

Analysis Class Design Document

- Developer: Swathi Danturi
- Date: 11/11/2024

freq_of_lang() Method Design

- define a new method in `analysis.py` called `freq_of_lang`
- `self` is the current instance of the class
- `self.devs` is the instance variable of the class
 - the elements of the list are dictionaries
 - each dictionary corresponds to a row in the text file
- define an accumulator `lang_freq` and initialize it to an empty dictionary
- iterate through each dictionary of the list `self.devs` using the iterable `dev_data`
- at each iteration:
 - get the `LanguageAdmired` string separated by `;` from `dev_data` using the same as key
 - split the above using `.split()` with `;` as delimited
 - assign the above list to `languages_admired`
 - iterate through each value `language` of the `languages_admired` using `for` loop
 - at each iteration:
 - check `language` should not equal `NA` using `if`:
 - `if` the `language` is not already in `lang_freq`:
 - initialize `lang_freq` of `language` to zero
 - increment the count of `lang_freq` of `language` by 1
- return `lang_freq`

lang_histogram() Method Design

- define a new method in `analysis.py` called `lang_histogram`
- `self` is the current instance of the class
- define an accumulator `lang_histo` and initialize it to an empty dictionary
- call `freq_of_lang` method using the current class instance `self`
- assign the return value of it to the variable `lang_freq`
- `lang_freq` contains key-value pairs of language-frequency
- using `for` loop iterate through the items of `lang_freq` with `lang, freq` as iterators
- at each iteration:
 - check if `freq` is in `lang_histo`
 - if not, then initialize `freq` of `lang_histo` to an empty list
 - append `lang` to `lang_histo` at the key `freq`
- return `lang_histo`

write_lang_popularity() Method Design

- define a new method in `analysis.py` called `write_lang_popularity`
- `self` is the current instance of the class
- call `lang_histogram` method using the current class instance `self`
- assign the return value of it to the variable `lang_histogram`

- open a new file `devopswrite.csv` using `with` in `write` mode with file variable `new_csv_file`
- define a local variable called `csv_writer`
- assign a new csv writer using `csv.writer` method on `new_csv_file` to `csv_writer`
- write the headers `[Frequency, Languages]` into the file using `writerow` method with `csv_writer`
- using `for` loop iterate through the items of `lang_histogram` with `freq`, `langs` as iterators
- at each iteration:
 - join the languages in `langs` using `join()` with `;` as delimiter
 - use `writerow` on `csv_writer` to write the row `[freq, langs]` into the file
- close the file `new_csv_file` using `close()`