

Parcialito II: SQL

Base de Datos 75.15/75.28/95.05 - Cátedra Román

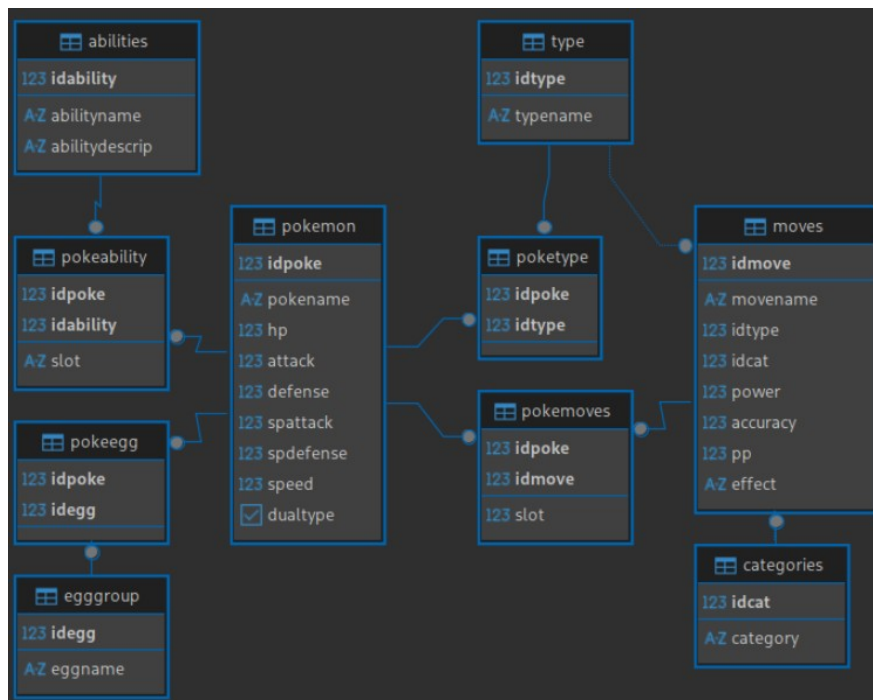
Segundo Cuatrimestre 2025

Melanie García Lapegna - 111848

Creación y armado de la base de datos

En el repositorio de github (<https://github.com/base-de-datos-fiuba/db-sql-2C25>) encontrarán los archivos .sql, que contiene el dataset a utilizar para entender schema de las tablas y el contenido de las mismas.

Esquema



Ejercicios

Considerando los esquemas del dataset de notas visto en clase, resolver los siguientes puntos escribiendo para cada uno de ellos una única consulta SQL que devuelva el resultado pedido.

1. Encontrar el **nombre**, **ataque** y **defensa** de todos los pokémon cuyo poder de todos sus movimientos de categoría '*Physical*' sea mayor a 85.

```
SELECT p.pokename AS nombre , p.attack AS ataque , p.defense AS defensa
FROM pokemon p
      INNER JOIN pokemoves pm ON (p.idpoke = pm.idpoke)
      INNER JOIN moves m ON (pm.idmove = m.idmove )
      INNER JOIN categories c ON (m.idcat= c.idcat )
WHERE c.category='Physical'
GROUP BY p.idpoke , p.pokename , p.attack , p.defense
HAVING min(m.power) > 85
```

-- Resultados --

/*

nombre	ataque	defensa
Machamp	130	80
Nidoking	102	77
Dugtrio	100	50
Muk	105	75
Nidoqueen	92	87
Diglett	55	25
Graveler	95	115
Sandslash	100	110
Sandshrew	75	85

*/

2. Encontrar todos los Pokemon tienen movimientos de tipo '*Normal*' y '*Flying*', pero que no tienen ninguna del tipo '*Fire*'.

```
SELECT DISTINCT p1.idpoke, p1.pokemonname AS nombre
FROM pokemon p1
    INNER JOIN pokemoves pk1 ON (p1.idpoke = pk1.idpoke )
    INNER JOIN moves m1 ON (pk1.idmove = m1.idmove)
    INNER JOIN type t1 ON (m1.idtype = t1.idtype)
    -- para que sea el mismo pokemon
    INNER JOIN pokemon p2 ON (p1.idpoke = p2.idpoke)
    INNER JOIN pokemoves pk2 ON (p2.idpoke = pk2.idpoke)
    INNER JOIN moves m2 ON (pk2.idmove = m2.idmove)
    INNER JOIN type t2 ON (m2.idtype = t2.idtype)
--pokemones que tienen movimientos de tipo 'Normal' y 'Flying'
WHERE t1.typename = 'Normal' AND t2.typename = 'Flying'
EXCEPT (
    -- pokemones que tienen movimientos de tipo Fire
    SELECT DISTINCT p.idpoke , p.pokemonname --distinct ya que un pokemon
--podria tener mas de un movimiento fire
    FROM pokemon AS p
        INNER JOIN pokemoves pk ON (p.idpoke = pk.idpoke )
        INNER JOIN moves m ON (pk.idmove = m.idmove)
        INNER JOIN type t ON (m.idtype = t.idtype)
    WHERE t.typename = 'Fire'
)

-- Resultados --
/*
idpoke|nombre      |
-----+-----+
    17|Pidgeotto   |
   144|Articuno    |
    18|Pidgeot     |
   145|Zapdos      |
    16|Pidgey      |
*/
```

3. Encontrar los 5 tipos de Pokemon con el mayor promedio de “ataque especial” (*spattack*). Devolver el **nombre del tipo** y su **promedio**. En caso de empates, resolverlo utilizando el nombre del tipo.

```
SELECT t.typename AS nombre_del_tipo, AVG(p.spattack) AS promedio
FROM poketype pt
    INNER JOIN type t on(pt.idtype = t.idtype)
    INNER JOIN pokemon p on(pt.idpoke = p.idpoke)
GROUP BY t.idtype , t.typename
ORDER BY AVG(p.spattack) DESC , t.typename
LIMIT 5
```

```
-- Resultados --
```

```
/*
nombre_del_tipo|promedio          |
-----+-----+
Dark           |125.0000000000000000|
Dragon         |125.0000000000000000|
Ghost          |115.0000000000000000|
Steel          |103.3333333333333333|
Electric       | 96.0000000000000000|
*/
```

4. Obtener el **ID**, **nombre** y la **velocidad** del Pokemon, para aquellos que tengan una suma de “*poder*” (*power*) de todos sus movimientos mayor a 250 y un promedio de “*precisión*” (*accuracy*) de al menos 95.

```
SELECT p.idpoke AS id, p.pokename AS nombre , p.speed AS velocidad
FROM pokemon p
      INNER JOIN pokemoves pm ON (p.idpoke = pm.idpoke)
      INNER JOIN moves m ON (pm.idmove = m.idmove)
GROUP BY p.idpoke , p.pokename , p.speed
HAVING sum(m.power) > 250 AND avg(m.accuracy) >= 95
```

-- Resultados --

/*

id	nombre	velocidad
18	Pidgeot	101
145	Zapdos	100
8	Wartortle	58
146	Moltres	90
6	Charizard	100
9	Blastoise	78
25	Pikachu	90
17	Pidgeotto	71

*/

5. Se necesita saber qué Pokemon/s tiene/n la estadística de '*defensa*' (*defense*) más baja, sólo considerando los Pokémon de tipo '*Ground*'. Devolver el **nombre** del Pokémon y su **defensa**.

```
SELECT p.pokeName AS nombre , p.defense AS defensa
FROM pokemon p
      INNER JOIN poketype pt ON (p.idpoke = pt.idpoke)
      INNER JOIN type t ON (pt.idtype = t.idtype)
WHERE t.typeName = 'Ground'
AND p.defense <= ALL (
      SELECT p.defense
      FROM pokemon p
      INNER JOIN poketype pt ON (p.idpoke = pt.idpoke)
      INNER JOIN type t ON (pt.idtype = t.idtype)
      WHERE t.typeName = 'Ground'
)

-- Resultados --
/*
nombre      |defensa|
-----+-----+
Diglett     |      25|
*/
```

6. Encontrar aquellos Pokemons que conocen movimientos de todas las categorías (categories) posibles.

```
SELECT p.idpoke AS id, p.pokename AS nombre
FROM pokemon p
      INNER JOIN pokemoves pm ON (p.idpoke = pm.idpoke)
      INNER JOIN moves m ON (pm.idmove = m.idmove)
      INNER JOIN categories c ON (m.idcat = c.idcat)
GROUP BY p.idpoke , p.pokename
HAVING count(DISTINCT m.idcat) = (
      --cantidad de categorias de movimientos
      SELECT count(*)
      FROM categories
)

-- Resultados --
/*
id|nombre      |
--+-----+
 3|Venusaur    |
 4|Charmander  |
92|Gastly      |
93|Haunter     |
94|Gengar      |
*/
```