

DOCTOR WHO DATABASE

PROJECT FOR CODE FIRST GIRLS

SILVIA ALONSO

[GITHUB.COM/DATASILVIA/DOCTOR-WHO-DATABASE](https://github.com/datasilvia/doctor-who-database)



ELECTION OF THE TOPIC

- 1.- **Rich and Expansive Universe.**- You can create numerous tables with extensive data and complex relationships among them.
- 2.- **Popular Cultural Appeal.**- Doctor Who is familiar to many people, making the project interesting and relatable.
- 3.- **Opportunity to Integrate Future Improvements.**- Starts as a small database that will grow when I add more data and tables.

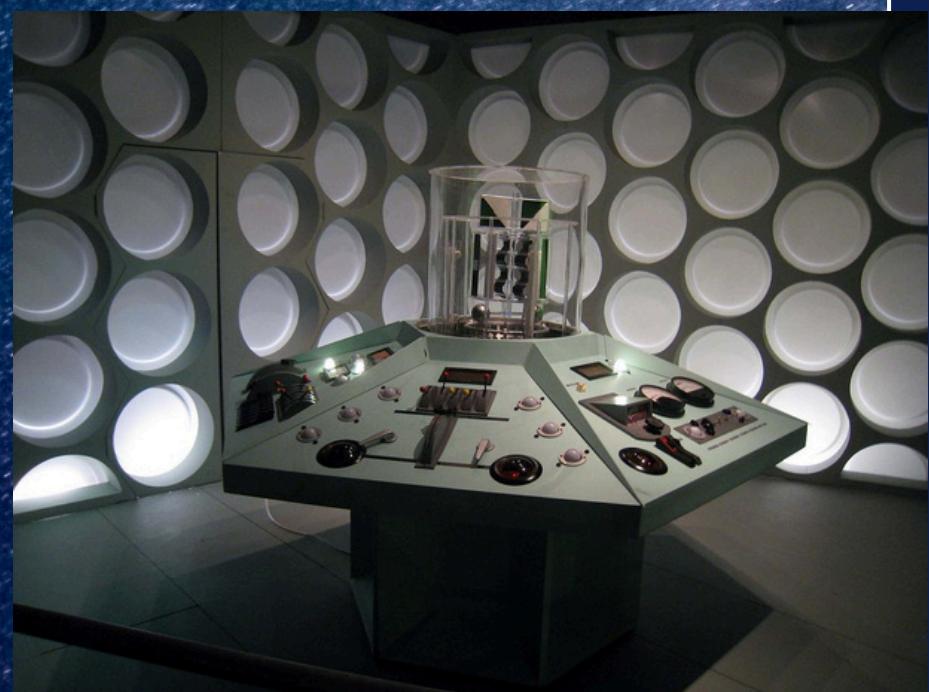


Table doctors

```
CREATE TABLE doctors(  
    id INT PRIMARY KEY,  
    name VARCHAR(50),  
    actor VARCHAR(80),  
    regeneration INT,  
    first_appearance YEAR);
```





Table companions

```
CREATE TABLE companions(  
id INT PRIMARY KEY,  
name VARCHAR(50),  
actor VARCHAR(80),  
first_appearance YEAR);
```

Table enemies

```
CREATE TABLE enemies(  
    id INT PRIMARY KEY,  
    name VARCHAR(50),  
    description VARCHAR(200),  
    first_appearance YEAR);
```





Table episodes

```
CREATE TABLE episodes(  
    id INT PRIMARY KEY,  
    name VARCHAR(200),  
    description VARCHAR(200),  
    year YEAR,  
    doctor_id INT,  
    FOREIGN KEY (doctor_id) REFERENCES doctors(id));
```



```
CREATE TABLE doctor_companion (          DOCTOR_COMPANION
    doctor_id INT,
    companion_id INT,
    PRIMARY KEY (doctor_id, companion_id),
    FOREIGN KEY (doctor_id) REFERENCES doctors(id),
    FOREIGN KEY (companion_id) REFERENCES companions(id));
```



```
CREATE TABLE episode_enemy(           EPISODE_ENEMY
    episode_id INT,
    enemy_id INT,
    PRIMARY KEY (episode_id, enemy_id),
    FOREIGN KEY (episode_id) REFERENCES episodes(id),
    FOREIGN KEY (enemy_id) REFERENCES enemies(id));
```



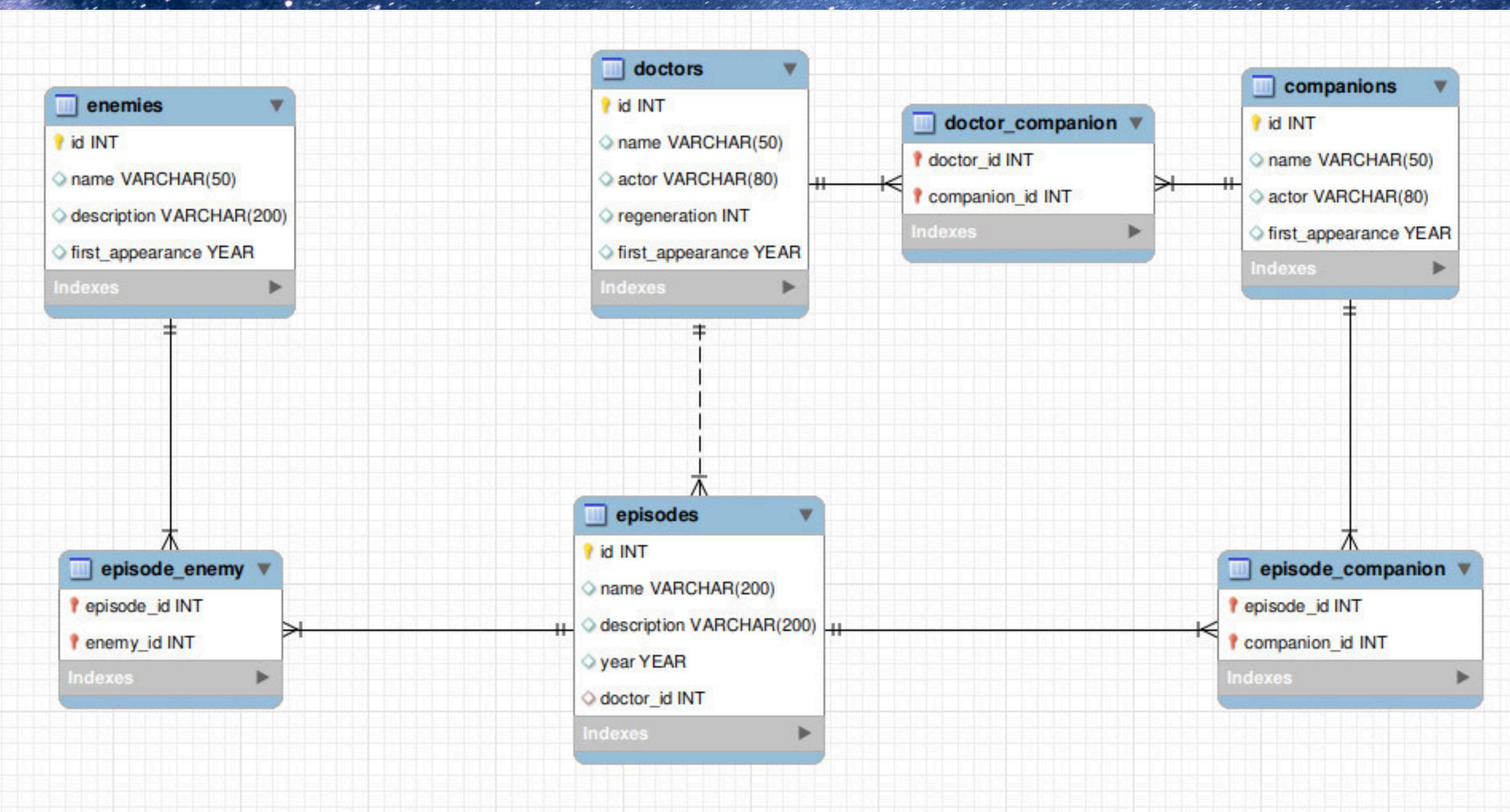
```
CREATE TABLE episode_companion (        EPISODE_COMPANION
    episode_id INT,
    companion_id INT,
    PRIMARY KEY (episode_id, companion_id),
    FOREIGN KEY (episode_id) REFERENCES episodes(id),
    FOREIGN KEY (companion_id) REFERENCES companions(id))
```



CREATING THE VIEW

```
CREATE VIEW episode_details AS
SELECT
    e.id AS episode_id,
    e.name AS episode_name,
    e.description AS episode_description,
    e.year AS episode_year,
    d.name AS doctor_name,
    c.name AS companion_name,
    en.name AS enemy_name
FROM
    episodes e
JOIN
    doctors d ON e.doctor_id = d.id
LEFT JOIN
    episode_companion ec ON e.id = ec.episode_id
LEFT JOIN
    companions c ON ec.companion_id = c.id
LEFT JOIN
    episode_enemy ee ON e.id = ee.episode_id
LEFT JOIN
    enemies en ON ee.enemy_id = en.id;
```

Entity Relationship Diagram (ERD)



FUTURE IMPROVEMENTS

ADD SEASONS AND SERIES DATA

Organize episodes by seasons and series to facilitate season-based analyses.

ADD LOCATIONS

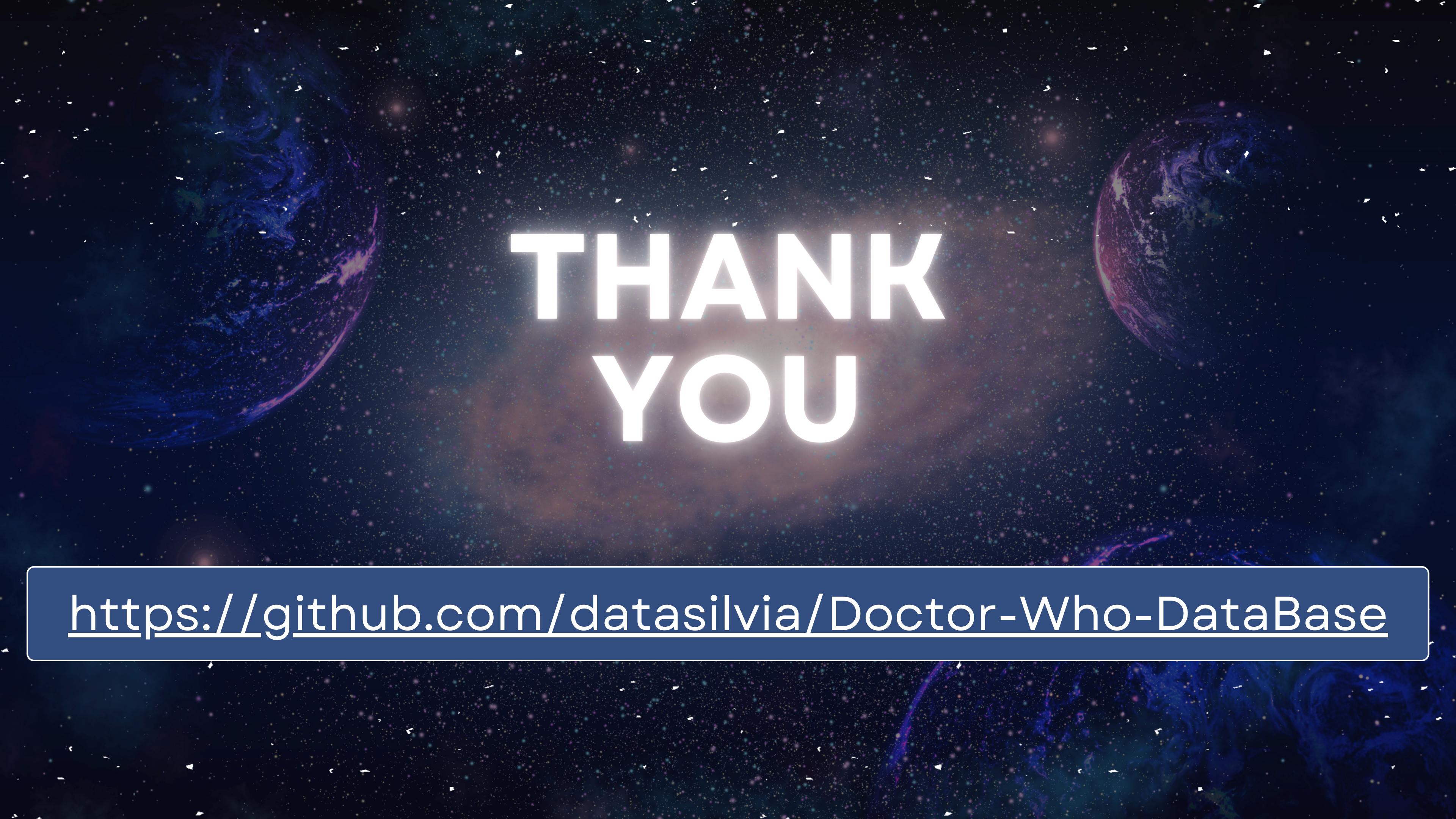
Add a table for locations to see where the Doctor's adventures take place.

REGENERATION EVENTS

Detail each Doctor's regeneration circumstances for continuity analysis.

IMPLEMENT DATA INTEGRITY TRIGGERS

Add triggers to ensure data consistency when adding or updating records.



THANK
YOU

<https://github.com/datasilvia/Doctor-Who-DataBase>