Warriors Project - Outline

Haleigh Novak

My database content is the 'cast' of the Warriors book series. It is a categorized list of the different characters, it contains the current important information about every character (even deceased Lives_Left = 0). Information including mother and fathers name, their skill level, their home clan, etc. I felt that the cast of Warriors can get a little bit confusing because there is so much information to remember, which is not the best unless you get really into the books. So this database simplifies the need for almost having to keep your own list by keeping up on all the current information of the cats and their clans in a searchable database. So when you come across a character that hasn't been brought up in a while then you don't have to go back three books to remember the character you can just search the database.

Warriors Project - Database Outline

Haleigh Novak

The Characters table contains all the 'personal' information about each character, containing; the characters parents names (if known), the characters name, age (measured in moons), amount of lives left (9-0), and each character is given an auto-incrementing ID for data relations between tables. The character table is related directly to three other tables: each character must have one skill level (relation: Character_Skills), must live in only one clan (relation: Clans), can have no certifications or several certifications based on their skill level (relation: Character_Certifications), and can have no job or several jobs based on what certifications they possess (relation: Character_Jobs). The primary key is Character_ID.

The Clans table contains the information about each clan [family group]. Each clan must have a name and a population. Each one is also given an auto-incrementing ID for data relations between tables. The primary key is the Clan_ID.

The Jobs table contains all jobs available to the cats. Each job requires at least one certification (relation: Certifications), and must have a title. Each job is also given an auto-incrementing ID for data relations between tables. The primary key is the Job_ID.

The Character_Jobs table contains a list of all character and job pairings using Character_ID (relation: Characters) and Job_ID (relation: Jobs). The primary key is a combination of the two id numbers.

The Certifications table contains all certifications available to the cats. Each certification must have a title and a skill level required to have the certification (relation: Skill_Levels). Each certification also has an auto-incrementing ID for data relations between tables. The primary key is a combination of the character id and the Certification_ID.

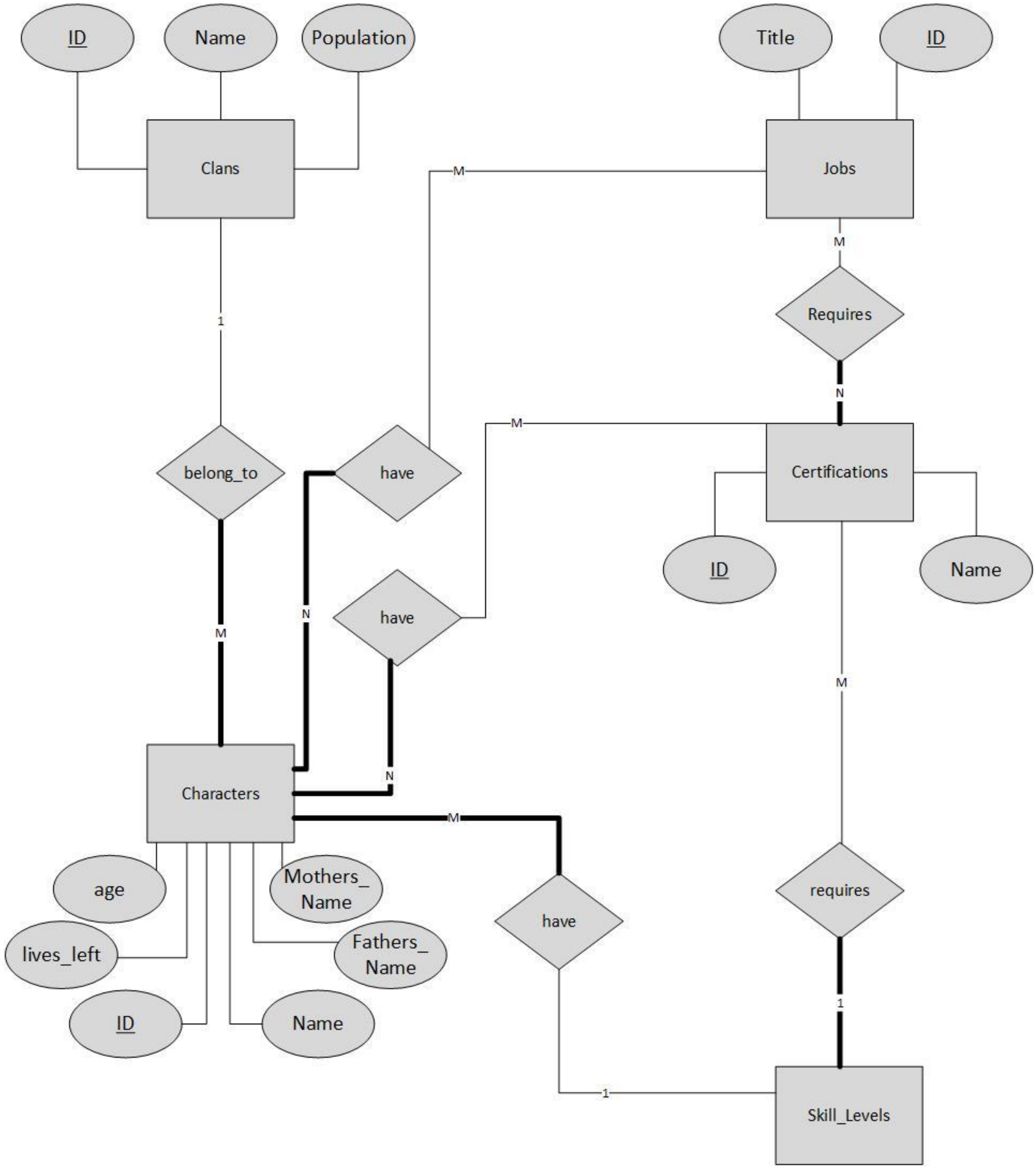
The Character_Certification table contains a list of all character and certification pairings using Character_ID (relation: Characters) and Certification_ID (relation: Certifications). The primary key is a combination of the two id numbers.

The Skills table contains all of the different skill levels [based on the level of training a cat has gotten thus far]. Each skill level must have a title and is given an auto-incrementing ID for data relations between tables. The primary key is the Skill_Level_ID.

The Character_Skills table contains a list of all character and skill pairings using Character_ID (relation: Characters) and Skill_ID (relation: Skill_Levels). The primary key is a combination of the two id numbers.

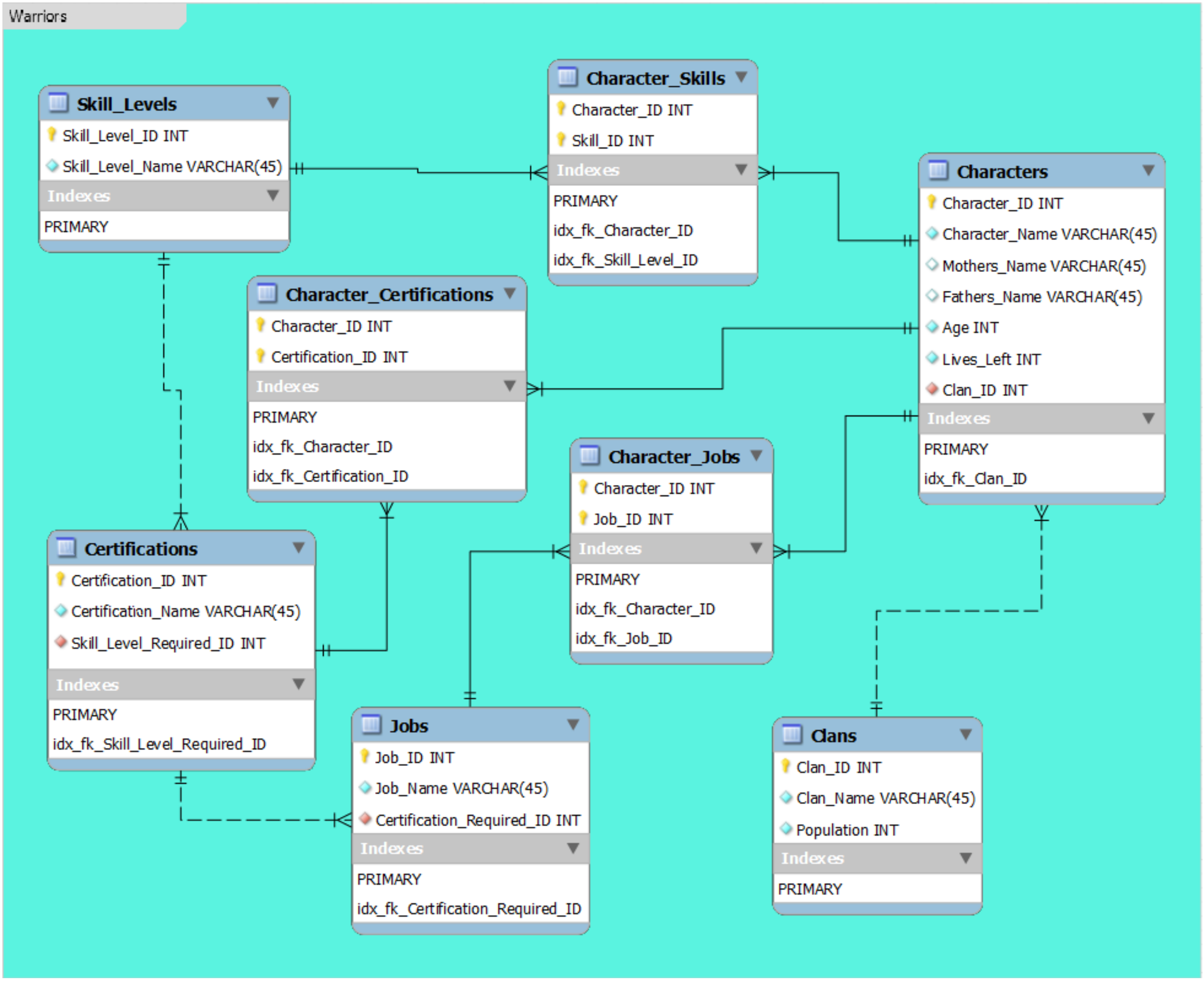
Warriors Project - ER Diagram

Haleigh Novak



Warriors Project - Database Schema

Haleigh Novak



Warriors Project - Table Creation Queries

Haleigh Novak

```
SET FOREIGN_KEY_CHECKS=0;
DROP TABLE IF EXISTS `Warriors_Characters`;
DROP TABLE IF EXISTS `Warriors_Clans`;
DROP TABLE IF EXISTS `Warriors_Skill_Levels`;
DROP TABLE IF EXISTS `Warriors_Character_Skills`;
DROP TABLE IF EXISTS `Warriors_Certifications`;
DROP TABLE IF EXISTS `Warriors_Character_Certifications`;
DROP TABLE IF EXISTS `Warriors_Jobs`;
DROP TABLE IF EXISTS `Warriors_Character_Jobs`;

CREATE TABLE Warriors_Characters (
  Character_ID INT NOT NULL AUTO_INCREMENT,
  Character_Name VARCHAR(45) NOT NULL,
  Age INT NOT NULL,
  Clan_ID INT NOT NULL,
  Mothers_Name VARCHAR(45) DEFAULT NULL,
  Fathers_Name VARCHAR(45) DEFAULT NULL,
  Lives_Left INT NOT NULL,
  PRIMARY KEY (Character_ID),
  FOREIGN KEY (Clan_ID) REFERENCES Warriors_Clans (Clan_ID)
  )ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE Warriors_Clans (
  Clan_ID INT NOT NULL AUTO_INCREMENT,
  Clan_Name VARCHAR(45) NOT NULL,
  Population INT NOT NULL,
  PRIMARY KEY (Clan_ID)
  )ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE Warriors_Skill_Levels (
  Skill_Level_ID INT NOT NULL AUTO_INCREMENT,
  Skill_Level_Name VARCHAR(45) NOT NULL,
  PRIMARY KEY (Skill_Level_ID)
  )ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE Warriors_Character_Skills (
  Character_ID INT NOT NULL,
  Skill_Level_ID INT NOT NULL,
  PRIMARY KEY (Character_ID,Skill_Level_ID),
  FOREIGN KEY (Character_ID) REFERENCES Warriors_Characters (Character_ID),
  FOREIGN KEY (Skill_Level_ID) REFERENCES Warriors_Skill_Levels (Skill_Level_ID)
  )ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE Warriors_Certifications (
  Certification_ID INT NOT NULL AUTO_INCREMENT,
  Certification_Name VARCHAR(45) NOT NULL,
  Skill_Level_Required_ID INT NOT NULL,
  PRIMARY KEY(Certification_ID),
  FOREIGN KEY (Skill_Level_Required_ID) REFERENCES Warriors_Skill_Levels (Skill_Level_ID)
  )ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE Warriors_Character_Certifications (
  Character_ID INT NOT NULL,
  Certification_ID INT NOT NULL,
  PRIMARY KEY (Character_ID,Certification_ID),
  FOREIGN KEY (Character_ID) REFERENCES Warriors_Characters (Character_ID),
  FOREIGN KEY (Certification_ID) REFERENCES Warriors_Certifications (Certification_ID)
  )ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE Warriors_Jobs (
  Job_ID INT NOT NULL AUTO_INCREMENT,
  Job_Name VARCHAR(45) NOT NULL,
  Certification_Required_ID INT NOT NULL,
  PRIMARY KEY(Job_ID),
  FOREIGN KEY (Certification_Required_ID) REFERENCES Warriors_Certifications(Certification_ID)
  )ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE Warriors_Character_Jobs (
  Character_ID INT NOT NULL,
  Job_ID INT NOT NULL,
  PRIMARY KEY (Character_ID,Job_ID),
  FOREIGN KEY (Character_ID) REFERENCES Warriors_Characters (Character_ID),
  FOREIGN KEY (Job_ID) REFERENCES Warriors_Jobs (Job_ID)
  )ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Warriors Project - General Use Queries

Haleigh Novak

Query: Insert

```
   INSERT INTO Warriors_Clans (Clan_Name,Population) VALUES ([name], [population]);
   INSERT INTO Warriors_Skill_Levels (Skill_Level_Name) VALUES ([name]);
   INSERT INTO Warriors_Certifications (Certification_Name,Skill_Level_Required_ID) VALUES ([certification name], (SELECT Skill_Level_ID FROM Warriors_Skill_Levels WHERE Skill_Level_Name = [skill name]));
   INSERT INTO Warriors_Jobs (Job_Name,Certification_Required_ID) VALUES ([job name],(SELECT Certification_ID FROM Warriors_Certifications WHERE Certification_Name = [certification name]));
   INSERT INTO Warriors_Characters (Character_Name,Age,Clan_ID,Mothers_Name,Fathers_Name,Lives_Left) VALUES ([character name],[age],(SELECT Clan_ID FROM Warriors_Clans WHERE Clan_Name = [clan name]),[moms name],[dads name],[lives left]);
  INSERT INTO Warriors_Character_Skills (Character_ID,Skill_Level_ID) VALUES ((SELECT Character_ID FROM Warriors_Characters WHERE Character_Name = [character name]),(SELECT Skill_Level_ID FROM Warriors_Skill_Levels WHERE Skill_Level_Name = [skill name]));
   INSERT INTO Warriors_Character_Certifications (Character_ID,Certification_ID) VALUES ((SELECT Character_ID FROM Warriors_Characters WHERE Character_Name = [character name]),(SELECT Certification_ID FROM Warriors_Certifications WHERE Certification_Name = [certification name]));
   INSERT INTO Warriors_Character_Jobs (Character_ID,Job_ID) VALUES ((SELECT Character_ID FROM Warriors_Characters WHERE Character_Name = [character name]),(SELECT Job_ID FROM Warriors_Jobs INNER JOIN Warriors_Certifications ON Certification_Required_ID = Certification_ID WHERE Job_Name = [job name] and Certification_Name = [certification name]));
```

Query: Select

```
   SELECT Clan_ID FROM Warriors_Clans WHERE Clan_Name = [clan name];
   SELECT Clan_ID FROM Warriors_Clans WHERE Population {<,>,=] [clan population];
   SELECT Character_ID FROM Warriors_Characters WHERE Character_Name = [character name];
   SELECT Character_ID FROM Warriors_Characters WHERE Age [<,>,=] [character age];
   SELECT Character_ID FROM Warriors_Characters WHERE Clan_ID = (SELECT Clan_ID FROM Warriors_Clans WHERE Clan_Name = [clan name]);
   SELECT Skill_Level_ID FROM Warriors_Skill_Levels WHERE Skill_Level_Name = [skill name];
   SELECT Certification_ID FROM Warriors_Certifications WHERE Certification_Name = [certification name];
   SELECT Job_ID FROM Warriors_Jobs WHERE Job_Name = [job name];
   SELECT Charatcer_ID FROM Warriors_Character_[Skills, Certifications, Jobs] WHERE [Skill_Level_ID, Certification_ID, Job_ID = (SELECT [Skill_Level, Certification, Job]_ID FROM Warriors_[Skill_Levels, Certifications, Jobs] WHERE [Skill_Level, Certification, Job]_Name = [name]);
```