Project Update – 08/07/2021

# To do’s

* Create a database schema and review the data needed in each table.
* Complete Panel data collection (code development, testing, and running)
* Prepare person data for gender prediction.
* Start reviewing and cleaning the data.

# What you did and created?

* Split the functionality of the get\_panel\_data() function into a sweet function containing the main functionality and a wrapper function that handles errors and general logging functionality.
* Removed error handling functionality from the log\_bind\_row() function to improve debugging and change error handling behaviour.
* Found and resolved an issue with panel\_id regex; typo in the str\_match() function.
* Solved a logging issued caused but an extra comma at the end of function calls clashing with function and variable calls in the.
* Expanded functionality of extract\_tables() and extract\_elements() to better handle missing elements and not error. Logging was made clears as a result.
* Successfully tested a single link run of sweet\_get\_panel\_data() and get\_panel\_data() [wrapper function]
* Add conditional running functionality to log\_bind\_row() testing if elements were found in the extract functions.
* Resolved a bind\_rows error where columns of same names but different datatypes were being bound together. Coerced all columns to characters for conversion to more appropriate datatypes later.
* Resolved an issue where the list of failed links was being overwritten.
* Investigated issue with #dgSiftByNum CSS elements not being picked up; css selector had a typo.
* Set up and ran code over all panel links.
* Created database schemas for the data, one optimised and one showing the dataset in current form.
* Saved all the data collected on panels.

# What decision have you made and how they were made?

* Decided to create a wrapper function and a sweet function for the get panel data. This made the functionality easier to read, maintain, and test.
* Decided to remove the error handling functionality from some lower level functions. This is so the errors would be caught better and easier to review.
* Decided on coercing all the data to character types. This resolved an error easily and can be reversed down the line when cleaning the data.

# Relevant References

|  |  |  |
| --- | --- | --- |
| **Name** | **Link** | **Notes** |
| Advanced R | http://adv-r.had.co.nz/Exceptions-Debugging.html#condition-handling | Used to develop error handling functionality |
| Stack Overflow: Advanced TryCatch statements | <https://stackoverflow.com/questions/12193779/how-to-write-trycatch-in-r> | Used to develop complex error handling functionality |
| GtR-1-API-v3.3 | <https://gtr.ukri.org/resources/GtR-1-API-v3.3.pdf> | Used to develop API access expressions |
| Progress Package Readme | https://github.com/r-lib/progress#readme | Used vinettes/tutorial to develop progress bar functionality |
| Rvest | https://rvest.tidyverse.org/ | Used to develop scaping from HTML documents functionality |
| JSONLITE | <https://cran.r-project.org/web/packages/jsonlite/vignettes/json-aaquickstart.html> | Used to request and convert JSON files into dataframes |
| Subset list based on a filter | https://stackoverflow.com/questions/6941506/subset-elements-in-a-list-based-on-a-logical-condition | Used to filter the elements of a list based on a condition and explain the use of the Filter function |
| rvest Error in open.connection(x, “rb”) | https://stackoverflow.com/questions/33295686/rvest-error-in-open-connectionx-rb-timeout-was-reached | Used to solve error with html document retreval and develop more stable method. |
| [How to convert variable (object) name into String](https://stackoverflow.com/questions/14577412/how-to-convert-variable-object-name-into-string) | https://stackoverflow.com/questions/14577412/how-to-convert-variable-object-name-into-string | Used in logging functionality to return the variable name parsed into a function |
| Regex101 | https://regex101.com/ | Used to develop regex expressions |
| DrawSQL | <https://drawsql.app/> | Used to create database schema |

# To do items for coming week

* Complete panel ranked list data collection (code development, testing, and running)
* Prepare person data for gender prediction.
* Start reviewing and cleaning the data.
* Finalise database schema.
* Start organising data by the schema.