

Evidences

Task completed using Perl 5, some requirements were not clear so I decided to add more functionalities to make sure that I achieve more than the minimum requirements requested.

1. Input and output file paths and names should be passed as script parameters rather than hard coded in the code.

Example: `<script_name>.pl -i <Original CSV file> -o <Output CSV file>`

To achieve this requirement I am using `Getopt::Long` to handle script parameters (`-i` for input file and `-o` for output file), which allows flexible input/output file paths.

- Running command using path to input file different from where the script is and writing the output file in another folder and receiving successful result.

```
lpspalma@DESKTOP-TSVB697:/mnt/c/Users/luksip/OneDrive/Documentos/Lucas/Aulas.Tutoriais/perlReadWrite$ perl read_write.pl -i "/mnt/c/Users/luksip/OneDrive/Documentos/Lucas/Aulas.Tutoriais/SampleCSVFile_11kb.csv" -o "/mnt/c/Users/luksip/OneDrive/Documentos/Lucas/Aulas.Tutoriais/perlReadWrite/output.csv"
Column column3:
Sum: 503690
Positive: 100 - 100.00%
Negative: 0 - 0.00%
Column column4:
Sum: 10452.15
Positive: 49 - 49.00%
Negative: 51 - 51.00%
Column column5:
Sum: 7742.849999999999
Positive: 100 - 100.00%
Negative: 0 - 0.00%
Column column6:
Sum: 1350.1
Positive: 100 - 100.00%
Negative: 0 - 0.00%
CSV file has been processed successfully and written to '/mnt/c/Users/luksip/OneDrive/Documentos/Lucas/Aulas.Tutoriais/perlReadWrite/output.csv'.
lpspalma@DESKTOP-TSVB697:/mnt/c/Users/luksip/OneDrive/Documentos/Lucas/Aulas.Tutoriais/perlReadWrite$
```

- Running command using path to input file in the same folder of the script and writing the output file in the same folder, receiving successful result.

```
lpspalma@DESKTOP-TSVB697:/mnt/c/Users/luksip/OneDrive/Documentos/Lucas/Aulas.Tutoriais/perlReadWrite$ perl read_write.pl -i SampleCSVFile_11kb.csv -o output.csv
Column column3:
Sum: 503690
Positive: 100 - 100.00%
Negative: 0 - 0.00%
Column column4:
Sum: 10452.15
Positive: 49 - 49.00%
Negative: 51 - 51.00%
Column column5:
Sum: 7742.849999999999
Positive: 100 - 100.00%
Negative: 0 - 0.00%
Column column6:
Sum: 1350.1
Positive: 100 - 100.00%
Negative: 0 - 0.00%
CSV file has been processed successfully and written to 'output.csv'.
lpspalma@DESKTOP-TSVB697:/mnt/c/Users/luksip/OneDrive/Documentos/Lucas/Aulas.Tutoriais/perlReadWrite$ |
```

- Running command using a path that doesn't have a valid file and showing how to handle the error.

```
lpspalma@DESKTOP-TSVB697:/mnt/c/Users/luksip/OneDrive/Documentos/Lucas/Aulas.Tutoriais/perlReadWrite$ perl read_write.pl -i anypath.csv -o output.csv
Error: Could not open 'anypath.csv': No such file or directory
lpspalma@DESKTOP-TSVB697:/mnt/c/Users/luksip/OneDrive/Documentos/Lucas/Aulas.Tutoriais/perlReadWrite$
```

- The output CSV file must contain the column headers which are missing from the original file, as well as the original data.

To achieve this requirement, I ensure the output file contains headers (column0, column1, column2) which are added if missing from the original file.

Since the headers name was not provided, I decided to use generic column names

```
my @header = map { "column$_ " } (0 .. $max_columns - 1);
```

Input file with no Header

```
lpspalma@DESKTOP-TSVB697:/mnt/c/Users/luksip/OneDrive/Documents/Lucas/Aulas.Tutoriais/perlReadWrite$ ls
CSVProcessor.pm      DataProcessorTest.pl      SampleCSVFile_11kb.csv      read_write.pl      '-$tview_task_1_Perl.docx'
CSVProcessorTest.pl  Interview_task_1_Perl.docx  different_path.png          same_path.png
DataProcessor.pm     README.md                  output.csv                  wrong_path.png

lpspalma@DESKTOP-TSVB697:/mnt/c/Users/luksip/OneDrive/Documents/Lucas/Aulas.Tutoriais/perlReadWrite$ cat SampleCSVFile_11kb.csv
1,"Eldon Base for stackable storage shelf, platinum",Muhammed MacIntyre,3,-213.25,38.94,35,Nunavut,Storage & Organization,0.8
2,"1.7 Cubic Foot Compact ""Cube"" Office Refrigerators",Barry French,293,457.81,208.16,68.02,Nunavut,Appliances,0.58
3,"Cardinal Slant-De Ring Binder, Heavy Gauge Vinyl",Barry French,293,46.71,8.69,2.99,Nunavut,Binders and Binder Accessories,0.39
4,R380,Clay Rozendal,483,1198.97,195.99,3.99,Nunavut,Telephones and Communication,0.58
5,Holmes HEPA Air Purifier,Carlos Soltero,515,30.94,21.78,5.94,Nunavut,Appliances,0.5
6,G.E. Longer-Life Indoor Recessed Floodlight Bulbs,Carlos Soltero,515,4.43,6.64,4.95,Nunavut,Office Furnishings,0.37
7,"Angle-D Binders with Locking Rings, Label Holders",Carl Jackson,613,-54.04,7.3,7.72,Nunavut,Binders and Binder Accessories,0.38
8,"SAFCO Mobile Desk Side File, Wire Frame",Carl Jackson,613,127.70,42.76,6.22,Nunavut,Storage & Organization,
9,"SAFCO Commercial Wire Shelving, Black",Monica Federle,643,-695.26,138.14,35,Nunavut,Storage & Organization,
10,Xerox 198,Dorothy Badders,678,-226.36,4.98,8.33,Nunavut,Paper,0.38
11,Xerox 1980,Neola Schneider,807,-166.85,4.28,6.18,Nunavut,Paper,0.4
12,Advantus Map Pennant Flags and Round Head Tacks,Neola Schneider,807,-14.33,3.95,2,Nunavut,Rubber Bands,0.53
13,Holmes HEPA Air Purifier,Carlos Daly,868,134.72,21.78,5.94,Nunavut,Appliances,0.5
14,"DS/HD IBM Formatted Diskettes, 200/Pack - Staples",Carlos Daly,868,114.46,47.98,3.61,Nunavut,Computer Peripherals,0.71
15,"Wilson Jones 1"" Hanging DubLock® Ring Binders",Claudia Miner,933,-4.72,5.28,2.99,Nunavut,Binders and Binder Accessories,0.37
16,Ultra Commercial Grade Dual Valve Door Closer,Neola Schneider,995,782.91,39.89,3.04,Nunavut,Office Furnishings,0.53
17,"#10-4 1/8"" x 9 1/2"" Premium Diagonal Seam Envelopes",Allen Rosenblatt,998,93.80,15.74,1.39,Nunavut,Envelopes,0.4
18,Hon 4-Shelf Metal Bookcases,Sylvia Foulston,1154,440.72,100.98,26.22,Nunavut,Bookcases,0.6
19,"Lesro Sheffield Collection Coffee Table, End Table, Center Table, Corner Table",Sylvia Foulston,1154,-481.04,71.37,69,Nunavut,Tables,0.68
20,g520,Jim Radford,1344,-11.68,65.99,5.26,Nunavut,Telephones and Communication,0.59
```

Output file with new header added

```
lpspalma@DESKTOP-TSVB697:/mnt/c/Users/luksip/OneDrive/Documents/Lucas/Aulas.Tutoriais/perlReadWrite$ ls
CSVProcessor.pm      DataProcessorTest.pl      SampleCSVFile_11kb.csv      output.csv      wrong_path.png
CSVProcessorTest.pl  Interview_task_1_Perl.docx  different_path.png          read_write.pl   '-$tview_task_1_Perl.docx'
DataProcessor.pm     README.md                  'no header input.png'      same_path.png

lpspalma@DESKTOP-TSVB697:/mnt/c/Users/luksip/OneDrive/Documents/Lucas/Aulas.Tutoriais/perlReadWrite$ cat output.csv
column0,column1,column2,column3,column4,column5,column6,column7,column8,column9
1,"Eldon Base for stackable storage shelf, platinum","Muhammed MacIntyre",3,-213.25,38.94,35,Nunavut,"Storage & Organization",0.8
2,"1.7 Cubic Foot Compact ""Cube"" Office Refrigerators","Barry French",293,457.81,208.16,68.02,Nunavut,Appliances,0.58
3,"Cardinal Slant-De Ring Binder, Heavy Gauge Vinyl","Barry French",293,46.71,8.69,2.99,Nunavut,"Binders and Binder Accessories",0.39
4,R380,Clay Rozendal,483,1198.97,195.99,3.99,Nunavut,Telephones and Communication,0.58
5,"Holmes HEPA Air Purifier","Carlos Soltero",515,30.94,21.78,5.94,Nunavut,Appliances,0.5
6,"G.E. Longer-Life Indoor Recessed Floodlight Bulbs","Carlos Soltero",515,4.43,6.64,4.95,Nunavut,"Office Furnishings",0.37
7,"Angle-D Binders with Locking Rings, Label Holders","Carl Jackson",613,-54.04,7.3,7.72,Nunavut,"Binders and Binder Accessories",0.38
8,"SAFCO Mobile Desk Side File, Wire Frame","Carl Jackson",613,127.70,42.76,6.22,Nunavut,"Storage & Organization",
9,"SAFCO Commercial Wire Shelving, Black","Monica Federle",643,-695.26,138.14,35,Nunavut,"Storage & Organization",
10,"Xerox 198","Dorothy Badders",678,-226.36,4.98,8.33,Nunavut,Paper,0.38
11,"Xerox 1980","Neola Schneider",807,-166.85,4.28,6.18,Nunavut,Paper,0.4
12,"Advantus Map Pennant Flags and Round Head Tacks","Neola Schneider",807,-14.33,3.95,2,Nunavut,"Rubber Bands",0.53
13,"Holmes HEPA Air Purifier","Carlos Daly",868,134.72,21.78,5.94,Nunavut,Appliances,0.5
14,"DS/HD IBM Formatted Diskettes, 200/Pack - Staples","Carlos Daly",868,114.46,47.98,3.61,Nunavut,"Computer Peripherals",0.71
15,"Wilson Jones 1"" Hanging DubLock® Ring Binders","Claudia Miner",933,-4.72,5.28,2.99,Nunavut,"Binders and Binder Accessories",0.37
```

3. The data in the original CSV must be parsed and stored inside a hash.
The `read_csv_data` subroutine reads the CSV data into a hash (`%csv_hash`), storing the parsed data.

```
sub read_csv_data {
    my ($input_file) = @_;
    open my $input, '<', $input_file or die "Error: Could not open '$input_file': $!\n";
    my $csv = Text::CSV->new({ binary => 1, auto_diag => 1 });

    my @rows;
    my $max_columns = 0;
    while (my $row = $csv->getline($input)) {
        push @rows, $row;
        my $num_columns = scalar @$row;
        $max_columns = $num_columns if $num_columns > $max_columns;
    }
    close $input;

    my @header = map { "column$_" } (0 .. $max_columns - 1);

    my %csv_hash;
    my $row_index = 0;

    foreach my $row (@rows) {
        my %row_hash;
        @row_hash{@header} = (@$row, ('') x ($max_columns - @$row));
        $csv_hash{$row_index} = \%row_hash;
        $row_index++;
    }

    return (\%csv_hash, \@header);
}
```

4. The sum of the column 3 values must be printed out.
Although the script calculates sums, counts, and percentages for specified columns, it doesn't specifically print the sum of column 3 values. But the requirement was achieved since the script print the sum of the column 3 together with the other columns.

```
lpsalma@DESKTOP-TSVB697:/mnt/c/Users/luksp/OneDrive/Documentos/Lucas/Aulas.Tutoriais/perlReadWrite$ perl read_write.pl -i SampleCSVFile_11kb.csv -o output.csv
Column column3:
Sum: 503690
Positive: 100 - 100.00%
Negative: 0 - 0.00%
Column column4:
Sum: 10452.15
Positive: 49 - 49.00%
Negative: 51 - 51.00%
Column column5:
Sum: 7742.849999999999
Positive: 100 - 100.00%
Negative: 0 - 0.00%
Column column6:
Sum: 1350.1
Positive: 100 - 100.00%
Negative: 0 - 0.00%
CSV file has been processed successfully and written to 'output.csv'.
lpsalma@DESKTOP-TSVB697:/mnt/c/Users/luksp/OneDrive/Documentos/Lucas/Aulas.Tutoriais/perlReadWrite$ |
```

In the result we are able to find the sum of the columns 3 to 6, columns that have numerical values, when is empty, the script read the field as 0.

5. The count of negative and positive column values should also display as well as the percentage of the total number of values.

Example: Negative: 10 - 50%, Positive: 10 - 50%

The `calculate_data` subroutine calculates counts and percentages for positive and negative values in specified columns, which meets this requirement.

```
lpspalma@DESKTOP-TSVB697:/mnt/c/Users/lukspl/OneDrive/Documents/Lucas/Aulas.Tutoriais/perlReadWrite$ perl read_write.pl -i SampleCSVFile_11kb.csv -o output.csv
CSV
Column column3:
Sum: 503600
Positive: 100 - 100.00%
Negative: 0 - 0.00%
Column column4:
Sum: 10452.15
Positive: 49 - 49.00%
Negative: 51 - 51.00%
Column column5:
Sum: 7742.8499999999
Positive: 100 - 100.00%
Negative: 0 - 0.00%
Column column6:
Sum: 1350.1
Positive: 100 - 100.00%
Negative: 0 - 0.00%
CSV file has been processed successfully and written to 'output.csv'.
lpspalma@DESKTOP-TSVB697:/mnt/c/Users/lukspl/OneDrive/Documents/Lucas/Aulas.Tutoriais/perlReadWrite$
```

Error Handling and Clear Messages: I've implemented error handling for file operations (die statements with clear error messages) and provided appropriate error messages. The error messages indicate file open failures (Could not open '<filename>' for writing: <error>).

Documentation and Structure: I added comments and structured subroutines (main, read_csv_data, calculate_data, write_output, etc.), which enhances readability and maintainability. Also, the Readme doc was added in the project detailing the execution, function details and test strategy.

Test Plan: Unit tests (Test::More) that cover positive, negative, and edge scenarios for both most part of the functions. These tests validate functionality and handle error cases effectively.