```
# ========================================
# Script:  Download and process World Bank indicators
# Author:  Paulina Skurzak
# Purpose: Download Health, Poverty, Education, Gender, Agriculture & Rural Development,
#          and Aid Effectiveness indicators for all countries
#          from 2000–2025, save wide-format and metadata CSVs.
#          Intended for use by World Pediatrics / Schroeder Center.
#========================================

#=========================
# 1.Load required libraries
#=========================
library(readxl)
library(wbstats)
library(tidyverse)
library(stringdist)

#=========================
# 2. Set working Directory
#=========================
setwd("~/Documents/GRI Research Fellow")

#=========================
# 3. Read indicator information from Excel
#=========================
health_data = read_excel("Data_raw/World_Bank_Indicator_info.xlsx", sheet = "Health")
poverty_data = read_excel("Data_raw/World_Bank_Indicator_info.xlsx", sheet = "Poverty")
education_data = read_excel("Data_raw/World_Bank_Indicator_info.xlsx", sheet =
"Education")
gender_data = read_excel("Data_raw/World_Bank_Indicator_info.xlsx", sheet = "Gender")
agriculture_data = read_excel("Data_raw/World_Bank_Indicator_info.xlsx", sheet =
"Agriculture & Rural Development")
aid_data = read_excel("Data_raw/World_Bank_Indicator_info.xlsx", sheet = "Aid
Effectiveness")

#=========================
# 4. Extract indicator IDs from URLs
#=========================
health_data = health_data %>%
  mutate(id = sub(".*/indicator/", "", myurl))
poverty_data = poverty_data %>%
  mutate(id = sub(".*/indicator/", "", myurl))
education_data = education_data %>%
  mutate(id = sub(".*/indicator/", "", myurl))
gender_data = gender_data %>%
  mutate(id = sub(".*/indicator/", "", myurl))
agriculture_data = agriculture_data %>%
  mutate(id = sub(".*/indicator/", "", myurl))
aid_data = aid_data %>%
  mutate(id = sub(".*/indicator/", "", myurl))

#=========================
# 5. Get list of countries from World Bank
#=========================
countries = wb_countries()

#=========================
# 6. Function to download indicator data
#=========================
download_indicator_data = function(ids, names, country_list) {
  all_data = tibble()

  for (i in seq_along(ids)) {
    id = ids[i]
```

```
    name = names[i]

    message("Downloading: ", name, " (", id, ")")

    dat = tryCatch({
      wb_data(indicator = id, start_date = 2000, end_date = 2025, return_wide = FALSE)
    }, error = function(e) {
      message("Failed: ", name)
      return(NULL)
    })

    if (!is.null(dat) && nrow(dat) > 0) {
      cleaned = dat %>%
        filter(iso3c %in% country_list$iso3c) %>%
        select(iso2c, iso3c, country, date, value, indicator_id = indicator) %>%
        mutate(indicator_id = id)

      all_data = bind_rows(all_data, cleaned)
    }
  }

  return(all_data)
}

#=========================
# 7. Download raw data for each category
#=========================
health_raw = download_indicator_data(health_data$id, health_data$Health, countries)
poverty_raw = download_indicator_data(poverty_data$id, poverty_data$Poverty, countries)
education_raw = download_indicator_data(education_data$id, education_data$Education,
countries)
gender_raw = download_indicator_data(gender_data$id, gender_data$Gender, countries)
agriculture_raw = download_indicator_data(agriculture_data$id,
agriculture_data$`Agriculture & Rural Development`, countries)
aid_raw = download_indicator_data(aid_data$id, aid_data$`Aid_Effectiveness`, countries)

#=========================
# 8. Pivot to wide format (each indicator = column)
#=========================
to_wide_format = function(df) {
  df %>%
    mutate(date = as.numeric(date)) %>%
    pivot_wider(
      id_cols = c(iso2c, iso3c, country, date),
      names_from = indicator_id,
      values_from = value
    ) %>%
    arrange(country, date)
}

health_wide = to_wide_format(health_raw)
poverty_wide = to_wide_format(poverty_raw)
education_wide = to_wide_format(education_raw)
gender_wide = to_wide_format(gender_raw)
agriculture_wide = to_wide_format(agriculture_raw)
aid_wide = to_wide_format(aid_raw)

#=========================
# 9. Create metadata files linking indicator_id to indicator_name
#=========================
health_meta = health_data %>% select(indicator_id = id, indicator_name = Health)
poverty_meta = poverty_data %>% select(indicator_id = id, indicator_name = Poverty)
education_meta = education_data %>% select(indicator_id = id, indicator_name = Education)
gender_meta = gender_data %>% select(indicator_id = id, indicator_name = Gender)
agriculture_meta = agriculture_data %>% select(indicator_id = id, indicator_name =
```

```
`Agriculture & Rural Development`)
aid_meta = aid_data %>% select(indicator_id = id, indicator_name = `Aid Effectiveness`)

#=========================
# 10. Create outputs folder if it doesn't exist
#=========================
dir.create("outputs", showWarnings = FALSE)

#=========================
# 11. Save raw and wide CSVs
#=========================
write_csv(health_wide,  "outputs/health_wide.csv")
write_csv(poverty_wide, "outputs/poverty_wide.csv")
write_csv(education_wide, "outputs/education_wide.csv")
write_csv(gender_wide, "outputs/gender_wide.csv")
write_csv(agriculture_wide, "outputs/agriculture_wide.csv")
write_csv(aid_wide, "outputs/aid_wide.csv")

write_csv(health_meta,   "outputs/health_metadata.csv")
write_csv(poverty_meta,  "outputs/poverty_metadata.csv")
write_csv(education_meta, "outputs/education_metadata.csv")
write_csv(gender_meta, "outputs/gender_metadata.csv")
write_csv(agriculture_meta, "outputs/agriculture_metadata.csv")
write_csv(aid_meta, "outputs/aid_metadata.csv")

#=========================
# 12. Completion message
#=========================
cat("\n All files saved in 'outputs/' folder.\n")
```