

-- CREATION OF DATABASE

```
/*
DROP TABLE pkmn_move;
DROP TABLE pkmn_type;
DROP TABLE trainer_item;
DROP TABLE trainer_pkmn;
DROP TABLE type_resistances;
DROP TABLE type_weaknesses;
DROP TABLE Item;
DROP TABLE Move;
DROP TABLE pokemon;
DROP TABLE trainer;
DROP TABLE type;
*/

CREATE TABLE Trainer (
    trainer_id INTEGER PRIMARY KEY AUTOINCREMENT,
    name VARCHAR(40) UNIQUE NOT NULL
);

CREATE TABLE Archetype (
    archetype_id INTEGER PRIMARY KEY AUTOINCREMENT,
    archetype_name VARCHAR(50) UNIQUE NOT NULL,
    archetype_description VARCHAR(300) NOT NULL
);

CREATE TABLE Trainer_Archetype (
    trainer_id INTEGER,
    archetype_id INTEGER,
    PRIMARY KEY (trainer_id, archetype_id),
    FOREIGN KEY (trainer_id) REFERENCES Trainer (trainer_id),
    FOREIGN KEY (archetype_id) REFERENCES Archetype (archetype_id)
);

CREATE TABLE Badge (
    badge_id INTEGER PRIMARY KEY AUTOINCREMENT,
    badge_name VARCHAR(50) UNIQUE
);
```

```
CREATE TABLE Trainer_Badges (  
  trainer_id INTEGER,  
  badge_id INTEGER,  
  date_badge_acquired DATE,  
  FOREIGN KEY (trainer_id) REFERENCES Trainer (trainer_id),  
  FOREIGN KEY (badge_id) REFERENCES Badge (badge_id),  
  PRIMARY KEY (trainer_id, badge_id)  
);
```

```
CREATE TABLE Item (  
  item_id INTEGER PRIMARY KEY AUTOINCREMENT,  
  name VARCHAR(40) DEFAULT 'Unknown',  
  cost INTEGER,  
  effect INTEGER,  
  item_description VARCHAR(300) NOT NULL  
);
```

```
CREATE TABLE Trainer_Item (  
  trainer_id INTEGER,  
  item_id INTEGER,  
  FOREIGN KEY (trainer_id) REFERENCES Trainer (trainer_id),  
  FOREIGN KEY (item_id) REFERENCES Item (item_id),  
  PRIMARY KEY (trainer_id, item_id)  
);
```

```
CREATE TABLE Pokemon (  
  u_id INTEGER PRIMARY KEY AUTOINCREMENT,  
  national_pokedex_id INTEGER NOT NULL,  
  name VARCHAR(40) DEFAULT 'Unknown',  
  ability VARCHAR(40),  
  level INTEGER DEFAULT 0,  
  health_points INTEGER DEFAULT 0,  
  attack INTEGER DEFAULT 0 CHECK (attack <= 250 AND attack >= 0),  
  defense INTEGER DEFAULT 0 CHECK (defense <= 250 AND defense >= 0),  
  special_attack INTEGER DEFAULT 0 CHECK (special_attack <= 250 AND special_attack >= 0),  
  special_defense INTEGER DEFAULT 0 CHECK (special_defense <= 250 AND special_defense >= 0),  
  speed INTEGER DEFAULT 0 CHECK (speed <= 250 AND speed >= 0)  
);
```

```
CREATE TABLE Egg_Group (  
    egg_group_name VARCHAR(40) PRIMARY KEY  
);
```

```
CREATE TABLE Pokemon_Egg_Groups (  
    u_id INTEGER,  
    egg_group_name VARCHAR(40) ,  
    FOREIGN KEY (u_id) REFERENCES Pokemon (u_id),  
    FOREIGN KEY (egg_group_name) REFERENCES Egg_Group (egg_group_name),  
    PRIMARY KEY (u_id, egg_group_name)  
);
```

```
CREATE TABLE Trainer_pkmn (  
    trainer_id INTEGER,  
    u_id INTEGER,  
    FOREIGN KEY (trainer_id) REFERENCES Trainer (trainer_id),  
    FOREIGN KEY (u_id) REFERENCES Pokemon (u_id),  
    PRIMARY KEY (trainer_id, u_id)  
);
```

```
CREATE TABLE Type (  
    type_name VARCHAR(40) PRIMARY KEY,  
    type_description VARCHAR(300) NOT NULL  
);
```

```
CREATE TABLE Type_Weaknesses (  
    type_name VARCHAR(40),  
    weak_to_type VARCHAR(40),  
    FOREIGN KEY (type_name) REFERENCES Type (type_name),  
    PRIMARY KEY (type_name, weak_to_type)  
);
```

```
CREATE TABLE Type_resistances (  
    type_name VARCHAR(40),  
    resists_type VARCHAR(40),  
    FOREIGN KEY (type_name) REFERENCES Type (type_name),  
    FOREIGN KEY (resists_type) REFERENCES Type (type_name),  
    PRIMARY KEY (type_name, resists_type)  
);
```

```

CREATE TABLE Pkmn_type (
  u_id INTEGER,
  type_name VARCHAR(40),
  FOREIGN KEY (u_id ) REFERENCES Pokemon (u_id ),
  FOREIGN KEY (type_name) REFERENCES Type (type_name),
  PRIMARY KEY (u_id , type_name)
);

CREATE TABLE Move (
  move_name VARCHAR(40) PRIMARY KEY,
  base_power INTEGER CHECK (base_power <= 200 AND base_power >= 0),
  power_type VARCHAR(40),
  move_description VARCHAR(300),
  FOREIGN KEY (power_type) REFERENCES Type (type_name)
);

CREATE TABLE Pkmn_move (
  u_id INTEGER,
  move_name VARCHAR(40),
  FOREIGN KEY (u_id) REFERENCES Pokemon (u_id),
  FOREIGN KEY (move_name) REFERENCES Move (move_name),
  PRIMARY KEY (u_id, move_name)
);

```

-- ADD TUPLES TO DATABASE (CHATGPT VALUES)

Constraints for each table:

Trainer: at least 10 tuples, max 20 tuples

Archetype: At least 5 tuples, max 10 tuples

Trainer_archetype: Same number of tuples that the Trainer table has. Assign each trainer an archetype.

Badge: All of the badges in the Pokemon red, blue, and yellow games. A tuple for each badge.

Trainer_Badges: At least half of the trainers must have at least 3 badges, with a quarter of the trainers having more than 5.

Item: At least 30 items. Have a mix of pokeballs, potions, and other popular items.

Trainer_item: Each trainer should have at least 2 items.

Pokemon: At least 30 different pokemon species from the Pokemon red, blue, and yellow games. And make sure that some tuples are the same pokemon species. Have realistic stats for each tuple. Ensure the pokemon Syther is there at least 3 times.

Egg_group: A tuple for each egg group in the Pokemon red, blue, and yellow games.

Pokemon_egg_group: Make sure that each pokemon species has their correct egg groups. It's okay for a species to have multiple egg_groups.

Trainer_pkmn: Make sure each trainer has at least 1 pokemon with a max number of 6 pokemon.

Type: A tuple for each type in the Pokemon red, blue, and yellow games.

Type_weaknesses: For each type, include a tuple for all of its weaknesses.

Type_resistance: For each type, include a tuple for all of its resistance (strength) types.

Pkmn_type: Correctly assign each pokemon species to its type.

Move: Have at least 70 different moves.

Pkmn_move: Make sure every pokemon tuple has at least 1 move with a max of 4 moves.

[Link to queries populating the database](#)