|  |  |
| --- | --- |
| Project Title | Kickstarter Video Games Success and Popularity in Steam |
| Team | Peter Witwer |
| Project Description | Use several datasets from Kaggle to practice using ETL with.  The goal will be to end with a database that has detailed information for each game that started on kickstarter, made it to steam, and then how successful it was after making it to steam |
| Relational or non-relational database? | Relational –pgadmin4 |
| Datasets | * kickstarter csv dataset * kickstarter games that made it to steam csv * Steam csv dataset that includes a variety of interesting columns for each game * Potentially other datasets depending on time constraint? |
| Concerns | * Does this meet project requirements? * Should I narrow/expand the scope of my project? (I am working alone) * Is there anything else I need to be considering here? This project seems deceptively simple. We’re not looking for anything special with the data, just practicing using ETL, correct? |

Kickstarter Video Games Success and Popularity in Steam

**ETL-Project**

My goal here is to track stats for every videogame that is crowdfunded on kickstarter and then released on Steam. I would like to expand it to include other platforms besides Steam but I don't know if I will get there.

I have gathered several datasets downloaded from kaggle and adjusted them to show relevant information for the games. I would really like to add a dataset that shows the sales of each game, so that the relative success of a game could be measured in more than one way, but I haven't gotten there yet.

It took a long time for me to figure out all the bits and pieces that I needed to scrape the game titles from kickstarter/steam. My first couple of tries to navigate through the page using splinter to hit the "view more" button were fruitless. Eventually I realized I could actually pull the url from the "view more" button which leads to a 'page 2' of the list of games and I could continue to pull the url for the next page until I reached the final page. So I finally knew how to iterate through all of the games kickstarter listed when I realized that the actual project pages hosted on the kickstarter website were wildly inconsistent. I took a break and then checked to see if the steam store had a more consistent layout for each game, because I really needed an html element that I could find reliably and as I expected the steam store has an identical layout for every game.

Finally I had everything I needed to scrape these webpages. I created a while-loop that would only break when the "view more" element on the kickstarter page was no longer visible. Inside that I created a for loop that would iterate through every steam store url on the given page and save the urls in a list. I then used a for-loop to go through every steam link that I had collected in order to extract the game titles. I used try and except for this, and noticed that a few of the steam links were faulty or required the user to authenticate their age.

The transforming process involved dropping a lot of dropping unneccesary columns from the kaggle datasets and shrinking them down to match the few titles they shared with the collection of game titles that I scraped from Kickstarter. The ign and the metacritic datasets had many duplicate values for each individual platform that needed to be removed, and I noticed that the kickstarter dataset had many instances where games were listed multiple times with exactly or almost exactly the same values in each row.

The loading process into PGadmin was relatively smooth as the practice I did for that homework assignment was still fresh in my brain, so creating tables and using the correct datatypes didn't take very much time. Unfortunately I did run into an issue trying to import my metacritic csv into the database and I have yet to solve that particular issue.

While I'm satisfied with the practice with webscraping and sql that I did in this project I had hoped for a better finished product. I was dissapointed with the relatively small amount of overlap between the game titles that I scraped from the kickstarter/steam store webpages and the datasets that I downloaded from kaggle. I would like to continue to add to this project because now that I feel much more confident web scraping I could create my own datasets for all of the game titles which would make the resulting dataset a lot more interesting and relevant. I fully intend to continue adding to this dataset once I have finished a basic outline that can be graded, and hopefully I can continue to polish the finished product until it is something I would be proud to show a potential employer.