# Overview

The goal of this is to write a simple command-line program that will print out a list of food trucks, given a source of food truck data from the San Francisco government's API.

# The Task

## Data

The San Francisco government's website has a public data source of food trucks (https://data.sfgov.org/Economy-and-Community/Mobile-Food-Schedule/jjew-r69b). The data can be accessed in a number of forms, including JSON, CSV, and XML. How you access the data is up to you, but you can find some useful information about making an API request to this data source here (https://dev.socrata.com/foundry/data.sfgov.org/bbb8-hzi6).

## The Problem

Write a command line program that prints out a list of food trucks that are open at the current date, when the program is being run. So if I run the program at noon on a Friday, I should see a list of all the food trucks that are open then.

### Additional notes

Please display the name and address of the trucks and sort the output alphabetically by name.

Please display results in pages of 10 trucks. That is: if there are more than 10 food trucks open, the program should display the first 10, then wait for input from the user before displaying the next 10 (or fewer if there are fewer than 10 remaining), and so on until there are no more food trucks to display.

Please do not include tests in your submission, but please do write code that would be easy to test.

## Criteria

We will primarily evaluate programs on code quality and output correctness.

For quality, we expect code to be easy to read and maintain, performant, and reliable. You should submit code that you are proud to have written.

For correctness, we expect that whenever the program is run, it displays all and only trucks open at that time, in pages of 10, as described above.

## Example

```
$ show-open-food-trucks
 NAME                       ADDRESS
 Mang Hang Catering         1 Thomas More Way
 Steve's Mobile Deli        145 King Street
```

You should use the programming language you feel most comfortable in. Use of any external libraries or packages is permitted and encouraged! This command line utility should run on linux or Mac OSX (if you need an exception to this, let us know).

We've provided basic stubs in a few languages, feel free to use them to get started:
- Java
- JavaScript/Node
- Python
- Ruby

# Submitting your work

Please email us a zipped folder containing your work, or you may put your solution on GitHub and send us the link. Your submission should include:
1. Your code.
2. A README file that contains instructions on:
   - how to install dependencies.
   - how to build your program.
   - how to run your program, with example commands on how to run it if necessary.
3. A one- or two-paragraph write-up describing, at a high level, what you would do differently if you were asked to build this as a full-scale web application. In your write-up, please focus on the technical differences between the command-line program and the web application, rather than on the product differences.

## Submission Notes

We recommend you spend roughly two hours on this project.

We don't expect your write-up to take more than thirty minutes to complete. Therefore, keep your response high-level and not more than two paragraphs. Do spend time polishing and packaging up your submission so it is easy to install, run, and review.

Our goal is to evaluate your submission as anonymously as possible, so please try to remove any identifying information from your code (your IDE may automatically add your name, for example).