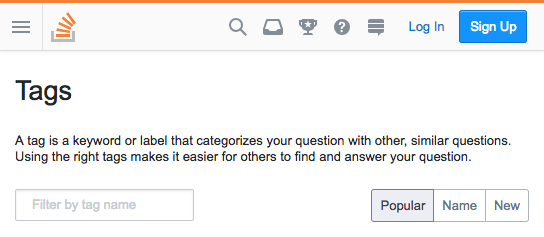
## **1. Data on tags over time**

How can we tell what programming languages and technologies are used by the most people? How about what languages are growing and which are shrinking, so that we can tell which are most worth investing time in?

One excellent source of data is [Stack Overflow](https://stackoverflow.com/), a programming question and answer site with more than 16 million questions on programming topics. By measuring the number of questions about each technology, we can get an approximate sense of how many people are using it. We're going to use open data from the [Stack Exchange Data Explorer](https://data.stackexchange.com/) to examine the relative popularity of languages like R, Python, Java and Javascript have changed over time.

Each Stack Overflow question has a **tag**, which marks a question to describe its topic or technology. For instance, there's a tag for languages like [R](https://stackoverflow.com/tags/r) or [Python](https://stackoverflow.com/tags/python), and for packages like [ggplot2](https://stackoverflow.com/questions/tagged/ggplot2) or [pandas](https://stackoverflow.com/questions/tagged/pandas).



We'll be working with a dataset with one observation for each tag in each year. The dataset includes both the number of questions asked in that tag in that year, and the total number of questions asked in that year.

The dataset on questions per year was downloaded from the [**Stack Exchange Data Explorer**](https://data.stackexchange.com/). The file is also downloadable [**here**](https://gist.github.com/dgrtwo/a30d99baa9b7bfc9f2440b355ddd1f75).

# **Task 1: Instructions**

Load the dataset and the packages required to analyze the dataset.

* Load the readr and dplyr packages.
* Load the dataset datasets/by\_tag\_year.csv into a variable named by\_tag\_year using the read\_csv() function (**not** read.csv()).
* Print by\_tag\_year.

R CODE TEMPLATE:

*# Load libraries*

*# .... YOUR CODE FOR TASK 1 ....*

*# .... YOUR CODE FOR TASK 1 ....*

​

​

*# Load dataset*

by\_tag\_year **<-** **....**

​

*# Inspect the dataset*

*# .... YOUR CODE FOR TASK 1 ....*

## 

## 

## 

## 

## 

## 

## **2. Now in fraction format**

This data has one observation for each pair of a tag and a year, showing the number of questions asked in that tag in that year and the total number of questions asked in that year. For instance, there were 54 questions asked about the .htaccess tag in 2008, out of a total of 58390 questions in that year.

Rather than just the counts, we're probably interested in a percentage: the fraction of questions that year that have that tag. So let's add that to the table.

# **Task 2: Instructions**

Create a new column that for each tag-year combination contains the fraction of questions in that year that have that tag.

* Use mutate() to add a column called fraction to by\_tag\_year, representing number divided by year\_total. Name the new table by\_tag\_year\_fraction.
* Print by\_tag\_year\_fraction.

R CODE TEMPLATE:

*# Add fraction column*

by\_tag\_year\_fraction **<-** **....**

​

*# Print the new table*

*# .... YOUR CODE FOR TASK 2 ....*

## 

## 

## **3. Has R been growing or shrinking?**

So far we've been learning and using the R programming language. Wouldn't we like to be sure it's a good investment for the future? Has it been keeping pace with other languages, or have people been switching out of it?

Let's look at whether the fraction of Stack Overflow questions that are about R has been increasing or decreasing over time.

# **Task 3: Instructions**

Filter for R tags.

* Use filter() to get only the observations from by\_tag\_year\_fraction that represent R, saving them as r\_over\_time.
* Print r\_over\_time.

R CODE TEMPLATE:

*# Filter for R tags*

r\_over\_time **<-** **....**

​

*# Print the new table*

*# .... YOUR CODE FOR TASK 3 ....*

## **4. Visualizing change over time**

Rather than looking at the results in a table, we often want to create a visualization. Change over time is usually visualized with a line plot.

# **Task 4: Instructions**

Load the visualization packages and plot fraction of R tags of overall questions over time with a line plot.

* Load the ggplot2 package.
* Plot r\_over\_time with year on the x-axis and fraction on the y-axis. Add a geom\_line() layer to the plot to create a line plot.

R CODE TEMPLATE:

*# Load ggplot2*

*# .... YOUR CODE FOR TASK 4 ....*

​

*# Create a line plot of fraction over time*

*# .... YOUR CODE FOR TASK 4 ....*

## 

## 

## 

## 

## 

## 

## 

## **5. How about dplyr and ggplot2?**

Based on that graph, it looks like R has been growing pretty fast in the last decade. Good thing we're practicing it now!

Besides R, two other interesting tags are dplyr and ggplot2, which we've already used in this analysis. They both also have Stack Overflow tags!

Instead of just looking at R, let's look at all three tags and their change over time. Are each of those tags increasing as a fraction of overall questions? Are any of them decreasing?

# **Task 5: Instructions**

Filter for the observations where tag is R, dplyr, or ggplot2, plot their fraction of overall questions over time with a line plot.

* Combine the tags "r", "dplyr" and "ggplot2" into a vector named selected\_tags using c().
* Use filter() on by\_tag\_year\_fraction, along with the %in% operator, to get only the subset of tags in selected\_tags. Name the new table selected\_tags\_over\_time.
* Visualize the popularity of these three tags with a line plot in ggplot2 (with year on the x-axis and fraction on the y-axis) using color to represent tag.

R CODE TEMPLATE:

*# A vector of selected tags*

selected\_tags **<-** **....**

​

*# Filter for those tags*

selected\_tags\_over\_time **<-** **....**

​

*# Plot tags over time on a line plot using color to represent tag*

*# .... YOUR CODE FOR TASK 5 ....*

## **6. What are the most asked-about tags?**

It's sure been fun to visualize and compare tags over time. The dplyr and ggplot2 tags may not have as many questions as R, but we can tell they're both growing quickly as well.

We might like to know which tags have the most questions *overall*, not just within a particular year. Right now, we have several rows for every tag, but we'll be combining them into one. That means we want group\_by() and summarize().

Let's look at tags that have the most questions in history.

# **Task 6: Instructions**

Find and sort the total number of questions for each tag.

* Use the group\_by() and summarize() verbs on by\_tag\_year to find the **total** number of questions for each tag, saving the column as tag\_total. Then use the arrange() verb to sort the table in descending order of the tag\_total column. Save the result to sorted\_tags.
* Print sorted\_tags.

R CODE TEMPLATE:

*# Find total number of questions for each tag*

sorted\_tags **<-** by\_tag\_year **%>%**

*# .... YOUR CODE FOR TASK 6 ....*

​

*# Print the new table*

*# .... YOUR CODE FOR TASK 6*

## **7. How have large programming languages changed over time?**

We've looked at selected tags like R, ggplot2, and dplyr, and seen that they're each growing. What tags might be *shrinking*? A good place to start is to plot the tags that we just saw that were the most-asked about of all time, including JavaScript, Java and C#.

# **Task 7: Instructions**

Filter for the largest tags and plot them on a line plot.

* Use the filter() verb to filter by\_tag\_year\_fraction only for the tags in highest\_tags, which are the six largest tags.
* Create a line plot of the fraction of questions each of these tags made up over time, using color to represent the tag.

You can extract just one column from a table using $. As seen in the sample code, sorted\_tags$tag extracts just the tag column.

You can get just the first six items from a vector using head().

R CODE TEMPLATE:

*# Get the six largest tags*

highest\_tags **<-** head(sorted\_tags**$**tag)

​

*# Filter for the six largest tags*

by\_tag\_subset **<-** **....**

​

*# Plot tags over time on a line plot using color to represent tag*

*# .... YOUR CODE FOR TASK 7 ....*

## **8. Some more tags!**

Wow, based on that graph we've seen a lot of changes in what programming languages are most asked about. C# gets fewer questions than it used to, and Python has grown quite impressively.

This Stack Overflow data is incredibly versatile. We can analyze *any* programming language, web framework, or tool where we'd like to see their change over time. Combined with the reproducibility of R and its libraries, we have ourselves a powerful method of uncovering insights about technology.

To demonstrate its versatility, let's check out how three big mobile operating systems (Android, iOS, and Windows Phone) have compared in popularity over time. But remember: this code can be modified simply by changing the tag names!

# **Task 8: Instructions**

Filter for specific tags then plot their fraction of overall questions over time with a line plot.

* Combine the tags "android", "ios" and "windows-phone" into a vector named my\_tags using c().
* Use filter() on by\_tag\_year\_fraction to get only the subset of tags in my\_tags. Name the new table by\_tag\_subset.
* Visualize the popularity of these tags with a line plot in ggplot2 (with year on the x-axis and fraction on the y-axis) using color to represent tag.

R CODE TEMPLATE:

*# Get tags of interest*

my\_tags **<-** **....**

​

*# Filter for those tags*

by\_tag\_subset **<-** **....**

​

*# Plot tags over time on a line plot using color to represent tag*

*# .... YOUR CODE FOR TASK 8 ....*

Congratulations on reaching the end of the Project!