

1. Project Discription

Our Pokemon Data Management System is designed to help trainers optimize their gameplay by allowing them to store, delete, and retrieve specific Pokemon data. In addition to basic data management, the system offers insights that support strategic decision-making. Users can view average attack values grouped by type name, identify type names with an average defense greater than 10, and find trainers who own at least one Pokémon from every category. Through these features, our system helps trainers make more informed choices and improve their overall game performance.

2. Schema Differences and Explanation

We did not make changes.

3. SQL Query List and Locations in Code

Functionality	SQL Query Summary	Line(s)
Queries1: INSERT Insert (1) insertPokemonTrainstable	Insert into PokemonTrains with FK check on TypeName	appService.js line: 135
Query2: Update	Updates any combination of non-PK fields: PokemonName, TypeName, PokemonGender, Ability, TrainerID.	appService.js, line: 188
Query3: Delete	Delete a tuple from PokemonTrains by PokemonID	appService.js, line: 234
Query4: Selection	Allowed to search for tuples using any number of AND/OR clauses and combinations of attributes in PokemonTrains	appService.js, line: 255
Query5: Projection	Project any number of attributes from the PokemonTrains table	appService.js, line: 280
Query6: Join	Join Trainer and PokemonTrains to find all Pokemon trained by a specific trainer	appService.js, line: 296

Query7: Aggregation With GROUP BY	Get average attack value grouped by TypeName	appService.js, line 316
Query8: Aggregation With HAVING	Get TypeName with average defense > 10	appService.js, line 334
Query9: Nested Aggregation	Get trainer names whose Pokemon have above-average total stats	appService.js, line: 353
Query10: Division	Get trainers who own at least one Pokemon from every category	appService.js, line: 375