinkering.
4. To ensure that PHPMyAdmin will work on a new admin account (optional):
   1. Use PHPMyAdmin to create a new admin account, e.g., "frog\_ad", with the password "a\_new\_prince" for both hostname '%' and 'localhost'
   2. PhpMyAdmin uses the default root account (with no initial password) via localhost.
   3. Thus, you will need to supply the new username and password to start up PhpMyAdmin by editing the file c:\xampp\phpMyAdmin\config.inc.php, search change the line to, for example:
      1. $cfg['Servers'][$i]['user'] = 'frog\_ad';
      2. $cfg['Servers'][$i]['password'] = 'a\_new\_prince';

**3.2 MySQL Clients Setup:**

It is common to use multiple clients to connect to a backend database server. In this course, we will use three clients in our classes. You may use your own favorite clients (e.g., MySQL Workbench). However, I may not be as helpful in these clients.



MySQL prompt: C:\xampp\mysql\bin\mysql.exe

[1] MySQL Command-Line Prompt: will be used in this class.

1. Come with (1) XAMPP/MariaDB or (2) MySQL 8.x. (Note that the two versions of mysql prompt are somewhat different.)
   1. MariaDB mysql: <https://mariadb.com/kb/en/mysql-command-line-client/>
   2. MySQL 8.x mysql: <https://dev.mysql.com/doc/refman/8.0/en/mysql.html>
2. A command line text-based MySQL-specific client.
3. You may set the PATH variable so you can call mysql prompt anywhere, such as by adding "c:\xampp\mysql\bin" in the PATH system environment variable.

***Example:***

**mysql –h *host* -u *user* -p**

or

**mysql –h *host* -u *user* -p -P port\_number**

[2] PhPMyAdmin

1. A Web-based GUI client focused on DB administration.
2. After starting both MySQL and Apache in XAMPP, go to localhost in your browser.
3. MySQL specific.

Start Apache

A screenshot of a computer

Description automatically generated

Create a ‘s1’ account

A screenshot of a computer

Description automatically generated

[3] HeidiSQL: will be used in this class

1. A general Windows GUI SQL client

A screenshot of a computer

Description automatically generated

[4] MySQL Workbench:

1. A GUI MySQL client that comes with MySQL 8.x (but not XAMPP)