**P CSP CodeX**

|  |  |  |
| --- | --- | --- |
| **LESSON: Traversing a List #3** | | **Time: 45 minutes** |
| **Project Goal:** Students will traverse a list to check if an item is in the list, and to filter the list.  **Learning Targets**   * I can use the console panel to input data. * I can traverse a list and check if a given answer is in the list. * I can create a list by appending random numbers. * I can create a list by appending input from the user. * I can filter a list to create a sub-list. | **Key Concepts**   * An empty list can be filled with data by using the append() function. * The console panel can be used for input as well as output. * The input() function returns a string. If an integer is needed, it must be converted using int(). * The display.print() uses strings. To print an integer, convert it using str(). | |
| **Assessment Opportunities**   * Traversing a List #3 Activity Guide * Class\_Schedule program * Number\_Sorter program * Animal\_Sorter program | **Success Criteria**   * Use input() and the console panel to fill a list * Fill an empty list using append() * Traverse a list and check if an item is in the list * Traverse a list to filter by a condition, creating a sub-list * Traverse a list and filter by checking each item, comparing it to a second list | |
| **AP CSP Framework**  **AAP-1.D** Develop data abstraction using lists to store multiple elements.  **AAP-2.N** Write expressions that use list indexing and list procedures.  **AAP-2.O** Write iteration statements to traverse a list.  **Computational Thinking Practice 3.B** Use abstraction to manage complexity in a program  **Computational Thinking Practice 3.C** Explain how abstraction manages complexity.  **Computational Thinking Practice 4.C** Identify and correct errors in algorithms and programs, including error discovery through testing. | **Materials**   * Traversing a List #3 slides * Traversing a List #3 Activity Guide / Answers * Unit 3 Review and Test Questions * Code solutions   + Class\_Schedule   + Number\_Sorter   + Animal\_Sorter | |
| **Teacher Notes**   * This lesson will be completed on the computer, using CodeSpace for programming. * Use the Sandbox in CodeSpace for programming. This lesson is not part of a mission. * The activity guide can be distributed digitally. Space is provided for students to take notes during the programming. Since answers will vary, an answer key is not provided. * Students will create three new programs. Each program is fairly short. * Follow the slides for instructions and guidance. * Solution code for all three programs are provided. | | |