

Catch Coding Challenge 2

Please complete the below challenge in PHP 8.0+ using the Symfony framework.

Task

Read a data file, process each record, and produce an output file.

The input file is in jsonlines format (<http://jsonlines.org>), with each record representing an ecommerce order. Each order contains data about the customer, shipping details, payment data, items purchased, and any applied discounts. The file is available in AWS S3 at <https://s3-ap-southeast-2.amazonaws.com/catch-code-challenge/challenge-1/orders.jsonl>

You need to write a program that produces a new output csv file out.csv with the data defined in the table below, with a single record per order. The output file should be a well-formed CSV (according to <https://csvlint.io>) file with the following fields, which represents a summary of the data from the input file.

order_id	The numeric order id
order_datetime	The datetime the order was placed in ISO 8601 format in the UTC timezone.
total_order_value	The dollar sum of all line items in the order, <i>excluding</i> shipping, with all discounts applied. Note, discounts do not apply to shipping.
average_unit_price	The average price of each unit in the order, in dollars.
distinct_unit_count	The count of unique units contained in the order.
total_units_count	The total number of units in the order
customer_state	The state code from the customer's shipping address, e.g. "Victoria"

Requirements

- The input file should be automatically downloaded from the internet each time the program is run
- Order records with 0 total order value should be excluded from the summary output.
- The program should be run via a single command line script.
- Include a README file that includes at the very least:
 - your name and contact details
 - instructions to bootstrap and run your program
 - anything else you want us to know about the program and your solution design
- Manage your project in a git repository, committing parts as you build out your program.

Recommendations

- Use third party libraries and packages where appropriate. If you specifically want to write a component you know is available off the shelf, write your reasons for doing so in the readme.
- Use a code formatter to ensure code-style consistency, a code linter or static analysis tool.

Bonus points

The below are suggestions for ways to show-off some additional skills. There is no requirement to do any of the bonus points. Also feel free to use your imagination and surprise us!

- Specify as an argument to the program an email to which the output should be sent.
- Use a third-party API to enrich the output data, e.g. geocode the customers address into latitude and longitude. Be creative!
- Validate your CSV output file programmatically via a third-party API either as part of the standard generation, or as a separate console command or script.
- Allow formatting of the output file to be in one or more other formats, e.g. jsonl, XML, Yaml, etc.

Submission

Please push your project to a private GitHub repository and share it with whoever interviewed you, their github profile will be provided in the email this is attached to.