Census_Language_Data

Jill Davis

3/25/2025

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

Voting Rights Act Minority Language by Jurisdiction

This document converts the table from https://www.federalregister.gov/documents/full_text/xml/2021/12/08/2021-26547.xml into a static data frame for analysis.

After converting the table to a data frame, this program takes just the data for Illinois and converts it into an Excel (.xlsx) file. However, the data frame can be filtered to any state using the same method.

Load required packages

```
#install.packages("xml2")
library(xml2)
```

Read in the data

```
# Read the XML file
xml_file <- read_xml("/Users/jill/Downloads/census_xml_file_test.xml")
```

Extract relevant xml nodes

```
# We want to extract all ROW nodes
rows <- xml_find_all(xml_file, "//ROW")
```

Create vectors to prepare data frame

```
# Initialize vectors
state_list <- c()
subdivision_list <- c()
language_list <- c()
current_state <- NA # Not all ROW nodes have a state listed, so we will set the value for current_state as NA to start
```

Clean the data by appending missing values to each node

```
# Iterate through each ROW node
for (row in rows) {
  # Check if the row contains a state (ENT with I="22") and update current_state
  state_node <- xml_find_first(row, "ENT[@I='22']")
  if (!is.na(state_node)) {
    current_state <- xml_text(state_node)
  }

  # Extract subdivision names (ENT with I="03")
  subdivision_nodes <- xml_find_all(row, "ENT[@I='03']")

  # Extract only the SECOND `<ENT>` as language (if it exists)
  language_node <- xml_find_all(row, "ENT[position()=2]")

  # Pair each subdivision with its corresponding language(s)
  for (sub_node in subdivision_nodes) {
    subdivision_name <- xml_text(sub_node)

    if (length(language_node) > 0) {  # Ensure there is a second ENT element
      language <- xml_text(language_node)
    } else {
      language <- NA  # Assign NA if no valid language found
    }

    # Append to lists
    state_list <- c(state_list, current_state)
    subdivision_list <- c(subdivision_list, subdivision_name)
    language_list <- c(language_list, language)
  }
}
```

Create a data frame from the vectors

```
# Create a data frame
df <- data.frame(State = state_list, Subdivision = subdivision_list, Language = language_list, stringsAsFactors = FALSE)

# Print first few rows
head(df)
```

```
##     State                Subdivision  Language
## 1 Alaska:     Aleutians East Borough   Yup'ik.
## 2 Alaska: Aleutians West Census Area Filipino.
## 3 Alaska:         Bethel Census Area   Yup'ik.
## 4 Alaska:        Bristol Bay Borough   Yup'ik.
## 5 Alaska:     Dillingham Census Area   Yup'ik.
## 6 Alaska:    Kenai Peninsula Borough   Yup'ik.
```

Filter by State

```
texas_df = df[df$State == "Texas:", ]
head(texas_df)
```

```
##      State     Subdivision  Language
## 239 Texas:  State Coverage Hispanic.
## 240 Texas:  Andrews County Hispanic.
## 241 Texas: Atascosa County Hispanic.
## 242 Texas:   Bailey County Hispanic.
## 243 Texas:      Bee County Hispanic.
## 244 Texas:    Bexar County Hispanic.
```

```
illinois_df = df[df$State == "Illinois:", ]
head(illinois_df)
```

```
##         State   Subdivision                       Language
## 126 Illinois:   Cook County                       Hispanic.
## 127 Illinois:   Cook County Asian Indian (including Sikh).
## 128 Illinois:   Cook County Chinese (including Taiwanese).
## 129 Illinois: DuPage County                       Hispanic.
## 130 Illinois:   Kane County                       Hispanic.
## 131 Illinois:   Lake County                       Hispanic.
```

Load to an Excel file

```
#install.packages("writexl")
library(writexl)
write_xlsx(illinois_df, "/Users/jill/Downloads/illinois_languages_census.xlsx")
```