

Nicholas Bui

CS 340 - Summer 2018

Pokemon Database

Final Version

Website:

URL: <http://web.engr.oregonstate.edu/~bui/index.html>

Feedback by the peer reviewer:

“Hi Nicholas, Your html looks clean and accurate. Based on your outline your html follows all the entities and relationships you mentioned. I don’t know if you have already planning to make changes in the future but I would recommend to separate add, delete, update and view under a different link. There is too much info under the index html. I would make the tables under a separate link as well. I would also make the one large table into few smaller ones that represents the relationships more accurately. For your queries maybe include few more select functions for the tables that you will be creating. As mentioned above I would make few more tables that connect the info to each other. Otherwise everything looks great and this is an interesting choice for your project. It represents how they build the pokemon game few years ago. Good work!!”

“Hi Nicholas, The site looks great. Very intuitive and straight forward. The only feedback I have is with respect to your update query. I think it needs a where clause, so you only update the row in question.”

“The schema Pokemon_Ability foreign keys does not match the sql foreign keys. In the schema the pokemon ID attribute references the availably id attribute of the availably entity and the availably id references the pokemone id attribute of the pokemone entity. In the sql the foreign key pid references the pokemon_id attribute of the Pokemon entity and the aid references the ability_id attribute of the ability entity. I believe the error is in the schema and that the errors showing the forging keys are mixed up.”

Actions based on the feedback:

I decided to simplify some of my queries by combining them with other table. I had a lot of missing queries from my previous draft as well. I ended up having to add many more select

queries in order to fulfill the requirements. I also changed my update and delete queries because they were referencing the wrong tables. Fixed some typos in the HTML as well, and cleaned up some of the unnecessary tags.

This was a good catch by the peer reviewer. I messed up assigning the foreign key references for the ability IDs. I went ahead and fixed it in the SQL file so that it matches the schema. I also fixed the schema so that "Pokemon_Ability" was referencing the correct attributes.

Upgrades to the Draft version:

The biggest update was from pruning a lot of the HTML page. Following the pruning, I also had to add quite a few more queries. I had a lot of missing SELECT queries, so the HTML had to be tweaked. Overall, the main changes were in adding more queries and tidying up the HTML page.

Project Outline

Pokemon have been a huge phenomenon for the past two decades. Pokemon stands for "Pocket Monsters" and was created in Japan at first. I will be creating a database that describes the different Pokemon by using their unique characteristics. There are many different Pokemon that have different abilities and types. A database would be a very effective way of organizing all of this information. Since there are almost 1000 Pokemon out today, I will only be focusing on the original 151 Pokemon that came out first.

Database Outline

The entities in my database are: Pokemon, gym leaders, abilities, towns, and types.

Table Descriptions:

The Pokemon table contains rows of unique Pokemon. Each row has the following attributes: ID number, name, and species name. The ID used to identify the Pokemon does not have to be auto incrementing because each Pokemon (1 through 151) will have a unique ID regardless. The primary key for this table is the ID number.

There are many different types of Pokemon. A Pokemon can be a minimum of 1 type and a maximum of 2 different types. The type table will have a type ID and type name as its two attributes. The type ID will be the primary key.

There are 8 gym leaders in the Pokemon world. Each gym leader has unique abilities as well as a favored "type" of Pokemon. Gym leaders reside in large towns and there can only be one gym leader per town. Each gym leader has an ID (auto-incrementing), a gym leader name, and a

type. The gym leader name will be unique because there are 8 unique gym leaders. The type will help us identify what “type” of Pokemon the specified gym leader specializes in. A gym leader can only specialize in one type of Pokemon. The primary key will be the gym leader ID.

There are many different abilities that Pokemon are able to use in the game. All the abilities in this game have unique names. In this table, each ability will be assigned an ID (auto-incrementing) and an ability name. Each ability will have a type attribute as well as a damage attribute. Different abilities can do the same amount of damage, but every ability will have a unique name. The primary key will be the ability ID.

There are many different towns in the Pokemon world. These towns are usually defined by their residing gym leader. Most towns have a gym leader. A town that has a gym leader can only have at most one gym leader. A gym leader can also only be in one town. Each town will be assigned a town ID, a town name, and a description.

Relationships:

The first relationship is the many-to-many relationship between Pokemon and types. A Pokemon can be many different types and a single type can have many different Pokemon. Understanding how this relationship works will have a large effect on how successful you are at playing the game. In short, each Pokemon has at least one type and at most two types. These types must be different (you cannot be fire/fire or water/water). In this database, we will represent this relationship in a different table where each row has the Pokemon ID and a type ID. The Pokemon ID and the type ID will make up the primary key.

Another relationship is also a many-to-many relationship between Pokemon and abilities. Pokemon can learn many different abilities and abilities can be learned by many different Pokemon. This relationship can be represented by another table using the Pokemon ID and ability ID as the primary key.

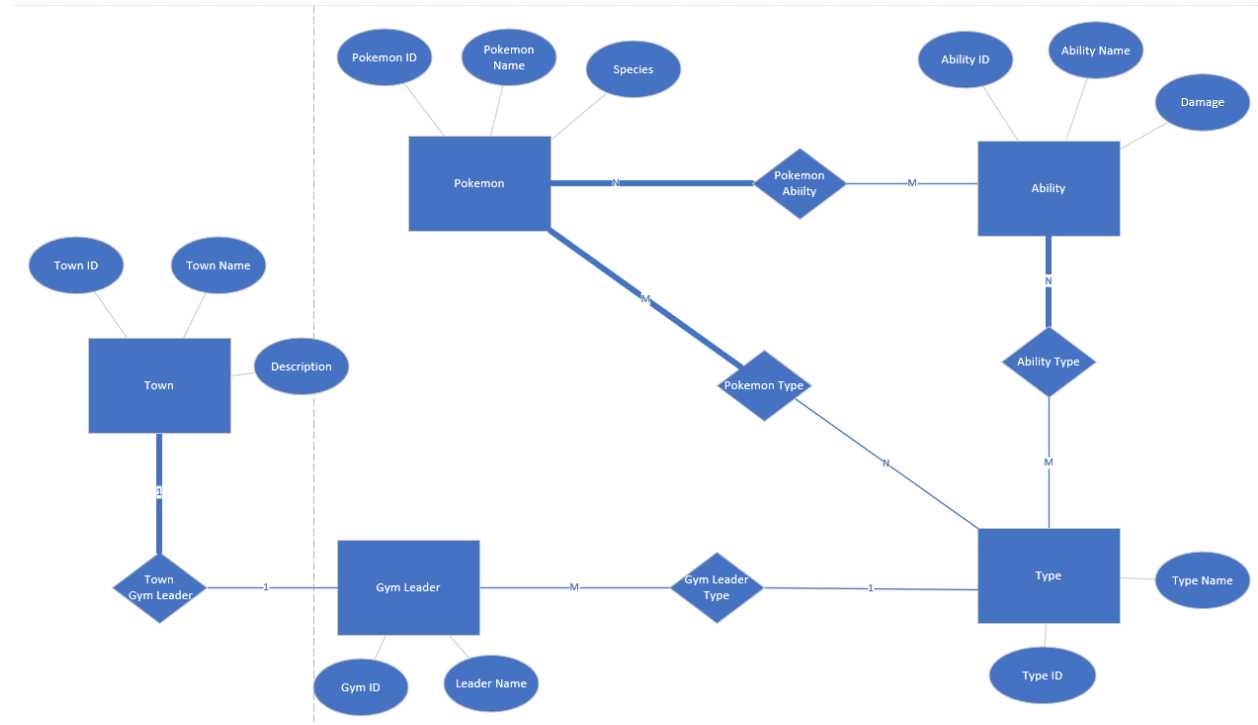
Another relationship is between abilities and types. Each ability can only have one type, and each type can have many different abilities. This is a many-to-one relationship.

Another relationship is between the gym leaders and the type of Pokemon they are using. Each gym leader can only use one type of Pokemon. Not all types have a gym leader associated with them. Therefore, this is a many-to-one relationship. The primary key for the table will be the gym leader ID and the type ID.

The last relationship is a one-to-one relationship between a gym leader and a town. Each town can only have one unique gym leader and each gym leader can only belong to one specific

town. This pairing cannot be changed. The primary key for this table will be the gym ID and the town ID.

ERD (Entity Relationship Diagram)



Schema

