Project Sprint 4

Rating National Parks

Goals for this sprint

Evaluate changing our database into BCNF, 3NF, or 4NF

- Success criteria: Demonstrate completion by showing the decomposition and/or proving that it is in BCNF, 3NF, or 4NF.
- Success criteria: explain why we chose BCNF, 3NF, or 4NF for our tables in Rating National Parks

Create new ERD from our new table configuration and evaluate them using canonical design principles

This would fulfill the course competency data modelling level 3

Incorporate inheritance and weak entity sets into database design

Why We Avoided Inheritance

1) Relational Database Principles
   - Normalization: Reduces redundancy, improves data integrity.
   - Flat Structure: Designed around simple table relationships.
2) Query Complexity
   - Complex Joins: Inheritance leads to multiple table joins, complicating queries.
   - Polymorphic Queries: Inefficient and cumbersome in SQL.
3) Data Integrity and Maintenance
   - Referential Integrity: Harder to maintain with inherited tables.
   - Transaction Complexity: Increased risk of partial updates and inconsistencies.
4) Performance Considerations
   - Indexing Challenges: More difficult with inherited structures.
   - Storage Overhead: Inefficient storage mechanisms.

Not having inheritance in our database makes the structure simpler and more maintainable by avoiding the complexities of inheritance. Allows for efficient data retrieval through optimized queries and indexing, and improves data integrity by making it easier to maintain and enforce.

Weak Data sets

What are they: Sets of data or tables that do not have a primary key of their own and rely on a foreign key to be uniquely identified.

National_parks_ratings is a weak data set

- Characteristics:
  - ***Dependent Existence:*** Each rating is linked to a specific user (user_id) and a specific park (park_id). It cannot exist without these relationships.
  - ***Composite Key:*** While it has a primary key (rating_id), the user_id and park_id are foreign keys essential for its identification and meaning.
- Explanation:
  - ***Dependence:*** national_parks_ratings relies on both Users and Parks tables. A rating must be associated with an existing user and park.
  - ***Foreign Keys:*** user_id and park_id ensure that ratings are linked correctly, enforcing referential integrity.

BCNF

In the Rating National Parks all the relations (tables) will be but into BCNF by using Decomposition to Split the tables into smaller ones that hold true to the BCNF Constraints.

BCNF was chosen to be used for the relations in this database because helps reduce data Duplications and ensures accurate data storage in the database.

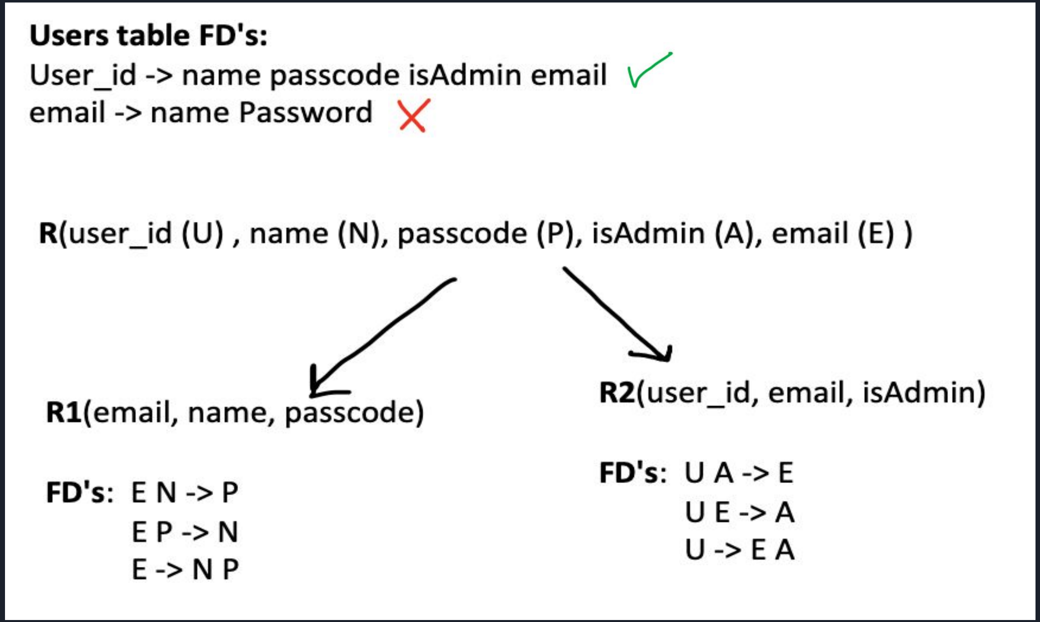
The following tables are already in BCNF:

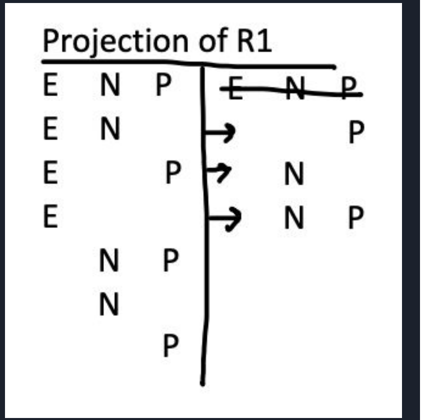
- Parks
- Weather_stations
- Ratings
- Locations

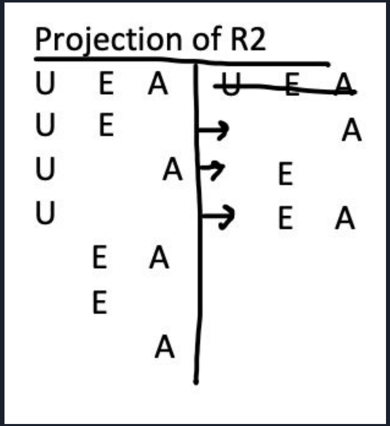
Decomposition Of Users table

In the images below, we utilized Decomposition to split the Users table into two new relations, ensuring they conform to BCNF (Boyce-Codd Normal Form).

Next, we identified the projections of the two new tables to determine their new functional dependencies (FDs).







Update User tables in Database

R1(Email, Name, Passcode)

```
CREATE TABLE User_private AS
SELECT Email, Name, Passcode
FROM Users;
```

RESULTS

Statement executed successfully

```
SELECT * FROM User_private;
```

RESULTS

Email	Name	Passcode
adam.carter@example.com	Adam Carter	AdaSter1
andrew.clark@example.com	Andrew Clark	Andiark8
ashley.hall@example.com	Ashley Hall	AshFallY
anthony.phillips@example.com	Anthony Phillips	AntripsJ
amanda.taylor@example.com	Amanda Taylor	Amajlorx
bruce.banner@example.com	Bruce Banner	Bru5nerl
benjamin.collins@example.com	Benjamin Collins	Benjins0

R2(Email, user_id, isAdmin)

```
CREATE TABLE User_Account AS
SELECT Email, User_id, isAdmin
FROM Users;
```

RESULTS

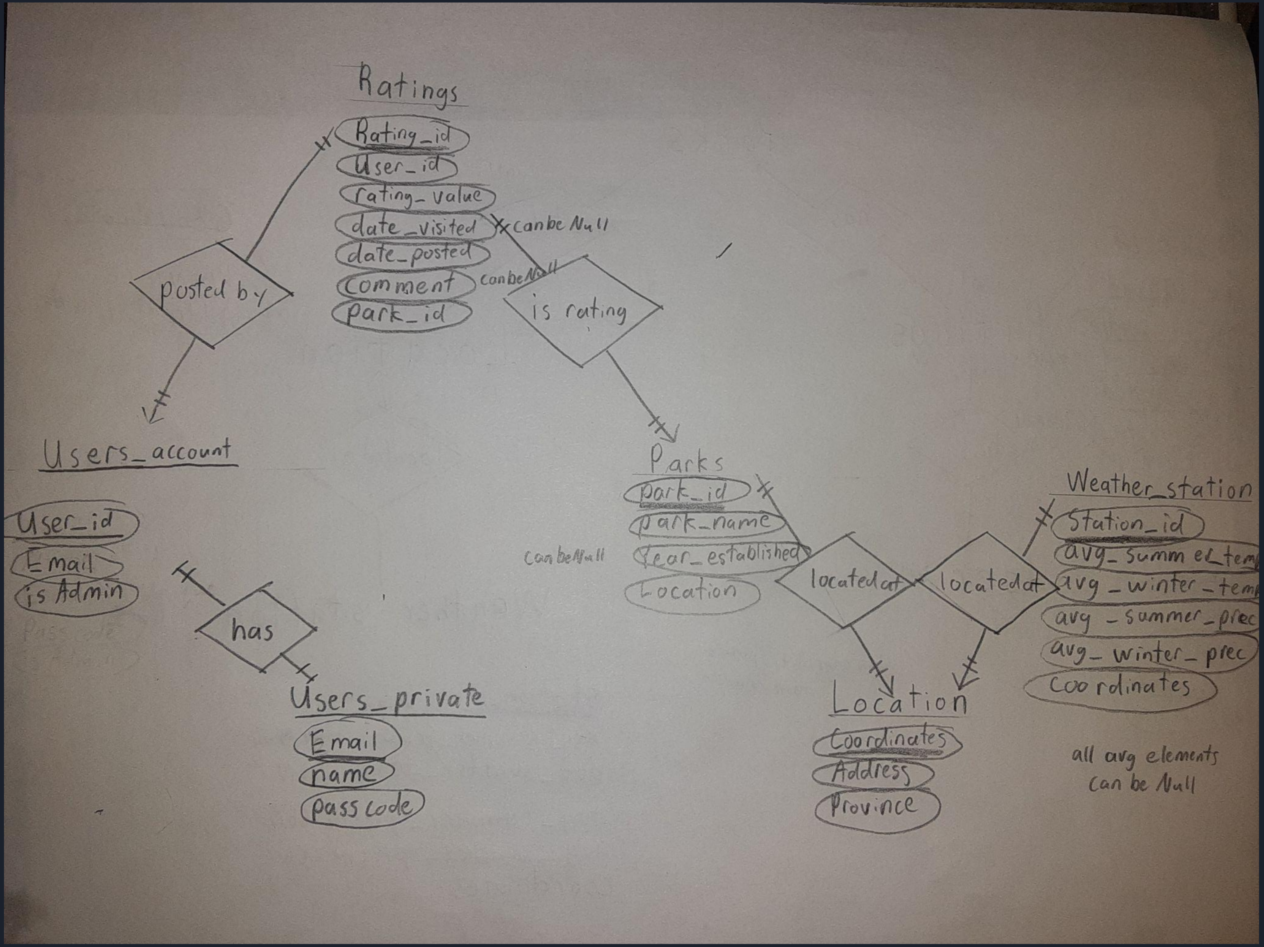
Statement executed successfully

```
SELECT * FROM User_Account;
```

RESULTS

Email	User_id	isAdmin
adam.carter@example.com	acarter42	1
andrew.clark@example.com	aclark98	1
ashley.hall@example.com	ahall60	0
anthony.phillips@example.com	aphillips89	0
amanda.taylor@example.com	ataylor27	0
bruce.banner@example.com	bbanner53	1
benjamin.collins@example.com	bcollins60	1

ERD Creation



ERD Analysis

Completeness: All features of application domain are captured

Correctness: Syntactically correct ER model with all relationships displayed

Minimality: Every aspect of the schema appears only once and is necessary for application domain

Expressiveness: Requirements are expressed and understandable without explanation

Readability: Respects aesthetic criteria for readability and consistency

Self-explanation: Annotations are expressed with the schema itself

Extensibility: Easily changed to adapt to new requirements as it is decomposed into separate modules

Normality: The relational database is in Boyce Codd normal form

Goals for next sprint

Simplify query and explain why it is better than a semantically equivalent query. Fulfills Data analytics level 3

Success criteria: created simplified queries which are easier to understand

Create a database connection and cursor in python and synchronize the python and database

Success criteria: ability to interact with database from a python app