PROJECT: ANALYZE THE POPULARITY OF PROGRAMMING LANGUAGES

How can you determine which programming languages and technologies are most widely used? Which languages are gaining or losing popularity, helping you decide where to focus your efforts?

One excellent data source is Stack Overflow, a programming question-and-answer site with more than 16 million questions on programming topics. Each Stack Overflow question is tagged with a label identifying its topic or technology. By counting the number of questions related to each technology, you can estimate the popularity of different programming languages.

In this project, you will use data from the Stack Exchange Data Explorer to examine the relative popularity of R compared to other programming languages.

You'll work with a dataset containing one observation per tag per year, including the number of questions for that tag and the total number of questions that year.

`stack_overflow_data.csv`

Column	Description
`year`	The year the question was asked (2008-2020)
`tag`	A word or phrase that describes the topic of the question, such as the programming language
`num_questions`	The number of questions with a certain tag in that year
`year_total`	The total number of questions asked in that year

```
# Load necessary packages
library(readr)
library(dplyr)
library(ggplot2)
```

Hidden output

```
# Load the dataset
data <- read_csv("stack_overflow_data.csv")
```

Hidden output

```
# View the dataset
head(data)
```

index	year	tag	num_questions	year_total
1	2008	treeview	69	168
2	2008	scheduled-tasks	30	168
3	2008	specifications	21	168
4	2008	rendering	35	168
5	2008	http-post	6	168
6	2008	static-assert	1	168

Rows: 6 Expand

```
# Start coding here
# Use as many cells as you like!
# Load the required package
library(dplyr)

# Load your data
data <- read.csv("stack_overflow_data.csv", header = TRUE)

# Check column names to confirm structure
names(data)
```

```
'year' · 'tag' · 'num_questions' · 'year_total'
```

```r
# Step 1: Filter the dataset for year 2020
data_2020 <- filter(data, year == 2020)

# Step 2: Total number of questions in 2020
year_total <- sum(data_2020$num_questions, na.rm = TRUE)

# Step 3: Filter for R-specific tag (assumes tag is lower-case "r")
r_questions <- filter(data_2020, tag == "r")

# Step 4: Calculate percentage
percentage <- (r_questions$num_questions / year_total) * 100

# Step 5: Create final output data frame
r_2020 <- data.frame(
  year = 2020,
  tag = "r",
  num_questions = r_questions$num_questions,
  year_total = year_total,
  percentage = percentage
)

# View it
print(r_2020)
```

```
  year tag num_questions year_total percentage
1 2020   r         52662    5452545  0.9658242
```

```r
# Step 1: Filter for relevant years
data_filtered <- filter(data, year >= 2015 & year <= 2020)

# Step 2: Group by tag and sum
tag_totals <- data_filtered %>%
  group_by(tag) %>%
  summarise(total_questions = sum(num_questions, na.rm = TRUE)) %>%
  arrange(desc(total_questions))

# Step 3: Get top 5 tags
highest_tags <- head(tag_totals$tag, 5)         # character vector
highest_tags_df <- head(tag_totals, 5)          # data frame (if needed)

# Print results
print(highest_tags)
print(highest_tags_df)
```

```
[1] "javascript" "python"     "java"       "android"    "c#"
# A tibble: 5 × 2
  tag        total_questions
  <chr>                <int>
1 javascript         1373634
2 python             1187838
3 java                982747
4 android             737330
5 c#                  730045
```

```
R...
```

```r
# Load required libraries
library(dplyr)
library(ggplot2)

# Compute yearly totals and percentage
year_totals <- data %>%
  group_by(year) %>%
  summarise(year_total = sum(num_questions, na.rm = TRUE))

data_percentage <- data %>%
  inner_join(year_totals, by = "year") %>%
  mutate(percentage = (num_questions / year_total) * 100)

# Filter for top 5 tags and years 2015-2020
data_subset <- data_percentage %>%
  filter(tag %in% highest_tags, year >= 2015)

# Create the line plot
ggplot(data_subset, aes(x = year, y = percentage, color = tag)) +
  geom_line(size = 1.2) +
  geom_point(size = 2) +
  labs(
    title = "Top 5 Programming Language Tags on Stack Overflow (2015-2020)",
    x = "Year",
    y = "Percentage of Total Questions",
    color = "Tag"
  ) +
  theme_minimal(base_size = 14)
```