



CSC 436 Assignment

Populating & Querying Your Database

Overview

In this assignment, you will navigate the full lifecycle of database management, starting with the setup and configuration of a new database in cPanel. With a focus on data integrity, you will learn to create tables with primary key, not null, unique, check, and foreign key constraints. Once your database is populated, you will dive into querying, mastering the art of retrieving, filtering, and aggregating data using SQL commands and operators. Finally, you will document your journey, detailing the steps taken to ensure data accuracy and consistency, and showcasing your database's structure, population, and insights extracted through SQL queries. Through this comprehensive experience, you will develop a robust understanding of database management principles and practices.

Part III: Querying Your Database

- **Single-Relation Queries**

- Use the **SELECT** clause to retrieve specific columns or all columns from a single table.
- Use the **FROM** clause to specify the table from which you want to retrieve data.
- Apply the **WHERE** clause with comparison and logical operators to filter rows based on specific conditions.
- Employ arithmetic operators (+, -, *, /) to perform calculations within your queries.
- Use the **ORDER BY** clause to sort query results based on specified column(s).

- **Multiple-Relation Queries**

- Perform joins (i.e., **INNER JOIN**, **LEFT JOIN**, **RIGHT JOIN**, **FULL OUTER JOIN**) to combine data from multiple tables based on related columns.
- Use the **NATURAL JOIN** to join tables based on columns with the same name.
- Apply aggregate functions (i.e., **SUM**, **AVG**, **COUNT**, **MIN**, **MAX**) to calculate summary statistics across groups of data.
- Use the **GROUP BY** clause to group query results based on one or more columns.
- Apply aggregate functions alongside **GROUP BY** to perform group-level calculations.

- Use the **HAVING** clause to filter group-level results based on aggregate function results.
- **Executing Queries**
 - Write SQL queries that leverage the above-mentioned commands and operators to extract meaningful insights from your database.
 - Test your queries in a SQL environment.
 - Verify the accuracy and relevance of query results by cross-referencing them with the expected outcomes.

Part IV: Documentation

You will detail the steps taken to ensure data accuracy, demonstrate database population, and showcase meaningful insights extracted from the database.

1. Describe how you ensured that each table in your database includes appropriate integrity constraints to enforce data accuracy and consistency, and how you performed data validation checks when inserting data. Include details on:
 - Implementing primary key constraints to uniquely identify each record.
 - Applying not null constraints to essential attributes.
 - Enforcing unique constraints to ensure attribute distinctiveness.
 - Setting check constraints to enforce specific conditions on attribute values.
 - Maintaining referential integrity through foreign key constraints.

Your description should outline the steps taken during the table creation process to ensure data integrity and accuracy, as well as the validation checks performed during data insertion.

The tables that represent the main entity sets in our ER diagram—*Characters*, *Locations*, *Props*, and *Users*—ensure data integrity through an ID primary key attribute. This allows multiple characters with identical names, races, and other identifying attributes to exist without issue, which will happen when there are characters re-used across multiple worlds. The tables representing lesser entity sets and relationship sets—*Contains*, *Creatures*, *NPCs*, *Players*, and *PlayingIn*—instead rely on foreign keys to ensure data integrity. Since the foreign keys are exclusively tied to the primary keys of other tables, there is not any risk of records that are not uniquely identifiable. Employing so many foreign key constraints also assists with data consistency because any change that would create an inconsistency would violate one of the foreign key constraints in place.

A majority of attributes are provided with a not null constraint because many of the attributes are crucial to playing the game on the user's end. For example, the *Prop* table has the

attributes *quantity* and *owner* to represent how many of an item there is and who owns it. The not null constraint is applied to *quantity* because it is important to know the quantity of an object, there can't be an undefined number of gold coins. However, the *owner* attribute can be null because not every item is owned by someone.

Since a majority of our data are simple strings or numbers, most of the checks will be on a user's permissions, rather than the data they are trying to input. For example, the *Characters* table includes the attributes *description*, *gmNotes*, and *partyNotes*. All of these are strings describing a given character, but who can access and edit them is different between each attribute. The *description* can be read by both the game master and players, but only the game master can edit it; *gmNotes* can be read and edited only by the game master; and *partNotes* can be read by both the game master and players, but only the players can edit it. Therefore, as long as users are inputting proper strings, the validation checks will be primarily concerned with the user's credentials, rather than what they are inputting.

2. Include screenshots of EACH table in your database. You can use the **Browse** tab and take a screenshot of the data that is displayed. This will demonstrate that you have successfully populated the database.

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> Characters	Browse Structure Search Insert Empty Drop	10	InnoDB	utf8_unicode_ci	32.0 KiB	-
<input type="checkbox"/> Contains	Browse Structure Search Insert Empty Drop	6	InnoDB	utf8_unicode_ci	48.0 KiB	-
<input type="checkbox"/> Creatures	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8_unicode_ci	32.0 KiB	-
<input type="checkbox"/> Locations	Browse Structure Search Insert Empty Drop	8	InnoDB	utf8_unicode_ci	16.0 KiB	-
<input type="checkbox"/> NPCs	Browse Structure Search Insert Empty Drop	5	InnoDB	utf8_unicode_ci	32.0 KiB	-
<input type="checkbox"/> Players	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8_unicode_ci	48.0 KiB	-
<input type="checkbox"/> PlayingIn	Browse Structure Search Insert Empty Drop	5	InnoDB	utf8_unicode_ci	64.0 KiB	-
<input type="checkbox"/> Props	Browse Structure Search Insert Empty Drop	5	InnoDB	utf8_unicode_ci	48.0 KiB	-
<input type="checkbox"/> Users	Browse Structure Search Insert Empty Drop	5	innodb	utf8_unicode_ci	16.0 KiB	-
9 tables	Sum	49	InnoDB	utf8_unicode_ci	336.0 KiB	0 B

Showing rows 0 - 9 (10 total, Query took 0.0001 seconds.)

SELECT * FROM `Characters`

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	ID	isAt	name	race	description	gmNotes	partyNotes
<input type="checkbox"/> Edit Copy Delete	1	2	Wilhelm Riquet	human	thief	Passion for animals	Played by Steve
<input type="checkbox"/> Edit Copy Delete	2	4	Commodore Stephahk	human	Recruitment officer at Fort Hranic	Rat who let attackers into fort	Weirdly chill guy
<input type="checkbox"/> Edit Copy Delete	3	4	Renly Gokel	human	Recruit specializing in alchemy	Harmless?	He is an evil terrorist
<input type="checkbox"/> Edit Copy Delete	4	2	Horacio Garzon	human	wine artisan	Passion for grapes	Played by George
<input type="checkbox"/> Edit Copy Delete	5	2	Jovi	Ka'Tavin	Animal sold in Siwanilua	Will be used to train 'Ride' skill	Bought by Wilhelm
<input type="checkbox"/> Edit Copy Delete	6	4	Olver Thumbless	human	Recruit who has no thumbs	Dies in Hranic Raid	His name is now 'Nubs'
<input type="checkbox"/> Edit Copy Delete	7	5	Xiarkdyoth	spider	Spider in Myrantahl Forests	Beast Aliyra encounters?	
<input type="checkbox"/> Edit Copy Delete	8	2	Leon Septar	human	Bouncer, detective	Passion for sneaking	Played by Henry
<input type="checkbox"/> Edit Copy Delete	9	5	Aliyra Maastehr	ghord	Apothecary from Ghordeiol	Wife to Ephram and Mother to Obram and Ilen	
<input type="checkbox"/> Edit Copy Delete	10	4	Chef Mya	human	Chef at Fort Hranic	Head chef	Falls in love with Wilhelm

✓ Showing rows 0 - 5 (6 total, Query took 0.0001 seconds.)

SELECT * FROM `Contains`

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 | Filter rows: Sort by key: None

Extra options

container	containee
6	1
6	2
6	3
6	4
6	5
4	7

✓ Showing rows 0 - 1 (2 total, Query took 0.0002 seconds.)

SELECT * FROM `Creatures`

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 | Filter rows: Sort by key: None

Extra options

ID	population
5	1
7	1

✓ Showing rows 0 - 7 (8 total, Query took 0.0002 seconds.)

SELECT * FROM `Locations`

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 | Filter rows: Sort by key: None

Extra options

	ID	name	description	gmNotes	partyNotes
<input type="checkbox"/> Edit Copy Delete	1	Kua Loranta	Desert nation ruled by the Kua Lorantene peoples	Under control of the foreign power Hrace	People have venomous spikes
<input type="checkbox"/> Edit Copy Delete	2	Siwanilua	Wealthy city ruled by Queen Yttrilyna	Hrace has soft control over the Queen	The royalty are rich but the people are poor
<input type="checkbox"/> Edit Copy Delete	3	Hanging Gardens of Azkenilua	Wonder built to the Goddess Azkenilua	Pilgrimage site	They do not grow grapes here
<input type="checkbox"/> Edit Copy Delete	4	Hranic Fort	Military island off the coast of Kua Loranta	Session start. Raided by Kua Lorantene attackers	Renly is here and he must be dangerous...
<input type="checkbox"/> Edit Copy Delete	5	Ghordeioli	Northern country also soft controlled by Hrace	Potential site for future campaigns	
<input type="checkbox"/> Edit Copy Delete	6	The World	This is the big world that contains everything	The good and evil happens here	
<input type="checkbox"/> Edit Copy Delete	7	Barracks	Barracks in a fort, weapons and stuff		
<input type="checkbox"/> Edit Copy Delete	8	Completely Unrelated World	This world doesn't have anything yet		

✓ Showing rows 0 - 4 (5 total, Query took 0.0001 seconds.)

SELECT * FROM `NPCs`

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

ID opinions

- 2 Cooperative with Hrace, indifferent to recruits
- 3 Friendly facade, secretly plotting destruction
- 6 Eager to prove himself despite disability
- 9 Caring and knowledgeable, misses her family
- 10 Warm and welcoming, develops feelings for Wilhelm

✓ Showing rows 0 - 2 (3 total, Query took 0.0001 seconds.)

SELECT * FROM `Players`

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

ID level playedBy

- 1 3 Steve
- 4 3 George
- 8 3 Henry

✓ Showing rows 0 - 4 (5 total, Query took 0.0002 seconds.)

SELECT * FROM `PlayingIn`

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

user	plays	world	role
Steve	1	6	player
George	4	6	player
Henry	8	6	player
George	NULL	8	gm
GM Individual	NULL	6	gm

Showing rows 0 - 4 (5 total, Query took 0.0001 seconds.)

SELECT * FROM `Props`

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	ID	isIn	name	description	gmNotes	partyNotes	itemType	rarity	quantity	owner
<input type="checkbox"/> Edit Copy Delete	1	2	Venom Spike	A spike harvested from a Kua Lorantene warrior	Can be used as a poison weapon	Found in Siwanilua market	Weapon	Uncommon	3	1
<input type="checkbox"/> Edit Copy Delete	2	4	Alchemy Kit	A set of tools for brewing potions	Belongs to Renly	Confiscated after the raid	Tool	Common	1	3
<input type="checkbox"/> Edit Copy Delete	3	2	Grape Wine	A fine bottle of Siwaniluan wine	Horacio made this himself	Worth good money	Consumable	Common	5	4
<input type="checkbox"/> Edit Copy Delete	4	4	Fort Manifest	A list of all recruits at Hranic Fort	Contains evidence against Stephahk	We need to get this	Document	Rare	1	2
<input type="checkbox"/> Edit Copy Delete	5	1	Desert Cloak	A cloak that blends into sand	Useful for desert travel	Bought in Kua Loranta	Armor	Common	1	8

Showing rows 0 - 4 (5 total, Query took 0.0002 seconds.)

SELECT * FROM `Users`

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	username	password
<input type="checkbox"/> Edit Copy Delete	George	password456
<input type="checkbox"/> Edit Copy Delete	GM Individual	password000
<input type="checkbox"/> Edit Copy Delete	Henry	password789
<input type="checkbox"/> Edit Copy Delete	No Friends Larry	password000
<input type="checkbox"/> Edit Copy Delete	Steve	password123

- Five SQL queries were written to extract meaningful insights from the database. Each query serves a distinct purpose and uses different SQL clauses and operators including:
 - A query leveraging SELECT, FROM, and WHERE clauses with comparison and logical operators to filter rows based on specific conditions.

This query will select all characters in the game that are humans. The expected result should display Wilhelm, Stephahk, Renly, Horacio, Olver, Leon, and Mya. This could be used for creating a plot element that only applies to humans, for example there could be a city where humans are discriminated against, and it would be helpful to easily know what characters are humans.

`SELECT * FROM `Characters` WHERE race = 'human';`

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	ID	isAt	name	race	description	gmNotes	partyNotes
<input type="checkbox"/> Edit Copy Delete	1	2	Wilhelm Riquet	human	thief	Passion for animals	Played by Steve
<input type="checkbox"/> Edit Copy Delete	2	4	Commodore Stephahk	human	Recruitment officer at Fort Hranic	Rat who let attackers into fort	Weirdly chill guy
<input type="checkbox"/> Edit Copy Delete	3	4	Renly Gokel	human	Recruit specializing in alchemy	Harmless?	He is an evil terrorist
<input type="checkbox"/> Edit Copy Delete	4	2	Horacio Garzon	human	wine artisan	Passion for grapes	Played by George
<input type="checkbox"/> Edit Copy Delete	6	4	Olver Thumbless	human	Recruit who has no thumbs	Dies in Hranic Raid	His name is now 'Nubs'
<input type="checkbox"/> Edit Copy Delete	8	2	Leon Septar	human	Bouncer, detective	Passion for sneaking	Played by Henry
<input type="checkbox"/> Edit Copy Delete	10	4	Chef Mya	human	Chef at Fort Hranic	Head chef	Falls in love with Wilhelm

☐ Check all | With selected: Edit Copy Delete Export

- A query employing arithmetic operators (+, -, *, /) to perform calculations within the queries.

This query will find all items in the database with a quantity over 3. This should return only the Venom Spike and Grape Wine items. This could be used to balance game item rewards, using this knowledge to avoid rewarding players with items that are already common in the world.

✓ Showing rows 0 - 1 (2 total, Query took 0.0001 seconds.)

`SELECT name, quantity FROM `Props` WHERE quantity >= 3;`

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: N

Extra options

name	quantity
Venom Spike	3
Grape Wine	5

- A query performing join expressions to combine data from multiple tables.

This query should return every character's name and location. This should return Wilhelm at Siwanilua, Stephahk at Hranic Fort, and so on. This can be used to easily track the locations of characters in the game, which becomes very difficult in a game with many moving pieces.

✓ Showing rows 0 - 9 (10 total, Query took 0.0002 seconds.)

```
SELECT Characters.name, Locations.name FROM Characters JOIN Locations ON Characters.isAt = Locations.ID;
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

name	name
Wilhelm Riquet	Siwanilua
Commodore Stephahk	Hranic Fort
Renly Gokel	Hranic Fort
Horacio Garzon	Siwanilua
Jovi	Siwanilua
Olver Thumbless	Hranic Fort
Xiarkydoth	Ghordeiol
Leon Septar	Siwanilua
Aliyra Maastehr	Ghordeiol
Chef Mya	Hranic Fort

- A query using the GROUP BY clause and applying aggregate functions to perform group-level calculations.

This query should return the number of characters at each location in the game. This should return 4 characters at Siwanilua, 4 characters at Hranic Fort and 2 characters at Ghordeiol. This can be used to determine what locations to focus on when preparing for a game session, as it's more important to prepare the locations players are currently present in.

✓ Showing rows 0 - 2 (3 total, Query took 0.0003 seconds.)

```
SELECT Locations.name, COUNT(Characters.ID) FROM Locations JOIN Characters ON Locations.ID = Characters.isAt GROUP BY Locations.name ORDER BY COUNT(Characters.ID);
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table

Extra options

name	COUNT(Characters.ID)
Ghordeiol	2
Siwanilua	4
Hranic Fort	4

☐ Show all | Number of rows: 25 | Filter rows: Search this table

Query results operations

Print Copy to clipboard Export Display chart Create view

This query should display the average level of the party's characters. This should return 3, as all the PCs currently in the database are at level 3. This will be used to determine how powerful the party is in order to balance monster encounters.

The screenshot displays a database query interface. At the top, a green status bar indicates "Showing rows 0 - 0 (1 total, Query took 0.0001 seconds.)". Below this, the SQL query `SELECT AVG(level) FROM Players;` is shown. A row of controls includes a checkbox for "Profiling" and links for "Edit inline", "Edit", "Explain SQL", "Create PHP code", and "Refresh". Below the query, a control bar shows "Show all", "Number of rows: 25", and a "Filter rows" search box. An "Extra options" button is also present. The query result is displayed as a table with one row:

AVG(level)
3.0000

. Below the table, another control bar is visible. At the bottom, a "Query results operations" bar contains buttons for "Print", "Copy to clipboard", "Export", and "Display chart".

Each SQL query is documented with its intended purpose, expected results, and the actual SQL query written.

4. The results of each SQL query are included showcasing the insights extracted from the database. Screenshots of the tables displaying the query results are provided.

Submission

When you're finished, complete the following steps to submit your work:

- ☐ Export your document with responses as a **PDF file AND save it inside your documentation** folder. Refer to the following for documentation on how to do this:
 - [Google Docs](#) (File → Download → PDF Document)

- [Microsoft Word](#) (*File → Save As / Export → PDF*)
 - [Pages](#) (*File → Export To → PDF*)
- ☐ Inside your repo, create a folder named **db** and export your database as an **SQL file** into that folder. The file should include all SQL code used to create tables, insert data, and run other queries. Follow the steps in the [Exporting A Database section](#) to export your database into a single SQL file.
- ☐ All group members should then be able to pull this SQL file from the repo and upload it to their own cPanel accounts (see [Importing A Database section](#)).
- ☐ Upload all your changes to GitHub.
- ☐ **If you're using GitHub Desktop (GUI)**, complete the [Uploading Changes \(GitHub Desktop\) section](#) to upload your changes from your local device to GitHub.
- ☐ **If you're using Git (CLI)**, complete the [Uploading Changes \(GitHub CLI\) section](#) to upload your changes from your local device to GitHub.

ONE group member must paste the URL of your GitHub repository in the provided textbox in Brightspace. Click the blue *Submit* button to successfully submit your work for this assignment.

Grading Rubric

You can refer to the **Populating & Querying Your Database grading rubric** given in Brightspace for this assignment to find details on how your submission will be graded.