FIRIA LABS

LESSON: Create Performance Task - practice #2

Time: 45-60 minutes

Overview:

Students learned and practiced using a loop to traverse a list. They are ready to prepare for the Create Performance Task. The program must meet a set of requirements to earn all the points. This practice will meet complete Practice #1 by adding a loop to the function.

Objectives:

- I can create a program that meets all the Create PT requirements
- I can write about parts of a program:
  - Lists
  - Functions

Standards:

2-AP-12 Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.

3A-AP-14 Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.

CSP Framework:

Computational Thinking Practices:

4.C Identify and correct errors in algorithms and programs, including error discovery through testing.

6.A Collaborate in the development of solutions.

Create PT Requirements:

- Create a list
- Use the list in a meaningful way
- Create a function with at least one parameter
- The function must have sequence, selection and iteration
- Values of the parameter must affect the section of code that is executed
- Call the function with argument

Preparation:

Make a copy of the assignment or put it in the LMS.
Prepare any formative assessments you want to use in the wrap-up

Links:

- Assignment
- Instructions slide deck
- Program code
- Daily reflection form

Agenda:

- Warm-up (5 minutes)
- Remix plan (20-40 minutes)
- Remix coding (30-60 minutes)
- Peer review (10-20 minutes)
- Remix coding (20 minutes)
- Wrap-up & Assessment (10 minutes)

Vocabulary:

- No new vocabulary during this lesson
- You can review Create PT vocabulary: **parameter, argument, function, sequential, selection, iteration**

Assessment:

- Daily reflection journal or Google form
- Rubric (check-list)
- Programming Journal
- Gallery Walk

Teaching Guide

Warm-up (10-15 minutes)

🧑‍🤝‍🧑 **Discuss** – Use a discussion strategy, like journaling, working at boards, selecting random students, or a form of think-pair-share.

Options for Warm-Up

- Use the slides (**slides 2-5**) and the CollegeBoard website to introduce the create performance task and its requirements
  - CollegeBoard website about exam
  - Use the video from AP Classroom: Create Performance Task: Overview
- Discuss/review Practice #1 and the requirements it meets.

The warm-up has students copy and paste code snippets into the document. This is basically the same as the wrap-up from Practice #1. You can skip this part if you want. However, it is a good review since there were two assignments in between. Also, students may have been unclear on selecting code for snippets the first time and this will give them another opportunity for practice.

Practice #2 - programming (10-15 minutes)

💻 Randomly group students into pairs for pair programming, or have students work with the same partner they had for Practice #1 or Traversal List program.

IMPORTANT!: This program uses display.print() statements for displaying text information from lists. This isn't the most interesting way to present information, but this is intentional. Students may want to do a similar project like this for their actual Create PT. Then they can use images, JPGs, sound, lighted pixels, etc. But if they do those techniques now, then it will be an assignment that they receive feedback on and they can't do it for the Create PT. So I would encourage students to keep their program really simple here so they can expand, modify, and make it interesting for the actual PT.

💡 **Teaching tip – Step 1:**

Students open their last two programs. They resave Practice #1 as Practice #2.

💡 **Teaching tip – Step 2:**

Students look at the code from Traverse List and identify the loop for traversal.

💡 **Teaching tip – Step 3:**

Students modify the intro() and then add the for loop traversal to the program.

💡 **Teaching tip – Step 4:**

Students will add the for loop traversal to the program.

💡 **Teaching tip – Step 5:**

Students will test and debug so that the code is error-free and each button works correctly.

💡 **Teaching tip – Challenge:**

Complete if there is time – not required. Students will add their own creativity to the code. One suggestion is given, and then they can also use their own creativity to improve the program.

✅ Review the success criteria for completeness. Assignment is ready to turn in. Both students should include their names on the document.

✅ To turn in the assignment, students should download their code (FILE-DOWNLOAD), which will be a text file. Then they should submit their file through Google Classroom or your LMS.

✅ **IMPORTANT!!**

Students should clear their CodeX by running their "Clear" program.

Wrap-Up (15-25 minutes)

The wrap-up, if you want to use it, will have the students write about the code requirements, specifically the lsit and the function. Students may need help with this part, so be prepared on how you will assist the students in understanding what is being asked and how to write about their code.

Formative Assessment:

- Daily reflection journal or Google form
- Programming journal
- Wrap-up questions
- Gallery Walk
- Exit ticket

Summative Assessment: Use the success criteria to evaluate the Practice PT project

SUCCESS CRITERIA:

❏ Add a loop to a function that will traverse a list
❏ Create a program that is error-free
❏ Create a program that works correctly for each input
❏ Create a program that meets all requirements for the Create Performance Task
❏ Copy and paste code snippets that meet the programming requirements
❏ Write about a list
❏ Write about a function