

CS6400 Phase 3 Project Resources:

Revised: 6/27/2017

Language Choices: (not exhaustive)

- **Web Based:** HTML, PHP, JS, CSS
 - **PHP: v7 preferred**
 - <http://www.w3schools.com/php/default.asp>
 - <http://php.net/manual/en/language.variables.basics.php>
 - **HTML5:** content of webpage
 - <http://www.w3schools.com/html/default.asp>
 - **JavaScript/JQuery (optional):** dynamic content changes
 - <http://www.w3schools.com/js/default.asp>
 - <https://www.tutorialspoint.com/jquery/>
 - **CSS3 (optional):** Design of webpage
 - <http://www.w3schools.com/css/default.asp>
 - **AJAX (optional):** Asynchronous JavaScript and XML
 - <http://api.jquery.com/jquery.ajax/>
 - **Bootstrap (optional):** open-source front-end framework (helps with standardization)
 - <http://www.w3schools.com/bootstrap/default.asp>
 - <https://v4-alpha.getbootstrap.com/components/forms/>
 - **Ruby on Rails:** <https://scotch.io/tutorials/a-crash-course-in-ruby>
- **Python Flask:**
 - <https://code.tutsplus.com/tutorials/creating-a-web-app-from-scratch-using-python-flask-and-mysql--cms-22972>
 - <https://scotch.io/tutorials/build-a-crud-web-app-with-python-and-flask-part-one>
 - https://www.tutorialspoint.com/python/python_database_access.htm
- **Java GUI:**
 - <https://docs.oracle.com/javase/tutorial/jdbc/basics/jdbcswing.html>
 - <http://www.developer.com/java/creating-a-jdbc-gui-application.html>
 - <http://www.javaguicodexample.com/javadesktopguimysql1.html>

Free Student Software/Upgrade Options:

<https://e5.onthehub.com/>
<http://software.oit.gatech.edu/>
<https://www.jetbrains.com/student/>
<https://education.github.com/pack>

Integrated Development Environments (IDEs):

<https://www.jetbrains.com/phpstorm/>
<https://netbeans.org/features/php/>
<https://www.jetbrains.com/pycharm/>
<https://www.jetbrains.com/datagrip/>

Note: Known Issue with **PHPStorm \$_POST always empty**: Fix- using the built in server instead.

<http://stackoverflow.com/questions/35290133/phpstorm-post-always-empty>
<https://intellij-support.jetbrains.com/hc/en-us/community/posts/206999125-PhpStorm-10-does-not-allow-POST-method>

Relational SQL Database- Language Choices: (Do NOT use SQLite) https://en.wikibooks.org/wiki/Converting_MySQL_to_PostgreSQL

MySQL <https://www.mysql.com/>

PostgreSQL: <https://www.postgresql.org/>

MS SQL Server: <https://www.microsoft.com/en-us/sql-server/sql-server-2016>

MariaDB: <https://mariadb.org/>

Front End: Client Side User Interface

Back End: Server Side with Database (PHP, SQL, Ruby, etc.)

Full Stack: Both Server and Client Side

<http://blog.udacity.com/2014/12/front-end-vs-back-end-vs-full-stack-web-developers.html>

Solution Stacks: set of software components needed for web development- <https://bitnami.com/stacks/infrastructure>

LAMP: Linux, Apache, MySQL, PHP

WAMP: Windows, Apache, MySQL, PHP

MAMP: Mac OS, Apache, MySQL, PHP

VM Options:

<https://www.vagrantup.com/>

<https://confluence.jetbrains.com/display/PhpStorm/Configuring+a+Vagrant+VM+for+Debugging>

SQL Data Types- follow “Best Practices”:

<http://searchsqlserver.techtarget.com/tip/Creating-SQL-Server-columns-A-best-practices-guide>

<http://stackoverflow.com/questions/7167604/how-accurately-should-i-store-latitude-and-longitude>

<http://www.agiledata.org/essays/keys.html>

<https://dev.mysql.com/doc/refman/5.7/en/data-types.html>

SQL Naming Conventions:

Entity/Table Names (singular noun): **UpperCamelCase** Example: REGULARUSER (note: without the ‘s’ singular)

Attribute/Column Names: **lowercase_underscore** Example: username

Primary **Surrogate Key (phase3 only)**: tableNameID Example: userID unsigned NOT NULL AUTO_INCREMENT (‘ID’ is capitalized)

Foreign Keys: FK_ChildTable_childColumn_ParentTable_parentColumn

<http://stackoverflow.com/questions/29561123/identify-parent-and-child-table-in-a-database>

ParentTable= User

ChildTable = AdminUser

fk_ChildTable_childColumn_ParentTable_parentColumn

FK Result: fk_AdminUser_email_User_email

Note: there are *many* conventions out there:

<http://www.vertabelo.com/blog/technical-articles/naming-conventions-in-database-modeling>

<https://launchbylunch.com/posts/2014/Feb/16/sql-naming-conventions/>

SQL Centric Characters:

Grave Accent ` = U+60 (good escape character)

<http://stackoverflow.com/questions/7857278/what-is-the-meaning-of-grave-accent-aka-backtick-quoted-characters-in-mysql>

Apostrophe ' = U+27 (good for \$param application variables)

<https://unicode-table.com/en/search/?q=%27>

Left Single Quote ‘ = U+2018 (**BAD**: queries will fail)

Right Single Quote ’ = U+2019 (**BAD**: queries will fail)

MySQL Procedural vs. Object Oriented

<http://php.net/manual/en/mysqli.quickstart.dual-interface.php>

<http://stackoverflow.com/questions/1530868/simple-explanation-php-oop-vs-procedural>

<https://www.binpress.com/tutorial/using-php-with-mysql-the-right-way/17>

PHP: Include vs. Require:

`include('folder/script.php');`

`include_once('folder/script.php)`

`require('folder/script.php)`

`require_once('folder/script.php)`

<http://www.techrepublic.com/article/10-tips-for-php-scripts-reuse-code-with-the-include-and-require-functions/>

PHP Data Objects: Error Handling:

<http://php.net/manual/en/pdo.error-handling.php>

<http://php.net/manual/en/debugger.php>

PHP Cookies (client side) vs. Session variables (server side):

<http://php.net/manual/en/book.session.php>

<http://stackoverflow.com/questions/6253633/cookie-vs-session>

Form Validation: options

http://www.w3schools.com/php/php_form_validation.asp

<https://jqueryvalidation.org/>

<https://v4-alpha.getbootstrap.com/components/forms/>

<https://docs.djangoproject.com/en/1.10/ref/forms/validation/>

<http://flask.pocoo.org/docs/0.12/patterns/wtforms/>

Text Encoding: source files and database settings:

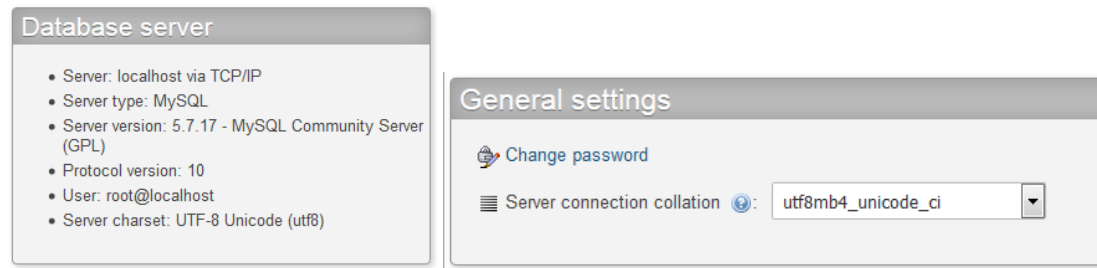
Check your text encoding on source .php files, phpmyAdmin settings, and db script.

<http://stackoverflow.com/questions/4777900/how-to-display-utf-8-characters-in-phpmyadmin>

<http://stackoverflow.com/questions/275411/php-output-showing-little-black-diamonds-with-a-question-mark/15138120>

<http://stackoverflow.com/questions/367711/what-is-the-best-collation-to-use-for-mysql-with-php>

<http://php.net/manual/en/mysqli.set-charset.php>



'_ci' refers to case insensitive



Notepad++ text editor: bottom right shows the file encoding (UTF-8)

Web Development/Debugging:

<https://developer.mozilla.org/en-US/docs/Tools>

<https://developers.google.com/web/tools/chrome-devtools/?hl=en>

<https://www.jetbrains.com/help/phpstorm/2016.2/php-debugging-session.html>

http://www.w3schools.com/js/js_debugging.asp

<https://www.keycdn.com/blog/web-development-tools/>

Security Extras: (not required)

Password Hashing:

```
string password_hash ( string $password , integer $algo [, array $options ] )
```

<http://php.net/manual/en/faq.passwords.php>

<http://php.net/manual/en/ref.password.php>

SQL Injection Prevention: `mysqli_real_escape_string()`

```
string mysqli_real_escape_string ( mysqli $link , string $escapestr )
```

<http://stackoverflow.com/questions/5741187/sql-injection-that-gets-around-mysql-real-escape-string/12118602#12118602>

https://www.owasp.org/index.php/PHP_Security_Cheat_Sheet

Additional Database Resources:

<http://www.cs.montana.edu/~halla/csci440/index.html>

https://lagunita.stanford.edu/courses/Engineering/db/2014_1/about

<http://web.stanford.edu/class/cs145/>