What went well?

I was able to successfully build a Command Line Interface (CLI) that allows users to see various types of information about video games and characters. Using Ruby and Active Record, I migrated several tables to a database, seeded data into those tables, and used different methods and associations to output the results.

What could have been improved?

I encountered the following difficulties:

* **Seeding information** – initially, I was unaware that the Faker gem in Ruby only outputted data when a specific term was used. I repeatedly encountered errors trying to seed data for some parts of the database, given that they were not part of the Faker library.
* **Setting up the project** – I faced a few challenges while setting up the various files needed for the project. Specifically, using ‘require’ and connecting to my database took a bit of time to get used to, as I sometimes found that I was using these different functionalities in the wrong ruby file.
* **Formatting tables** – when outputting large tables such as character of video game information, the formatting was all out of place, which required some independent research to fix.

Key takeaways

The key takeaways from this project are:

* The importance of familiarising myself with the different gems and understanding how to properly set up projects that so that gems can be used, and databases can be linked.
* The importance of understanding the associations between different classes, which makes writing methods far more efficient.

Process

To start, I set up various files in Ruby that were needed to successfully create a CLI using Ruby. These files included ‘environment.rb’, ‘Gemfile’, ‘Rakefile’, as well as using the ‘require’ method to link the relevant gems, classes, files, and database. I decided on a group of classes that were connected to each other and could make use of the ‘belongs\_to’ and ‘has\_many’ methods. I set up ‘Game’, ‘Character’, and ‘Genre’ classes, as well as a ‘Game App’ class to run the CLI from. I built out the structure of the CLI and wrote several methods that could display relevant information to users. I spent time formatting the CLI to ensure that it was user friendly and easily readable.