



Integrations Engineer Technical Task



Languages - Choose One:

1. Ruby (preferred)
2. Python
3. Powershell
4. Other Scripting Language of your choice

Task - Using the provided CSV file:

Please attempt Questions 1-9. You will present your program code and execute the program to generate the output as part of the 2nd round interview & presentation. Your task should be submitted for review no later than 48 hours before your interview time/date.

Instructions

Create a program using your choice of scripting languages, implementing best practices techniques that execute steps 1-9. If for any reason you cannot finish the task or fully complete any of the steps then please ensure you bring and show your attempt. Your task should submitted no later than 24 hours before your interview

1. Manually create a root program directory (folder) named "PoppuloTask", **this directory will store your program/code & the required folders/files defined below.**
2. Within the root program directory, ***programmatically*** create a subdirectory named "***year-month-day***", where "year-month-day" is a value specific to the day the program is run, **which will contain all of the below defined files.**
3. Generate a text file, named "headers.txt", containing a string of the provided headers only.
4. Generate a separate CSV file per 'Department' value found within the provided CSV file, name the CSV file '*departmentValue.csv*', where *departmentValue* is the actual Department value itself.
5. Generate a separate CSV file named 'no_countries.csv', where the 'Country' field is omitted from the file entirely.
6. Generate a 'Master' XML file named 'master.xml', containing all of the data found within the provided CSV file, using the XML template seen below.
7. Generate a separate 'Child' XML file per 'Department' value, using the created CSV files (see point 5 above), named '*departmentValue.xml*', using the XML template seen below.
8. When generating the XML files, within the XML node named "data", embed the 'Master' CSV file data or 'Child' CSV file data, as created (see points 7 & 8).
9. Generate a CSV file named "csv_report.csv", containing the below report for example

Field Name	Percentage Filled	Percentage Not Filled	Total Values	Distinct Values	Values Not Filled
Email Address	100.000%	0.000%	305,584	305,584	0
First Name	99.805%	0.195%	304,987	64,868	597

Provide values relative per field, below each header, as seen in the above example. The source data file should be used to generate these stats

XML Example:

```
<subscriber_import_job>
  <accept_terms>true</accept_terms>
  <reactivate_api_removed>false</reactivate_api_removed>
  <reactivate_admin_removed>true</reactivate_admin_removed>
  <reactivate_bounced_removed>false</reactivate_bounced_removed>
  <tags>
    <tag name="Some Tag" />
  </tags>
  <subscriber_data>
    <columns>email,Surname,FirstName,Department,Country</columns>
    <skip_first_line>true</skip_first_line>
    <field_separator>comma</field_separator>
    <data>
      email,Surname,FirstName,Department,Country
      xyz@newsweaver.ie,Roycroft,George,Finance,Ireland
      hsdafgj@newsweaver.co.uk,Wolfenstein,Peter,Marketing,France
    </data>
  </subscriber_data>
</subscriber_import_job>
```

Version history

Change history for this document:

Document Revisions			
Version	Author	Comment	Date Modified
1.0	Robert O'Brien	Initial version	08 November 2017
1.1	Brendan O'Connor	V2	10 July 2018
1.2	Carl Ledwidge	V3 Update & rev	22 Nov 2019
1.3	Carl Ledwidge	V4 Update & rev	22 June 2021