


LESSON: Create Performance Task - practice #3		Time: 90 minutes
<p>Overview:</p> <p>Students have completed all the missions and several supplemental lessons. 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 take a completed program and modify it to meet all of the requirements.</p> <p>Students will use the Display2 program from the Functions, Parameters and Local Variables lesson.</p>		<p>Objectives:</p> <ul style="list-style-type: none"> • I can create a meaningful list • I can use a list in code in a meaningful way • I can create a function with a parameter • I can use the parameter in an if statement • I can create a function with iteration and selection
<p>Standards:</p> <p>2-AP-12 Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.</p> <p>3A-AP-14 Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.</p>	<p>CSP Framework:</p> <p>Computational Thinking Practices:</p> <p>4.C Identify and correct errors in algorithms and programs, including error discovery through testing.</p> <p>6.A Collaborate in the development of solutions.</p>	<p>Create PT Requirements:</p> <ul style="list-style-type: none"> • 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
<p>Preparation:</p> <p>Make a copy of the assignment or put it in the LMS.</p> <p>Prepare any formative assessments you want to use in the wrap-up</p>	<p>Links:</p> <ul style="list-style-type: none"> • Assignment • Instructions slide deck • Starter code (if needed) • Program code solution • Daily reflection form 	<p>Agenda:</p> <ul style="list-style-type: none"> • Warm-up (5 minutes) • Coding (40-60 minutes) • Wrap-up & Assessment (15 minutes)
<p>Vocabulary:</p> <ul style="list-style-type: none"> • No new vocabulary during this lesson • You can review Create PT vocabulary: parameter, argument, function, sequential, selection, iteration 		
<p>Assessment:</p> <ul style="list-style-type: none"> • Daily reflection journal or Google form • Rubric (check-list) • Wrap-up completion • Gallery Walk 		


Teaching Guide

Warm-up (5 minutes)

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

- Have students do the warm-up before using the slides
- You can change the warm-up to something else if your students need a different review

Coding (40-60 minutes)

 Students can work individually, with the same partner as they had for the last lesson, or with a new random partner.


IMPORTANT!: Students will use their Display2 program from the last lesson and modify it. They need to have it completed and accessible. Alternatively, starter code is available in the folder if students didn't finish or if their code is unusable or inaccessible.

Teaching tip – Intro:

Go through slides 2 through 4 with the class, as a review of the Create Performance Task and to set up the Practice Project for today.

Teaching tip – Coding:

Students open their code for Display2 and follow the instructions on the slide to make modifications. You will want to check in with the students regularly to see if they are having problems or don't understand a step. Hopefully they will use each other to help with any trouble spots, but be prepared to go over the instructions in case the class is struggling.

 Review the success criteria for completeness. Assignment is ready to turn in. If working in pairs, both students should include their names on the document. Students can download their program file and submit through LMS, or any way you prefer for submission.

IMPORTANT!!

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

Wrap-Up (15 minutes)

The wrap-up will have the students identify parts of the project that are requirements for the Create PT. Hopefully students won't need help with this part, but be prepared in case the students don't understand what is being asked and how to select the correct code snippets.

Formative Assessment:

- Daily reflection journal or Google form
- Wrap-up questions
- Completed program
- Exit ticket

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



SUCCESS CRITERIA:

- Modify a program to meet the requirements for the Create PT
- Create two lists
- Create functions for intro and ending
- Modify a function with a parameter to use a loop to traverse a list
- Use (access information from) the two lists in the for loop
- Use the parameter in an if statement
- Test and debug the program so that it runs as expected
- Identify the requirements in the program code by pasting images of the code in the document