

**University of British Columbia, Vancouver**

Department of Computer Science

---

# CPSC 304 Project Cover Page

Milestone #:   2  

Date:   July 21, 2024  

Group Number:   7  

Name	Student Number	CS Alias (User ID)	Preferred Email Address
Hayden Chan	49928377	p3x1c	haydenphchan@gmail.com
Kristen Tan	55327829	q7n0y	kristendenise9@gmail.com
Cody Zhao	36025468	r0j8e	codyz2770@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your email address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

## **Project Summary**

### **What is the domain of the application? Describe it.**

The domain of our application is in gaming and information management, specifically for Pokemon players. It focuses on information about different types of Pokemon for Pokemon Players providing a comprehensive database and management system for Pokemon-related information. The domain is Pokemon/Trainer/Moves/Games.

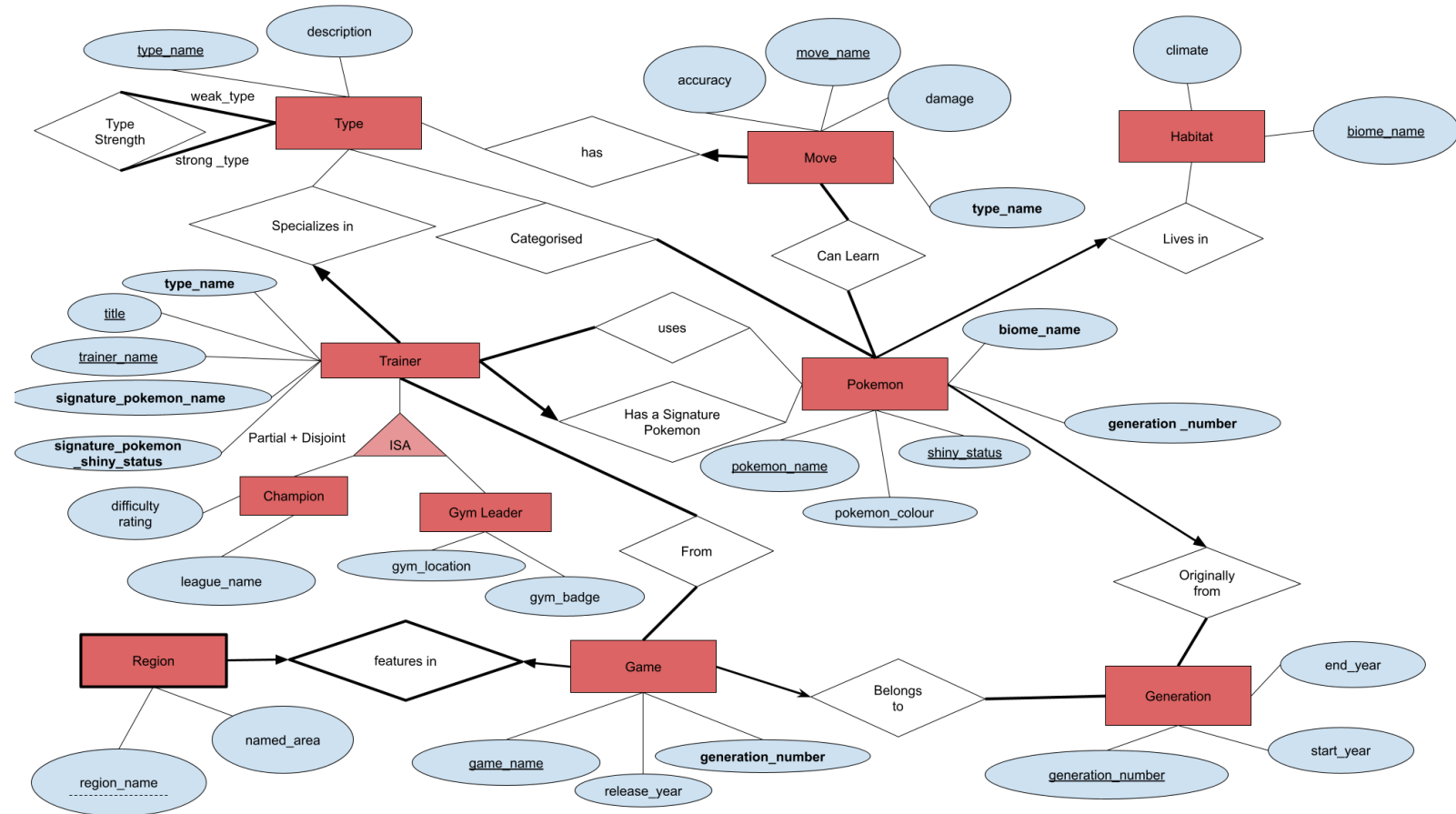
### **What aspects of the domain are modeled by the database?**

The database models various details within the pokemon world. This includes but not limited to, Pokemons, their types, their moves, where they live, and what champions use them. It also models the games and generations that the games are set in.

For example, in real life applications, users can use the data within the database to strategize when playing the pokemon games. Developers can use this information to create several features such as

- Choosing the optimal Pokemon for upcoming battles.
- Recommend the best moves for a given Pokemon based on the accuracy and damage that is needed for the battle.
- Allows players to see what type of Pokemon is weak against or strong against each other.
- Prepare players for battles against a champion team utilizing the difficulty rating.

## ER Diagram



## Changes Made:

- Added "difficulty\_rating" as an Attribute of "Champion" entity
- Added "gym\_location" as an attribute of "gym\_leader" entity
- Added a "has a signature pokemon" relationship
- Added "title" as a composite key to "Trainer" entity
- Reworded "Type" entity's attribute from "descriptions" to "description"
- Reworded "matchup" relationship into "Type Strength" relationships
- Reworded "Region" entity's attribute "named\_area" from "size"
- Added various foreign keys to ensure relationships can be accessed
- Added "shiny\_status" and "pokemon\_colour" to allow for decomposition

***Unchanged Comments:***

Regarding the region being a weak entity, we believe that region is associated with a game because, in many instances, the same region shows up in different games, and there are changes made to the region that makes it unique in each game, although keeping the same name and many iconic locations, while still having some changes and new areas. This makes a region not uniquely identifiable purely by its name, but uniquely identifiable once you combine the region name and the game name.

For example, the Hoenn region features in both Pokemon Ruby and Pokemon Omega Ruby. However, while sharing the name “Hoenn”, there are many differences that make the region in the two games different. A few examples would be differences in towns (Ex. Mauville City was overhauled into Mauville Hills, a large shopping complex), Gym designs (Ex. More complex puzzles in Dewford and Fortree gym) , and new locations (Ex. Sea Mauville and new mirage spots). All of these changes make these two regions unique, but they still share the name “Hoenn”, making it a weak entity that requires the game name to uniquely identify the exact region.

Regarding the suggestion to add a currently learned moves relationship between pokemon and moves, a Pokemon’s learned moves is variable. Pokemon come with random moves chosen from its list of moves that it can learn when captured, meaning two instances of the exact same Pokemon can come with different movesets. Thus, we thought it would make more sense to only track learnable moves for each Pokemon. Our project is also meant to be an informative database, so having it track a user’s Pokemon with their learned moves would fall outside its functionality.

## **Schema**

Type(type\_name: VARCHAR, description: VARCHAR)

TypeStrength(strong\_type\_name: VARCHAR, weak\_type\_name: VARCHAR)

Move(move\_name: VARCHAR, accuracy: INTEGER, damage: INTEGER, type\_name: VARCHAR)

CanLearn (pokemon\_name: VARCHAR, shiny\_status: BIT, move\_name: VARCHAR)

Habitat(biome\_name: VARCHAR, climate: VARCHAR)

Trainer(title: VARCHAR, trainer\_name: VARCHAR, type\_name: VARCHAR, signature\_pokemon\_name: VARCHAR, signature\_pokemon\_shiny\_status: BIT)

Champion(title: VARCHAR, trainer\_name: VARCHAR, difficulty\_rating: INTEGER, league\_name: VARCHAR)

Gym Leader(title: VARCHAR, trainer\_name: VARCHAR, gym\_location: VARCHAR, gym\_badge: VARCHAR)

Uses (trainer\_name: VARCHAR, title: VARCHAR, pokemon\_name: VARCHAR, shiny\_status: BIT)

Categorised (pokemon\_name: VARCHAR, shiny\_status: BIT, type\_name: VARCHAR)

Pokemon(pokemon\_name: VARCHAR, shiny\_status: BIT, pokemon\_colour: VARCHAR, generation\_number: INTEGER, biome\_name: VARCHAR)

Game(game\_name: VARCHAR, release\_year: YEAR, generation\_number: INTEGER)

From(trainer\_name: VARCHAR, title: VARCHAR, game\_name VARCHAR)

Game\_Region(game\_name: VARCHAR, region\_name: VARCHAR, named\_area: INTEGER)

Generation(generation\_number: INTEGER, start\_year: YEAR, end\_year: YEAR)

PK:

- Type.type\_name
- TypeStrength.strong\_type\_name, TypeStrength.weak\_type\_name
- Move.move\_name
- CanLearn.pokemon\_name, CanLearn.move\_name, CanLearn.shiny\_status
- Habitat.biome\_name
- Game.game\_name
- Trainer.trainer\_name, Trainer.title
- Champion.trainer\_name, Champion.title
- GymLeader.trainer\_name, GymLeader.title
- Uses.trainer\_name, Uses.title, Uses.pokemon\_name, Uses.shiny\_status
- Categorised.pokemon\_name, Categorised.shiny\_status, Categorised.type\_name
- Pokemon.pokemon\_name, Pokemon.shiny\_status
- Game.game\_name
- From.trainer\_name, From.title, From.game\_name
- Game\_Region.game\_name, Game\_Region.region\_name
- Generation.generation\_number

CK:

- Type.description
- Champion.league\_name
- GymLeader.gym\_location, GymLeader.gym\_badge

FK:

- TypeStrength.strong\_type\_name, TypeStrength.weak\_type\_name
- Move.type\_name
- CanLearn.pokemon\_name, CanLearn.move\_name, CanLearn.shiny\_status
- Trainer.type\_name, Trainer.signature\_pokemon\_name, Trainer.signature\_pokemon\_shiny\_status
- Uses.trainer\_name, Uses.title, Uses.pokemon\_name, Uses.shiny\_status

- Categorised.pokemon\_name, Categorised.shiny\_status, Categorised.type\_name
- Pokemon.generation\_number, Pokemon.biome\_name
- Game.generation\_number
- From.trainer\_name, From.title, From.game\_name
- Game\_Region.game\_name

#### NOT NULL:

- Type.description
- Move.type\_name
- Habitat.climate
- Champion.difficulty\_rating, Champion.league\_name
- GymLeader.gym\_location, GymLeader.gym\_badge
- Pokemon.pokemon\_colour
- Game.release\_year
- Game\_Region.named\_area
- Generation.start\_year, Generation.end\_year

#### UNIQUE:

- Type.description
- Champion.league\_name
- GymLeader.gym\_location, GymLeader.gym\_badge
- Generation.start\_year, Generation.end\_year

## **Functional Dependencies (FDs)**

Type.type\_name -> Type.description

Type.description -> Type.type\_name

Moves.move\_name -> Moves.damage

Moves.move\_name -> Moves.accuracy

Moves.move\_name -> Moves.type\_name

Habitat.biome\_name -> Habitat.climate

Trainer.title -> Trainer.type\_name

Trainer.trainer\_name, Trainer.title -> Trainer.signature\_pokemon\_name

Trainer.trainer\_name, Trainer.title -> Trainer.signature\_pokemon\_shiny\_status

Champion.league\_name -> Champion.title

Champion.league\_name -> Champion.trainer\_name

GymLeader.gym\_location -> GymLeader.title

GymLeader.gym\_location -> GymLeader.trainer\_name

GymLeader.gym\_badge -> GymLeader.title

GymLeader.gym\_badge -> GymLeader.trainer\_name

Pokemon.pokemon\_name -> Pokemon.generation\_number

Pokemon.pokemon\_name -> Pokemon.biome\_name

Pokemon.pokemon\_name, Pokemon.shiny\_status -> Pokemon.pokemon\_colour

Game.game\_name -> Game.release\_year

Game.game\_name -> Game.generation\_number

Game.game\_name, Region.region\_name -> Region.named\_area

Generation.generation\_number -> Generation.start\_year

Generation.generation\_number -> Generation.end\_year



## **Normalization**

***“Pokemon” entity was decomposed down to “Pokemon\_Basic\_Info” and “Pokemon\_Colour”***

Pokemon(pokemon\_name: VARCHAR, shiny\_status: BIT, pokemon\_colour: VARCHAR, generation\_number: INTEGER, biome\_name: VARCHAR)

*From the FDs mentioned before:*

Pokemon.pokemon\_name -> Pokemon.generation\_number

Pokemon.pokemon\_name -> Pokemon.biome\_name

Pokemon.pokemon\_name, Pokemon.shiny\_status -> Pokemon.pokemon\_colour

*We can see that pokemon\_name -> generation\_number, biome\_name violates 3NF because pokemon\_name is not a superkey*

Pokemon\_Basic\_Info (pokemon\_name: VARCHAR, generation\_number: INTEGER, biome\_name: VARCHAR)

Pokemon\_Colour (pokemon\_name: VARCHAR, shiny\_status: BIT, colour: VARCHAR)

***“Trainer” entity was decomposed down to “Title\_Types” and “Trainer\_Info”***

Trainer( title: VARCHAR, trainer\_name: VARCHAR, type\_name: VARCHAR, signature\_pokemon\_name: VARCHAR, signature\_pokemon\_shiny\_status: BIT)

*From the FDs mentioned before:*

Trainer.title -> Trainer.type\_name

Trainer.trainer\_name, Trainer.title -> Trainer.signature\_pokemon\_name

Trainer.trainer\_name, Trainer.title -> Trainer.signature\_pokemon\_shiny\_status

*We can see that the title -> type\_name FD violates 3NF as title is not a superkey.*

Title\_Type (title: VARCHAR, type\_name: VARCHAR)

Trainer\_Info (title: VARCHAR, trainer\_name: VARCHAR,  
signature\_pokemon\_name: VARCHAR, signature\_pokemon\_shiny\_status: BIT)

*Other tables are already in 3NF and do not need normalizing*

***Below are all the table's schemas (Bolded are the updated tables)***

Type(type\_name: VARCHAR, description: VARCHAR)

TypeStrength(strong\_type\_name: VARCHAR, weak\_type\_name: VARCHAR)

Move(move\_name: VARCHAR, accuracy: INTEGER, damage: INTEGER, type\_name:  
VARCHAR)

CanLearn (pokemon\_name: VARCHAR, shiny\_status: BIT, move\_name: VARCHAR)

Habitat(biome\_name: VARCHAR, climate: VARCHAR)

**Title\_Type (title: VARCHAR, type\_name: VARCHAR)**

**Trainer\_Info (title: VARCHAR, trainer\_name: VARCHAR, signature\_pokemon\_name:  
VARCHAR, signature\_pokemon\_shiny\_status: BIT)**

Champion(title: VARCHAR, trainer\_name: VARCHAR, difficulty\_rating: INTEGER,  
league\_name: VARCHAR)

Gym Leader(title: VARCHAR, trainer\_name: VARCHAR, gym\_location: VARCHAR,  
gym\_badge: VARCHAR)

Uses (trainer\_name: VARCHAR, title: VARCHAR, pokemon\_name: VARCHAR,  
shiny\_status: BIT)

Categorised (pokemon\_name: VARCHAR, shiny\_status: BIT, type\_name: VARCHAR)

**Pokemon\_Basic\_Info (pokemon\_name: VARCHAR, generation\_number: INTEGER,  
biome\_name: VARCHAR)**

**Pokemon\_Colour (pokemon\_name: VARCHAR, shiny\_status: BIT, colour: VARCHAR)**

Game(game\_name: VARCHAR, release\_year: YEAR, generation\_number: INTEGER)

From(trainer\_name: VARCHAR, title: VARCHAR, game\_name VARCHAR)

Game\_Region(game\_name: VARCHAR, region\_name: VARCHAR, named\_area: VARCHAR)

Generation(generation\_number: INTEGER, start\_year: YEAR, end\_year: YEAR)

PK (Bolded are updates, rest are the same)

- Type.type\_name
- TypeStrength.strong\_type\_name, TypeStrength.weak\_type\_name
- Move.move\_name
- CanLearn.pokemon\_name, CanLearn.move\_name, CanLearn.shiny\_status
- Habitat.biome\_name
- Game.game\_name
- **Title\_Type.title**
- **Trainer\_Info.trainer\_name, Trainer\_Info.title**
- Champion.trainer\_name, Champion.title
- GymLeader.trainer\_name, GymLeader.title
- Uses.trainer\_name, Uses.title, Uses.pokemon\_name, Uses.shiny\_status
- Categorised.pokemon\_name, Categorised.shiny\_status, Categorised.type\_name
- **Pokemon\_Basic\_Info.pokemon\_name**
- **Pokemon\_Colour.pokemon\_name, Pokemon\_Colour.pokemon\_colour**
- Pokemon.pokemon\_name, Pokemon.shiny\_status
- Game.game\_name
- From.trainer\_name, From.title, From.game\_name
- Game\_Region.game\_name, Game\_Region.region\_name
- Generation.generation\_number

CK (Same as before)

- Type.description
- Champion.league\_name
- GymLeader.gym\_location, GymLeader.gym\_badge

FK (Bolded are updates, rest are the same)

- TypeStrength.strong\_type\_name, TypeStrength.weak\_type\_name
- Move.type\_name
- CanLearn.pokemon\_name, CanLearn.move\_name, CanLearn.shiny\_status
- **Title\_Type.type\_name**
- **Trainer\_Info.signature\_pokemon\_name,**  
**Trainer\_Info.signature\_pokemon\_shiny\_status**
- Uses.trainer\_name, Uses.title, Uses.pokemon\_name, Uses.shiny\_status
- Categorised.pokemon\_name, Categorised.shiny\_status,  
Categorised.type\_name
- **Pokemon\_Basic\_Info.generation\_number,**  
**Pokemon\_Basic\_Info.biome\_name**
- Game.generation\_number
- From.trainer\_name, From.title, From.game\_name
- Game\_Region.game\_name

## **SQL DDL**

```
CREATE TABLE Type (  
    type_name VARCHAR PRIMARY KEY,  
    type_description VARCHAR NOT NULL,  
    UNIQUE(type_description)  
);
```

```
CREATE TABLE TypeStrength (  
    strong_type_name VARCHAR,  
    weak_type_name VARCHAR,  
    PRIMARY KEY(strong_type_name, weak_type_name),  
    FOREIGN KEY(strong_type_name)  
        REFERENCES Type(type_name)  
    FOREIGN KEY(weak_type_name)  
        REFERENCES Type(type_name)  
);
```

```
CREATE TABLE Move (  
    move_name VARCHAR PRIMARY KEY,  
    accuracy INTEGER,  
    damage INTEGER,  
    type_name VARCHAR NOT NULL,  
    FOREIGN KEY(type_name)
```

REFERENCES Type(type\_name)

ON DELETE NO ACTION

ON UPDATE CASCADE

);

CREATE TABLE CanLearn (

pokemon\_name VARCHAR,

shiny\_status BIT,

move\_name VARCHAR,

PRIMARY KEY(pokemon\_name, shiny\_status, move\_name)

FOREIGN KEY(pokemon\_name, shiny\_status)

REFERENCES Pokemon(pokemon\_name, shiny\_status)

ON DELETE NO ACTION

ON UPDATE CASCADE

FOREIGN KEY(move\_name)

REFERENCES Move(move\_name)

ON DELETE NO ACTION

ON UPDATE CASCADE

);

CREATE TABLE Habitat (

biome\_name VARCHAR PRIMARY KEY,

```
        climate VARCHAR NOT NULL,  
  
);  
  
CREATE TABLE Title_Type (  
  
    title VARCHAR PRIMARY KEY,  
  
    type_name VARCHAR,  
  
    FOREIGN KEY(type_name)  
  
        REFERENCES Type(type_name)  
  
        ON DELETE NO ACTION  
  
        ON UPDATE CASCADE  
  
);
```

```
CREATE TABLE Trainer_Info (  
  
    title VARCHAR,  
  
    trainer_name VARCHAR,  
  
    signature_pokemon_name VARCHAR,  
  
    signature_pokemon_shiny_status BIT,  
  
    PRIMARY KEY(title, trainer_name),  
  
    FOREIGN KEY(signature_pokemon_name,  
signature_pokemon_shiny_status)  
  
        REFERENCES Pokemon(pokemon_name, shiny_status)  
  
        ON DELETE NO ACTION  
  
        ON UPDATE CASCADE
```

);

CREATE TABLE Champion (

title VARCHAR,

trainer\_name VARCHAR,

difficulty\_rating INTEGER NOT NULL,

league\_name VARCHAR NOT NULL,

PRIMARY KEY (trainer\_name, title),

FOREIGN KEY (trainer\_name, title)

REFERENCES Trainer(trainer\_name, title)

ON DELETE NO ACTION

ON UPDATE CASCADE,

UNIQUE(league\_name)

);

CREATE TABLE Gym Leader (

title VARCHAR,

trainer\_name VARCHAR,

gym\_location VARCHAR NOT NULL,

gym\_badge VARCHAR NOT NULL,

PRIMARY KEY (trainer\_name, title),

FOREIGN KEY (trainer\_name, title)



```
REFERENCES Trainer(trainer_name, title)

ON DELETE NO ACTION

ON UPDATE CASCADE,

UNIQUE(gym_location, gym_badge)

);

CREATE TABLE Uses (

    trainer_name VARCHAR,

    title VARCHAR,

    pokemon_name VARCHAR,

    shiny_status BIT,

    PRIMARY KEY(trainer_name, title, pokemon_name, shiny_status),

    FOREIGN KEY(trainer_name, title)

        REFERENCES Trainer(trainer_name, title)

        ON DELETE NO ACTION

        ON UPDATE CASCADE

    FOREIGN KEY(pokemon_name, shiny_status)

        REFERENCES Pokemon(pokemon_name, shiny_status)

        ON DELETE NO ACTION

        ON UPDATE CASCADE

);
```

```
CREATE TABLE Categorised (  
    pokemon_name VARCHAR,  
    shiny_status BIT,  
    type_name VARCHAR,  
    PRIMARY KEY(pokemon_name, shiny_status, type_name)  
    FOREIGN KEY(pokemon_name, shiny_status)  
        REFERENCES Pokemon(pokemon_name, shiny_status)  
        ON DELETE NO ACTION  
        ON DELETE CASCADE  
    FOREIGN KEY(type_name)  
        REFERENCES Type(type_name)  
        ON DELETE NO ACTION  
        ON DELETE CASCADE  
);
```

```
CREATE TABLE Pokemon_Basic_Info (  
    pokemon_name VARCHAR PRIMARY KEY,  
    generation_number INTEGER,  
    biome_name VARCHAR,  
    FOREIGN KEY(generation_number)  
        REFERENCES Generation(generation_number)  
        ON DELETE NO ACTION
```

```
        ON UPDATE CASCADE

FOREIGN KEY(biome_name)

        REFERENCES Habitat(biome_name)

        ON DELETE NO ACTION

        ON UPDATE CASCADE

);
```

```
CREATE TABLE Pokemon_Colour (

    pokemon_name VARCHAR PRIMARY KEY,

    shiny_status BIT,

    colour VARCHAR,

    UNIQUE(colour)

);
```

```
CREATE TABLE Game (

    game_name VARCHAR PRIMARY KEY,

    release_year YEAR NOT NULL,

    generation_number INTEGER NOT NULL,

    FOREIGN KEY(generation_number)

        REFERENCES Generation(generation_number)

        ON DELETE NO ACTION
```

ON UPDATE CASCADE

);

CREATE TABLE From (

trainer\_name VARCHAR,

title VARCHAR,

game\_name VARCHAR,

PRIMARY KEY(trainer\_name, title, game\_name)

FOREIGN KEY(trainer\_name, title)

REFERENCES Trainer(trainer\_name, title)

ON DELETE NO ACTION

ON UPDATE CASCADE

FOREIGN KEY(game\_name)

REFERENCES Game(game\_name)

ON DELETE NO ACTION

ON UPDATE CASCADE

);

CREATE TABLE Game\_Region (

game\_name VARCHAR,

region\_name VARCHAR,

named\_area VARCHAR NOT NULL,

PRIMARY KEY(game\_name, region\_name)

FOREIGN KEY(game\_name)

REFERENCES Game(game\_name)

ON DELETE CASCADE

);

CREATE TABLE Generation (

generation\_number INTEGER PRIMARY KEY,

start\_year YEAR NOT NULL,

end\_year YEAR NOT NULL,

UNIQUE(start\_year, end\_year)

);

## **INSERT Statements**

### ***Move Inserts***

```
INSERT
INTO Move (move_name, damage, accuracy, type_name)
VALUES ('Water Gun', 40, 100, 'Water')
```

```
INSERT
INTO Move (move_name, damage, accuracy, type_name)
VALUES ('Stealth Rock', NULL, NULL, 'Rock')
```

```
INSERT
INTO Move (move_name, damage, accuracy, type_name)
VALUES ('Hyper Beam', 150, 90, 'Normal')
```

```
INSERT
INTO Move (move_name, damage, accuracy, type_name)
VALUES ('Fire Blast', 110, 85, 'Fire')
```

```
INSERT
INTO Move (move_name, damage, accuracy, type_name)
VALUES ('Leech Seed', NULL, 90, 'Grass')
```

### ***Type Inserts***

```
INSERT
INTO Type (type_name, description)
VALUES ('Normal', 'The Normal type is the most basic type of Pokémon. They are very common and appear from the very first route you visit.')
```

```
INSERT
INTO Type (type_name, description)
VALUES ('Fire', 'Fire is one of the three basic elemental types along with Water and Grass, which constitute the three starter Pokémon.')
```

```
INSERT
INTO Type (type_name, description)
VALUES ('Rock', 'Rock is a solid type as one might expect. Like Steel, Rock Pokémon usually have high defense')
```

```
INSERT
INTO Type (type_name, description)
VALUES ('Water', 'Water is one of the three basic elemental types along with Fire and Grass, which constitute the three starter Pokémon')
```

```
INSERT
INTO Type (type_name, description)
VALUES ('Grass', 'Grass is one of the three basic elemental types along with Fire and Water, which constitute the three starter Pokémon.')
```

### ***Habitat Inserts***

```
INSERT  
INTO Habitat (biome_name, climate)  
VALUES ('Ocean', 'Warm')
```

```
INSERT  
INTO Habitat (biome_name, climate)  
VALUES ('Town', 'Temperate')
```

```
INSERT  
INTO Habitat (biome_name, climate)  
VALUES ('Desert', 'Hot')
```

```
INSERT  
INTO Habitat (biome_name, climate)  
VALUES ('Snowy Mountain', 'Cold')
```

```
INSERT  
INTO Habitat (biome_name, climate)  
VALUES ('Forest', 'Tropical')
```

### ***Generation Inserts***

```
INSERT  
INTO Generation (generation_number, start_year, end_year)  
VALUES (1, 1996, 1998)
```

```
INSERT  
INTO Generation (generation_number, start_year, end_year)  
VALUES (2, 1999, 2001)
```

```
INSERT  
INTO Generation (generation_number, start_year, end_year)  
VALUES (3, 2002, 2005)
```

```
INSERT  
INTO Generation (generation_number, start_year, end_year)  
VALUES (4, 2006, 2009)
```

```
INSERT  
INTO Generation (generation_number, start_year, end_year)  
VALUES (5, 2010, 2012)
```

### ***Game Inserts***

```
INSERT  
INTO Game (game_name, release_year, generation_number)  
VALUES ('Pokemon Red', 1996, 1)
```

```
INSERT  
INTO Game (game_name, release_year, generation_number)  
VALUES ('Pokemon Crystal', 2000, 2)
```

```
INSERT  
INTO Game (game_name, release_year, generation_number)  
VALUES ('Pokemon FireRed', 2004, 3)
```

```
INSERT  
INTO Game (game_name, release_year, generation_number)  
VALUES ('Pokemon Pearl', 2006, 4)
```

```
INSERT  
INTO Game (game_name, release_year, generation_number)  
VALUES ('Pokemon Black', 2010, 5)
```

### ***Game\_Region Inserts***

```
INSERT  
INTO Game_Region (game_name, region_name, named_area)  
VALUES ('Pokemon Red', 'Kanto', 50)
```

```
INSERT  
INTO Game_Region (game_name, region_name, named_area)  
VALUES ('Pokemon Crystal', 'Johto', 46)
```

```
INSERT  
INTO Game_Region (game_name, region_name, named_area)  
VALUES ('Pokemon FireRed', 'Kanto', 69)
```

```
INSERT  
INTO Game_Region (game_name, region_name, named_area)  
VALUES ('Pokemon Pearl', 'Sinnoh', 75)
```

```
INSERT  
INTO Game_Region (game_name, region_name, named_area)  
VALUES ('Pokemon Black', 'Unova', 55)
```



### ***From Inserts***

```
INSERT  
INTO From (title, trainer_name, game_name)  
VALUES ('Champion', 'Cynthia', 'Pokemon Pearl')
```

```
INSERT  
INTO From (title, trainer_name, game_name)  
VALUES ('Youngster', 'Joey', 'Pokemon Crystal')
```

```
INSERT  
INTO From (title, trainer_name, game_name)  
VALUES ('Fisher', 'Chip', 'Pokemon FireRed')
```

```
INSERT  
INTO From (title, trainer_name, game_name)  
VALUES ('Rival', 'Blue', 'Pokemon Red')
```

```
INSERT  
INTO From (title, trainer_name, game_name)  
VALUES ('Pilot', 'Chase', 'Pokemon Black')
```

### ***Pokemon Basic Info Inserts***

```
INSERT  
INTO Pokemon_Basic_Info (pokemon_name, generation_number, biome_name)  
VALUES('Giratina', 4, 'Distortion World')
```

```
INSERT  
INTO Pokemon_Basic_Info (pokemon_name, generation_number, biome_name)  
VALUES('Chimchar', 4, 'Forest')
```

```
INSERT  
INTO Pokemon_Basic_Info (pokemon_name, generation_number, biome_name)  
VALUES('Rayquaza', 3, 'atmosphere')
```

```
INSERT  
INTO Pokemon_Basic_Info (pokemon_name, generation_number, biome_name)  
VALUES('Eternatus', 8, 'the Max Lair')
```

```
INSERT  
INTO Pokemon_Basic_Info (pokemon_name, generation_number, biome_name)  
VALUES('Lickitung', 1, 'Lakes')
```

### ***Pokemon Colour Inserts***

```
INSERT  
INTO Pokemon_Colour(pokemon_name, shiny_status, colour)  
VALUES('Lickitung', 0, 'Pink')
```

```
INSERT  
INTO Pokemon_Colour(pokemon_name, shiny_status, colour)  
VALUES('Rayquaza', 0, 'Green')
```

```
INSERT  
INTO Pokemon_Colour(pokemon_name, shiny_status, colour)  
VALUES('Gengar', 1, 'light purple')
```

```
INSERT  
INTO Pokemon_Colour(pokemon_name, shiny_status, colour)  
VALUES('Lucario', 1, 'Yellow')
```

```
INSERT  
INTO Pokemon_Colour(pokemon_name, shiny_status, colour)  
VALUES('Kyogre', 1, 'magenta')
```

### ***Can Learn Inserts***

```
INSERT  
INTO PokemonMovesets(pokemon_name, move_name, shiny_status)  
VALUES('Dialga', 'Roar of time', 0)
```

```
INSERT  
INTO PokemonMovesets(pokemon_name, move_name, shiny_status)  
VALUES('Palkia', 'Spacial Rend', 1)
```

```
INSERT  
INTO PokemonMovesets(pokemon_name, move_name, shiny_status)  
VALUES('Pikachu', 'Thunderbolt', 0)
```

```
INSERT  
INTO PokemonMovesets(pokemon_name, move_name, shiny_status)  
VALUES('Metapod', 'Harden', 1)
```

```
INSERT  
INTO PokemonMovesets(pokemon_name, move_name, shiny_status)  
VALUES('Lickitung', 'Lick', 0)
```

### ***Type Strength Inserts***

```
INSERT  
INTO TypeStrength(strong_type_name, weak_type_name)  
VALUES('Water', 'Fire')
```

```
INSERT  
INTO TypeStrength(strong_type_name, weak_type_name)  
VALUES('Ground', 'Electric')
```

```
INSERT  
INTO TypeStrength(strong_type_name, weak_type_name)  
VALUES('Fighting', 'Normal')
```

```
INSERT  
INTO TypeStrength(strong_type_name, weak_type_name)  
VALUES('Poison', 'Grass')
```

```
INSERT  
INTO TypeStrength(strong_type_name, weak_type_name)  
VALUES('Fairy', 'Dragon')
```

### ***Trainer Info Inserts***

```
INSERT  
INTO Trainer_Info(title, trainer_name, signature_pokemon_name,  
signature_pokemon_shiny_status)  
VALUES('Water Gym Leader', 'Misty', 'Staryu', 1)
```

```
INSERT  
INTO Trainer_Info(title, trainer_name, signature_pokemon_name,  
signature_pokemon_shiny_status)  
VALUES('Rock-Type Gym Leader', 'Brock', 'Onix', 0)
```

```
INSERT  
INTO Trainer_Info(title, trainer_name, signature_pokemon_name,  
signature_pokemon_shiny_status)  
VALUES('Fisher', 'Chip', 'Magikarp', 0)
```

```
INSERT  
INTO Trainer_Info(title, trainer_name, signature_pokemon_name,  
signature_pokemon_shiny_status)  
VALUES('Youngster', 'Joey', 'Ratata', 0)
```

```
INSERT  
INTO Trainer_Info(title, trainer_name, signature_pokemon_name,  
signature_pokemon_shiny_status)  
VALUES('Dragon Champion', 'Cynthia', 'Garchomp', 0)
```

### ***Title Types Inserts***

```
INSERT  
INTO Title_Types (title, type_name)  
VALUES('Fisher','Water')
```

```
INSERT  
INTO Title_Types (title, type_name)  
VALUES('Water Gym Leader', 'Water')
```

```
INSERT  
INTO Title_Types (title, type_name)  
VALUES('Youngster','Normal')
```

```
INSERT  
INTO Title_Types (title, type_name)  
VALUES('Dragon Champion','Dragon')
```

```
INSERT  
INTO Title_Types (title, type_name)  
VALUES('Pilot','Flying')
```

### ***Champion Inserts***

```
INSERT  
INTO Trainer(title, trainer_name, type_name, difficulty_rating, league_name)  
VALUES('Psychic Champion', 'Blue', 'Psychic', 5, 'Indigo League')
```

```
INSERT  
INTO Trainer(title, trainer_name, type_name, difficulty_rating, league_name)  
VALUES('Dragon Champion', 'Lance', 'Dragon', 4, 'Indigo League')
```

```
INSERT  
INTO Trainer(title, trainer_name, type_name, difficulty_rating, league_name)  
VALUES('Dragon Champion', Cynthia, 'Dragon', 5, 'Sinnoh Pokémon League')
```

```
INSERT  
INTO Trainer(title, trainer_name, type_name, difficulty_rating, league_name)  
VALUES('Electric Champion', 'Ashe', 'Electric', 4, 'Alola Elite Four')
```

```
INSERT  
INTO Trainer(title, trainer_name, type_name, difficulty_rating, league_name)  
VALUES('Bug Champion', 'Alder', 'Bug', 4, 'Black Elite Four')
```

### ***Gym Leader Inserts***

```
INSERT  
INTO Gym_Leader(title, trainer_name, gym_location, gym_badge)  
VALUES('Rock Gym Leader', 'Brock', 'Pewter City', 'Boulder Badge')
```

```
INSERT  
INTO Gym_Leader(title, trainer_name, gym_location, gym_badge)  
VALUES('Water Gym Leader', 'Misty', 'Cerulean City', 'Cascade Badge')
```

```
INSERT  
INTO Gym_Leader(title, trainer_name, gym_location, gym_badge)  
VALUES('Electric Gym Leader', 'Lt. Surge', 'Vermillion City', 'Thunder Badge')
```

```
INSERT  
INTO Gym_Leader(title, trainer_name, gym_location, gym_badge)  
VALUES('Grass Gym Leader', 'Erika', 'Celadon City', 'Rainbow Badge')
```

```
INSERT  
INTO Gym_Leader(title, trainer_name, gym_location, gym_badge)  
VALUES('Poison Gym Leader', 'Koga', 'Fuchsia City', 'Soul Badge')
```

### ***Uses inserts***

```
INSERT  
INTO Uses(trainer_name, title, pokemon_name, shiny_status)  
VALUES('Brock', 'Rock Gym Leader', 'Geodude', 0)
```

```
INSERT  
INTO Uses(trainer_name, title, pokemon_name, shiny_status)  
VALUES('Misty', 'Water Gym Leader', 'Staryu', 0)
```

```
INSERT  
INTO Uses(trainer_name, title, pokemon_name, shiny_status)  
VALUES('Brock', 'Rock Gym Leader', 'Onix', 0)
```

```
INSERT  
INTO Uses(trainer_name, title, pokemon_name, shiny_status)  
VALUES('Erika', 'Grass Gym Leader', 'Tangela', 0)
```

```
INSERT  
INTO Uses(trainer_name, title, pokemon_name, shiny_status)  
VALUES('Erika', 'Poison Gym Leader', 'Vileplume', 0)
```

### ***Categorized inserts***

```
INSERT  
INTO Categorized(pokemon_name, shiny_status, type_name)  
VALUES('Kyogre', 0, 'Water')
```

```
INSERT  
INTO Categorized(pokemon_name, shiny_status, type_name)  
VALUES('Gengar', 0, 'Ghost')
```

```
INSERT  
INTO Categorized(pokemon_name, shiny_status, type_name)  
VALUES('Gengar', 0, 'Poison')
```

```
INSERT  
INTO Categorized(pokemon_name, shiny_status, type_name)  
VALUES('Charjabug', 0, 'Bug')
```

```
INSERT  
INTO Categorized(pokemon_name, shiny_status, type_name)  
VALUES('Charjabug', 0, 'Electric')
```