Legend of Zelda MySQL Database Spec Guide

Erik Heaney

August 18, 2017

Oregon State University

Overview

This relational database stores data about the popular Nintendo-exclusive video game franchise, *The Legend of Zelda*. There are seventeen original *Legend of Zelda* games. Each game has a set of recurring characters, Link and Zelda, and a set of unique characters. The same is true about the games locations. The database is composed of four entities: games, consoles, characters, and locations. These entities and their attributes are defined below. Each entity has a unique ID.

Game and console data, including release dates and total sales, were gathered from their individual wikipedia pages.

Game: A game within the *Legend of Zelda* franchise. This does not include remakes or special editions (e.g. *Ocarina of Time 3DS*).

> **Title:** The full name of the game.
>
> **Release Date:** The date game was released. For early games, where games were released on a region-by-region basis, the earliest release region was selected (usually Japan).
>
> **Sales**: Sales in millions of copies.

Console: A Nintendo console. This does not include version upgrades or models (e.g. GameBoy Color, Nintendo DSi). Multiple games can be on a single system; likewise, multiple systems can have the same game.

> **Name:** The full name of the console.
>
> **Release Date:** The date game was released. For earlier consoles, where consoles were released on a region-by-region basis, the earliest release region was selected (usually Japan).
>
> **Sales:** Sales in millions.
>
> **Handheld:** A boolean value: 1 indicates handheld system and 0 indicates non-handheld. The Nintendo Switch is classified as a non-handheld.

Location: A location that appears within the *Legend of Zelda* franchise.

> **Name:** The name of the location.

Character: A character that appears within the *Legend of Zelda* franchise. Some characters appear in many games. A game has many characters.

> **Name:** Name of character.
>
> **Location:** The location where the character is from. A character may only be from one location.

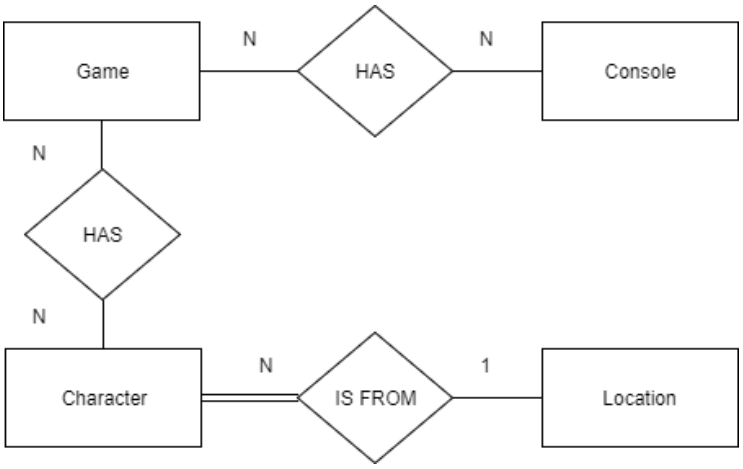
Relationships

Consoles and Games: Many-to-Many

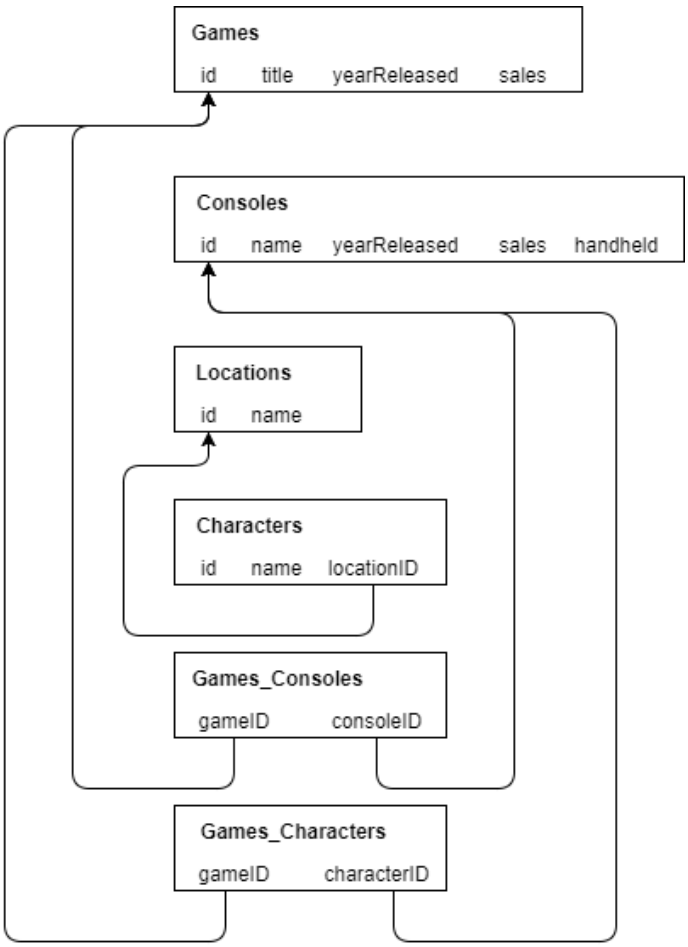
Games and Characters: Many-to-Many

Locations and Characters: Many-to-One-And-Only-One

ER Diagram



Schema



Data Definition Queries: CREATE

```
CREATE TABLE Games (
	id INT(11) AUTO_INCREMENT NOT NULL,
	title VARCHAR(255) NOT NULL,
	yearReleased DATE,
	sales FLOAT(11),
	PRIMARY KEY (id)
) ENGINE = InnoDB;

CREATE TABLE Consoles (
	id INT(11) AUTO_INCREMENT NOT NULL,
	name VARCHAR(255) NOT NULL,
	yearReleased DATE,
	sales FLOAT(11),
	handheld BOOLEAN,
	PRIMARY KEY (id)
) ENGINE = InnoDB;

CREATE TABLE Locations (
	id INT(11) AUTO_INCREMENT NOT NULL,
	name VARCHAR(255) NOT NULL,
	PRIMARY KEY (id),
) ENGINE = InnoDB;
```

```
CREATE TABLE Characters (
	id INT(11) AUTO_INCREMENT NOT NULL,
	locationID INT(11) NOT NULL,
	name VARCHAR(255) NOT NULL,
	FOREIGN KEY (locationID) REFERENCES Locations(id)
	ON DELETE CASCADE,
	PRIMARY KEY (id)
) ENGINE = InnoDB;

CREATE TABLE Games_Consoles (
	gameID INT(11),
	consoleID INT(11) NOT NULL,
	FOREIGN KEY (gameID) REFERENCES Games(id)
	ON DELETE CASCADE,
	FOREIGN KEY (consoleID) REFERENCES Consoles(id)
	ON DELETE CASCADE,
	PRIMARY KEY (gameID, consoleID)
) ENGINE = InnoDB;

CREATE TABLE Games_Characters (
	gameID INT(11) NOT NULL,
	characterID INT(11) NOT NULL,
	FOREIGN KEY (gameID) REFERENCES Games(id)
	ON DELETE CASCADE,
	FOREIGN KEY (characterID) REFERENCES Characters(id)
	ON DELETE CASCADE,
	PRIMARY KEY (gameID, characterID)
) ENGINE = InnoDB;
```

Data Definition Queries: INSERT

```
INSERT INTO Games (title, yearReleased, sales)
values
('The Legend of Zelda', '1986-02-21', 6.51), ('Zelda II: The Adventure of
Link', '1987-01-14', 4.38), ('The Legend of Zelda: A Link to the Past', '1991-
11-21', 4.61), ('The Legend of Zelda: Link\'s Awakening', '1993-06-06', 3.83),
('The Legend of Zelda: Ocarina of Time', '1998-11-21', 7.60), ('The Legend of
Zelda: Majora\'s Mask', '2000-04-27', 3.36), ('The Legend of Zelda: Oracle of
Seasons/Ages', '2001-02-27', 3.98), ('The Legend of Zelda: The Wind Waker',
'2003-12-13', 4.43),('The Legend of Zelda: Four Swords Adventures', '2004-12-
02', .81), ('The Legend of Zelda: Minish Cap', '2004-11-04', 1.76),('The Legend
of Zelda: Twilight Princess', '2006-11-19', 8.58),('The Legend of Zelda:
Phantom Hourglass', '2007-06-23', 4.76), ('The Legend of Zelda: Spirit Tracks',
'2009-12-07', 2.98), ('The Legend of Zelda: Skyward Sword', '2011-11-18',
3.67), ('The Legend of Zelda: A Link Between Worlds', '2013-11-22', 3.90),
('The Legend of Zelda: Tri Force Heroes', '2015-10-22', 1.14),('The Legend of
Zelda: Breath of the Wild', '2017-03-03', 3.88);

INSERT INTO Consoles (name, yearReleased, sales, handheld)
values
('Nintendo Entertainment System (NES)', '1983-07-15', 61.91, 0), ('Super
Nintendo Entertainment System (SNES)', '1990-11-21', 49.10, 0), ('Nintendo 64',
'1996-06-23', 32.93, 0), ('Nintendo GameCube', '2001-09-14', 21.74, 0),
('Nintendo Wii', '2006-10-20', 101.63, 0), ('Nintendo Wii U', '2012-11-18',
```

```
13.56, 0), ('Nintendo Switch', '2017-03-03', 4.7, 0), ('Game Boy', '1989-04-
21', 118.69, 1), ('Game Boy Advance', '2001-03-21', 81.51, 1), ('Nintendo DS',
'2004-11-21', 18.79, 1), ('Nintendo 3DS', '2011-02-26', 67.08, 1);

INSERT INTO Locations (name)
values
('Hyrule'), ('Goron City'), ('Zora''s Domain'),('Gerudo Desert'),('Kokiri
Forest'), ('Lon Lon Ranch');

INSERT INTO Characters (name, locationID)
values
('Princess Zelda', 1), ('Ganondorf', 1), ('Ganon', 1), ('Link', 5), ('King
Zora', 3), ('Princess Ruto', 3), ('Epona', 6), ('Deku Tree', 5), ('Talon', 6),
('Darunia', 2), ('Nabooru', 4);

INSERT INTO Games_Consoles (gameID, consoleID)
values
(1, 1), (2, 1), (3, 2), (4, 8), (5, 3), (6, 3), (7, 8), (8, 4), (9, 4), (10,
9), (11, 4), (11, 5), (12, 10), (13, 10), (14, 5), (15, 11), (16, 11), (17, 6),
(17, 7);

INSERT INTO Games_Characters (gameID, characterID)
values
(1, 1), (2, 1), (3, 1), (4, 1), (5, 1), (6, 1), (7, 1), (8, 1), (9, 1), (10,
1), (11, 1), (12, 1), (13, 1), (14, 1), (15, 1), (16, 1), (17, 1), (1, 4), (2,
4), (3, 4), (4, 4), (5, 4), (6, 4), (7, 4), (8, 4), (9, 4), (10, 4), (11, 4),
(12, 4), (13, 4), (14, 4), (15, 4), (16, 4), (17, 4), (1, 3), (2, 3), (3, 3),
(4, 3), (5, 3), (7, 3), (8, 3), (9, 3), (11, 3), (12, 3), (15, 3), (17, 3), (1,
2), (2, 2), (3, 2), (4, 2), (5, 2), (7, 2), (8, 2), (9, 2), (11, 2), (12, 2),
(15, 2), (17, 2), (3 ,5), (5 ,5),(7 ,5), (5, 6), (5, 7), (6, 7), (10, 7), (11,
7), (17, 7),(5, 8),(8, 8),(17, 8),(5, 9),(7, 9),(10, 9),(15, 9),(5, 10),(5,
11);
```

Data Manipulation Queries

Adding/Changing Relationships

```
INSERT INTO Games_Consoles SET consoleID=?, gameID=?

INSERT INTO Games_Characters SET gameID=?, characterID=?

UPDATE Characters SET locationID=? WHERE id=?
```

Removing Relationships

```
DELETE FROM Games_Characters WHERE (gameID=? AND characterID=?)

DELETE FROM Games_Consoles WHERE (consoleID=? and gameID=?)
```

Displaying Relationships

Games

```
SELECT id, title, DATE_FORMAT(yearReleased, \'%m-%d-%Y\') AS yearReleased,
sales FROM Games WHERE id=?


SELECT c.id, c.name, DATE_FORMAT(c.yearReleased, \'%m-%d-%Y\') AS yearReleased,
c.sales, c.handheld FROM Consoles c INNER JOIN Games_Consoles gc ON c.id =
gc.consoleID INNER JOIN Games g ON g.id = gc.gameID WHERE g.id=?

SELECT c.id, c.name FROM Characters c INNER JOIN Games_Characters gc ON c.id =
gc.characterID INNER JOIN Games g ON g.id = gc.gameID WHERE g.id=?


SELECT DISTINCT l.id, l.name FROM Locations l INNER JOIN Characters c ON l.id =
c.locationID INNER JOIN Games_Characters gc ON gc.characterID = c.id INNER JOIN
Games g ON g.id = gc.gameID WHERE g.id=?
```

Consoles

```
SELECT id, name, DATE_FORMAT(yearReleased, \'%m-%d-%Y\') AS yearReleased,
sales, handheld FROM Consoles WHERE id=?


SELECT g.id, g.title, DATE_FORMAT(g.yearReleased, \'%m-%d-%Y\') AS
yearReleased, g.sales FROM Games g INNER JOIN Games_Consoles gc ON g.id =
gc.gameID INNER JOIN Consoles c ON c.id = gc.consoleID WHERE c.id=?


SELECT DISTINCT ch.id, ch.name FROM Characters ch INNER JOIN Games_Characters
gch ON ch.id = gch.characterID INNER JOIN Games g ON g.id = gch.gameID INNER
JOIN Games_Consoles gc ON gc.gameID = g.id INNER JOIN Consoles c ON c.id =
gc.consoleID WHERE c.id=?


SELECT DISTINCT l.id, l.name FROM Locations l INNER JOIN Characters ch ON l.id
= ch.locationID INNER JOIN Games_Characters gch ON gch.characterID = ch.id
INNER JOIN Games g ON g.id = gch.gameID INNER JOIN Games_Consoles gc ON
gc.gameID = g.id INNER JOIN Consoles c ON c.id = gc.consoleID WHERE c.id=?
```

Characters

```
SELECT id, name FROM Characters WHERE id=?


SELECT DISTINCT g.id, g.title, DATE_FORMAT(g.yearReleased, \'%m-%d-%Y\') AS
yearReleased, g.sales FROM Games g INNER JOIN Games_Characters gc ON g.id =
gc.gameID INNER JOIN Characters c ON c.id = gc.characterID WHERE c.id=?


SELECT DISTINCT c.id, c.name, DATE_FORMAT(c.yearReleased, \'%m-%d-%Y\') AS
yearReleased, c.sales, c.handheld FROM Characters ch INNER JOIN
Games_Characters gch ON ch.id = gch.characterID INNER JOIN Games g ON g.id =
gch.gameID INNER JOIN Games_Consoles gc ON gc.gameID = g.id INNER JOIN Consoles
c ON c.id = gc.consoleID WHERE ch.id=?


SELECT DISTINCT l.id, l.name FROM Locations l INNER JOIN Characters ch ON l.id
= ch.locationID WHERE ch.id=?
```

Locations

```
SELECT id, name FROM Locations WHERE id=?
```

```
SELECT DISTINCT g.id, g.title, DATE_FORMAT(g.yearReleased, \'%m-%d-%Y\') AS
yearReleased, g.sales FROM Games g INNER JOIN Games_Characters gc ON g.id =
gc.gameID INNER JOIN Characters c ON c.id = gc.characterID INNER JOIN Locations
l ON l.id = c.locationID WHERE l.id=?


SELECT DISTINCT c.id, c.name, DATE_FORMAT(c.yearReleased, \'%m-%d-%Y\') AS
yearReleased, c.sales, c.handheld FROM Locations l INNER JOIN Characters ch ON
ch.locationID = l.id INNER JOIN Games_Characters gch ON ch.id = gch.characterID
INNER JOIN Games g ON g.id = gch.gameID INNER JOIN Games_Consoles gc ON
gc.gameID = g.id INNER JOIN Consoles c ON c.id = gc.consoleID WHERE l.id=?


SELECT DISTINCT c.id, c.name FROM Locations l INNER JOIN Characters c ON l.id =
c.locationID WHERE l.id=?
```

Entity Name Search

```
SELECT title FROM Games WHERE title LIKE %term%

SELECT name FROM Consoles WHERE name LIKE %term%

SELECT name FROM Locations WHERE name LIKE %term%

SELECT name FROM Characters WHERE name LIKE %term%
```

User Guide

The *Legend of Zelda* database has four sections:

1. **Adding a relationship:** A user may add a relationship between a game and a console or a game and a characters, as both of these pairs of entities possess a many-to-many relationship. A user may also change a character's home world, as a character possesses a one-and-only-one relationship with location.
2. **Searching an entity:** A user may search for a title or name of an entity string. The search result will return all entity instances that match the query with the entity type (e.g. game) and entity name.
3. **Displaying relationships:** A user may see which entities relate to each other, either directly or transitively. A user may also remove an entity from a relationship.
4. **Displaying all data:** A user may view all data in the tables at the bottom of the page. A user may also delete a table entry by clicking the delete button.