

Technical Assessment Project

Included	Purpose
Technical Assessment Project Document	To provide the individual with background information and challenges
CSVs with student, staff, section, roster, and parent data	Data files used to answer questions for the proposed challenges

Introduction

The data files (provided separately) are representative of a set of CSV files created by extracting data from a customer's Student Information System (SIS) and sent to ParentSquare. The challenges below demonstrate the candidate's ability to write scripts to process these data files and generate meaningful output.

Challenges

Using the scripting language of your choice (e.g., Ruby, Python, Javascript, or other), write a command-line script to solve at least 4 of the challenges below. For each challenge you select, please provide the code for your script and an example of the script's output.

1. Print the 3 most commonly used languages and the count of parents for each language.
2. Print a list of students (student_id, first_name, last_name) who do not have a cell phone number.
3. Print a list of students (student_id, first_name, last_name) who are enrolled in a section with a course_name of Physics 9.
4. Print a list of students (student_id, first_name, last_name) who do not have parents listed in the parents.csv or have parents with no contact information.
5. Print a list of sections (section_id, course_name) who do not have any students enrolled.
6. Print a list of sections and the students in each section (section_id, course_name, student_ids).
7. Print a list of staff members (staff_id, first_name, last_name) who are connected to a section.

Bonus Challenges

Again, using the scripting language of your choice, write a command-line script to solve any of the challenges below. For each challenge you select, please provide the code for your script and an example of the script's output.

1. Print a language mapping for all the language codes in the parents.csv file that would correspond to the ISO-629-1 codes found [here](#).

Below is an example what the script's output should look like:

```
English: en,  
Spanish: es,  
...
```

2. Check if any phone numbers appear in two or more of these files: students.csv, staff.csv, and parents.csv.

Submission

With your submission, please include a readme.txt file that includes:

- The programming language you used and its version number
- How to properly run the code provided for each challenge (the command to type to run it, any additional installation of libraries, etc.)

Evaluation

We will evaluate your submission as follows:

- Did you provide a readme.txt file containing the information requested?
- Are the instructions you provided clear and accurate?
- Does your script run without errors?
- Does the script solve the challenge correctly by producing the correct output?
- Does the script solve the challenge in an efficient manner, in as few steps as possible?