SE126 – Winter 2025

Lab #4: Parallel List Processing

This is a two-part program where you will work on creating and populating parallel lists based on file data, then create and write data to a file.

**PART 1**

Write a program that utilizes the **got\_emails.csv** file. Store the file data into 1D parallel lists, then use the information in the lists to assign additional data to each employee. Use the tables below to assign each employee in the file a unique email address, a department, and a unique phone extension.

|  |  |
| --- | --- |
| **Table 1** | |
| **Field #** | **File Data** |
| 0 | First Name (unique) |
| 1 | Last Name |
| 2 | Age |
| 3 | Screen Name (unique) |
| 4 | House Allegiance |
| **For each employee, create the following 3 pieces of data and store to parallel lists:** | |
| \* | Email --> Screen Name + “@westeros.net” |
| \* | Department --> see table 2 below |
| \* | Phone Extension --> see table 2 below |

|  |  |  |
| --- | --- | --- |
| **Table 2** | | |
| **House Allegiance** | **Department** | **Phone Extension** |
| House Stark | Research & Development | 100 - 199 |
| House Targaryen | Marketing | 200 - 299 |
| House Tully | Human Resources | 300 - 399 |
| House Lannister | Accounting | 400 - 499 |
| House Baratheon | Sales | 500 - 599 |
| The Night’s Watch | Auditing | 600 - 699 |

\*Make sure each employee has a unique phone extension found within the department’s specified range. \*

When you are complete, display the following data for each employee (first name, last name, department, email, and phone extension) to the user.

Use the following field width guide to avoid unaligned field displays:

print(f"{'FIRST':8} {'LAST':10} {'EMAIL':30} {'DEPARTMENT':23} {'EXT':3}")

**PART 2**

Once you have completed populating all eight parallel lists and displaying the five required back to the user (and in the same Python file), create and write the following data for each employee to a file named **westeros.csv**: first name, last name, email, department, and phone extension. NOTE: each employee’s data should be on its own record (row) within the newly created file. You will most likely end up with an extra empty line at the end of the file (this is okay for this lab as we will not be reprocessing the data found in this new file).

Once the file is ready, close it and alert the user via a displayed message. Also tell them how many employees are in the file, and the total count of employees for each department.

**Lab #4 – Parallel List Processing & Creating/Writing to Files**

|  |  |
| --- | --- |
| **Description** | **Points** |
| **Documentation**   * Starting (Name, Date, Lab #, Prompt) - 5 * Internal (meaningful notes amongst code) - 15 | 20 |
| **Code Runs Without Errors** | 20 |
| **“Clean” and Organized Display**   * All columns have headers - 5 * All employee data is readable and all columns are in alignment -5 | 10 |
| **Lab Specific: New Material**   * All data from file is stored to lists - 5 * Lists are processed to:   + Determine and store additional employee data (email, dept, phone ext) - 10   + Display employee data to user - 3   + Write employee data to new file - 7 | 25 |
| Lab Specific: Core Concepts   * Requested data is stored to a new file named westeros.csv - 5 * Correct output values   + Displayed employee data - 5   + westeros.csv data - 5   + Total employees - 5   + Department Totals - 5 | 25 |
| Total | 100 |