POKÉMONING

a Pokémon database
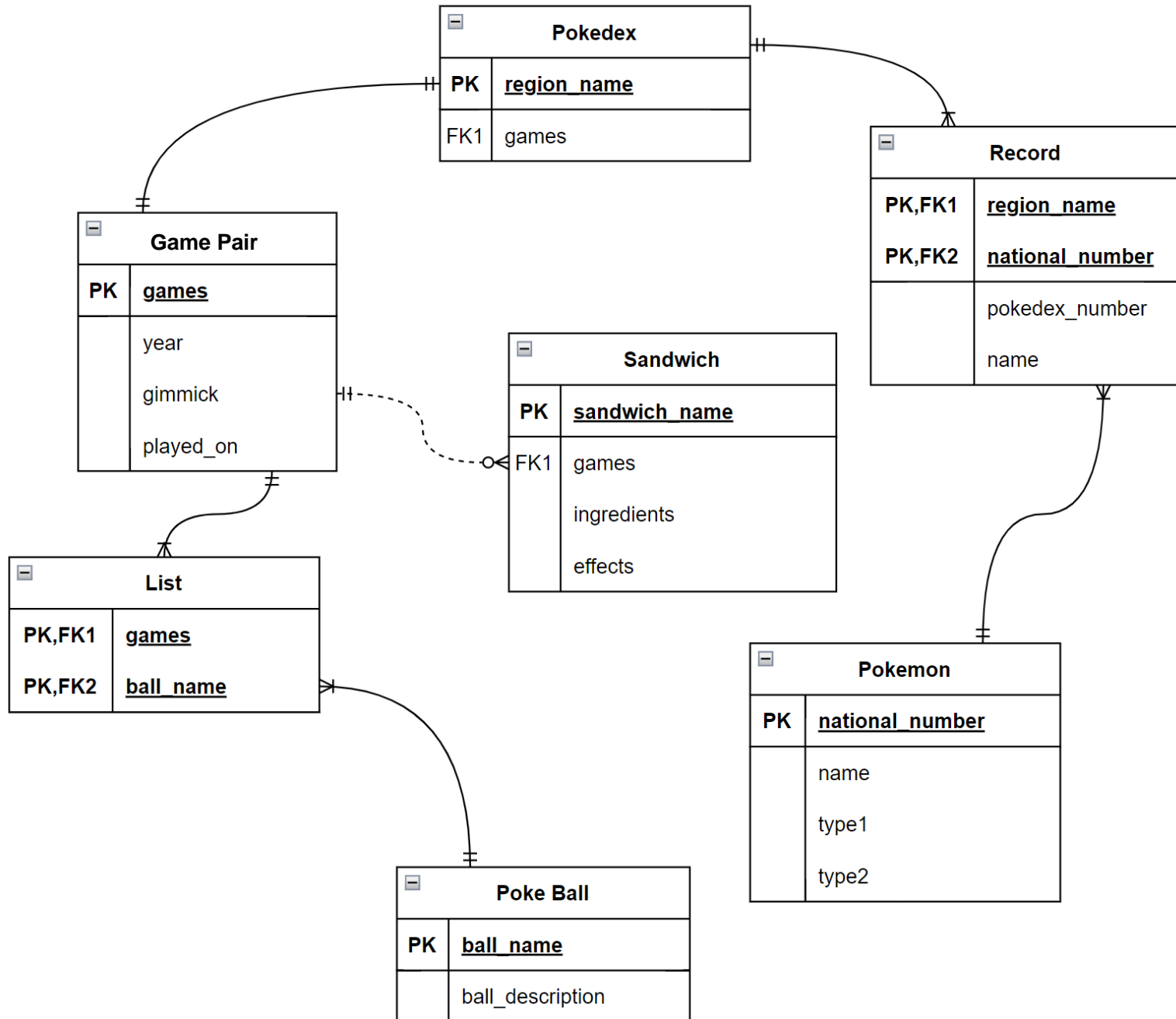
Grace Lane

Overview

What is it?

- Sword and Shield (2019)
- Brilliant Diamond and Shining Pearl (2021)
- Scarlet and Violet (2022)
  - Pokedex
  - Pokémon
  - Poke balls
  - Sandwich recipes

Milestone 1

ER Diagram

ER Diagram

Entities

- Game Pair
- Pokedex
- Record
- Pokémon
- List
- Poke Ball
- Sandwich

Milestone 2

Normalization

Normalization

- game_pair (<u>games</u>, year, gimmick, played_on)

- pokedex (<u>region_name,</u> games)

- pokemon (<u>national_number</u>, name, type1, type2)

- record (<u>region_name</u>, <u>national_number</u>, pokedex_number, name)

- pokeball (<u>ball_name</u>, ball_description)

- list (<u>year</u>, <u>ball_name</u>, games

- sandwich (<u>sandwich_name</u>, games, ingredients, effects)

- 1NF- no repeating groups, all data values are atomic, each field has a unique name, and it has a primary key.

- 2NF- in 1NF and all non primary key attributes are dependent on all parts of the primary key.

- 3NF- in 2NF and all non primary key attributes are not dependent on each other.

Milestone 3

SQL

```sql
create table game_pair (
games varchar(35) not null primary key,
year int not null,
gimmick varchar(15),
played_on varchar(15) not null
);


create table pokedex (
region_name varchar(7) not null primary key,
games varchar(35) not null,
foreign key (games) references game_pair(games) on update cascade on delete cascade
);


create table record (
region_name varchar(7) not null,
national_number int not null,
pokedex_number int not null,
name varchar(15) not null,
primary key (region_name, national_number),
foreign key (region_name) references pokedex(region_name) on update cascade on delete cascade,
foreign key (national_number) references pokemon(national_number) on update cascade on delete cascade
);


create table list (
games varchar (35) not null,
ball_name varchar(15) not null,
primary key (games, ball_name),
foreign key (games) references game_pair(games) on update cascade on delete cascade,
foreign key (ball_name) references pokeball(ball_name) on update cascade on delete cascade
);
```

```sql
create table pokemon (
national_number int not null primary key,
name varchar(15) not null,
type1 varchar(8) not null,
type2 varchar(8)
);


create table pokeball (
ball_name varchar(15) not null primary key,
ball_description varchar(170) not null
);


create table sandwich (
sandwich_name varchar(35) not null primary key,
games varchar(20) not null,
ingredients varchar(120) not null,
effects varchar(100) not null,
foreign key (games) references game_pair(games) on update cascade on delete cascade
);
```

Entity Tables

Queries

```sql
-- Query 1- inner join
-- Find all the games (games) that have Safari Ball.
select distinct g.games
from game_pair g
inner join list l
on g.games = l.games
inner join pokeball b
on b.ball_name = l.ball_name and b.ball_name like 'Safari Ball';
```

games
▶ Brilliant Diamond and Shining Pearl
Sword and Shield

```sql
-- Query 2- sum, count, avg, etc.
-- Find the number of Pokémon who have dual typing.
select count(national_number)
from pokemon
where type2 is not null;
```

count(national_number)
▶ 511

```sql
-- Query 3- subquery
-- Find the Pokemon (national_number and name) that has the smallest national number in the Paldea region.
select name, national_number
from record
where region_name = 'Paldea'
and national_number = (select min(national_number) from record where region_name = 'Paldea');
```

name	national_number
▶ Pikachu	25

Queries

```
-- Query 4- group by and having
-- Find the primary types that have more than 50 Pokemon with that primary type.
select type1, count(national_number)
from pokemon
group by type1
having count(national_number) > 50;
```

	type1	count(national_number)
▶	Grass	97
	Fire	65
	Water	133
	Bug	83
	Normal	117
	Electric	58
	Psychic	60
	Rock	57

```
-- Query 5- left outer or right outer join
-- List the Pokemon (national_number and name) in the games and how many regions they're in.
select p.national_number, p.name, count(r.region_name) as regions
from pokedex d
left outer join record r
on d.region_name = r.region_name
left outer join pokemon p
on p.national_number = r.national_number
group by p.national_number
order by p.national_number;
```

	national_number	name	regions
▶	4	Charmander	1
	5	Charmeleon	1
	6	Charizard	1
	10	Caterpie	1
	11	Metapod	1
	12	Butterfree	1
	25	Pikachu	3
	26	Raichu	3
	35	Clefairy	2
	36	Clefable	2
	37	Vulpix	1
	38	Ninetales	1
	39	Jigglypuff	1
	40	Wigglytuff	1
	41	Zubat	1
	42	Golbat	1
	43	Oddish	1

Milestone 4

Application

Techniques

- ERD for organizing the entities and their relationships

- SQL for storing all the entities and their data

- HTML for basics of the website

- CSS for styling

- PHP for connecting to the database

- Procreate for creating the logo and favicon

Problems/Difficulties

- Styling

  - Trial and error

- Learning new languages

  - CSS

  - PHP

Website Link

https://css1.seattleu.edu/~glane/website/homepage.html