


LESSON: Create Performance Task - practice #5		Time: 45-60 minutes
<p><b>Overview:</b></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 two programs from missions and merge them into one program that will meet all of the requirements.</p> <p>Students will use the <b>Heart2</b> program and <b>Billboard</b> program from the missions. Do not have students use a remix for this practice!</p>		<p><b>Objectives:</b></p> <ul style="list-style-type: none"> <li>• I can combine programs into one program</li> <li>• I can create a function with a parameter</li> <li>• I can use the parameter in an if statement</li> <li>• I can create a function with iteration and selection</li> <li>• I can use global variables in a program and in functions</li> </ul>
<p><b>Standards:</b></p> <p><b>2-AP-12</b> Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.</p> <p><b>3A-AP-14</b> Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.</p>	<p><b>CSP Framework:</b></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><b>Create PT Requirements:</b></p> <ul style="list-style-type: none"> <li>• Create a list</li> <li>• Use the list in a meaningful way</li> <li>• Create a function with at least one parameter</li> <li>• The function must have sequence, selection and iteration</li> <li>• Values of the parameter must affect the section of code that is executed (used in an if statement)</li> <li>• Call the function with argument</li> </ul>
<p><b>Preparation:</b></p> <p><b>Make a copy</b> of the assignment or put it in the LMS.</p> <p><b>Prepare</b> any formative assessments you want to use in the wrap-up</p>	<p><b>Links:</b></p> <ul style="list-style-type: none"> <li>• <a href="#">Assignment</a></li> <li>• <a href="#">Instructions slide deck</a></li> <li>• <a href="#">Mission 6 Heart2 (if needed)</a></li> <li>• <a href="#">Mission 7 Billboard (if needed)</a></li> <li>• <a href="#">Program code solution</a></li> <li>• Daily reflection form</li> </ul>	<p><b>Agenda:</b></p> <ul style="list-style-type: none"> <li>• Warm-up (5 minutes)</li> <li>• Coding (30-40 minutes)</li> <li>• Wrap-up (15 minutes)</li> </ul>
<p><b>Vocabulary:</b></p> <ul style="list-style-type: none"> <li>• No new vocabulary during this lesson</li> <li>• You can review Create PT vocabulary: <b>parameter, argument, function, sequential, selection, iteration</b></li> </ul>		
<p><b>Assessment:</b></p> <ul style="list-style-type: none"> <li>• Daily reflection journal or Google form</li> <li>• Rubric (check-list) / program completion</li> <li>• Assignment completion</li> <li>• Gallery Walk</li> </ul>		


## 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.

- Slides 2-3
- Review the requirements for the Create Performance Task
- If a review is not needed, skip over the slides and get right to the project, starting on Slide 4

### Coding (30-40 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 **Heart2** program from Mission 6 and their **Billboard** program from Mission 7. These programs were selected because **Heart2** is a relatively simple and short program, and **Billboard** uses a list. Starter code is available in the folder if students don't have access to their own code.

#### Teaching tip – Coding: Slides 4-26

Step 1 - Step 6

Students open their code for Heart2 and Billboard and follow the instructions on the slides to make a new program. 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.

#### Teaching tip – Coding (Optional): Slide 27

Step 6

This slide mentions ways to improve the program. Students DO NOT need to do this. These are just brain joggers in case they want to do a similar project for their Create PT. That is when they should do something to improve the program. The purpose of the slide is just to help them realize they can do more than the minimum for a better program, after meeting the requirements.

#### Teaching tip – Wrap-up: Slides 28-29

Step 7

These slides are for the wrap-up. See below.

#### **IMPORTANT!!**

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

### Wrap-Up (15 minutes)

If the program creation takes an entire class period, you may want to continue the assignment the next day for the wrap-up. Alternatively, students can turn in the Practice #5 and do the wrap-up portion as a different code segment practice the next day. Decide what works best for your students.

The wrap-up has two parts. The first part is to practice creating their PPR (personal project reference) by saving images of their code snippets and inserting them into the document. Students can also create a PDF of their entire code.



The second part will have students think about possible values of the parameter (`my_choice`), and the effect of each value on the code. `My_choice` could have the value 1, and then the heartbeat code would run. `My_choice` could also have the value 2, and then the billboard code would run. `My_choice` could also have the value 0 (or any other value) and then nothing will happen in the function, and the main program will keep executing the loop.

Hopefully students won't need help with either part, but be prepared in case the students don't understand what is being asked and how to select the correct code snippets or values for the parameter.

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.

Formative Assessment:

- Daily reflection journal or Google form
- Completed program
- Completed assignment (wrap-up)
- Exit ticket

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

#### **SUCCESS CRITERIA:**

- Merge to programs into one program that meets the requirements for the Create PT
- Create a function with a parameter
- Use the parameter in an if statement
- Have a loop in the function
- Create a list and store data in the list
- Use (access information from) the the list
- Create a function for intro
- Correctly use global variables in a main program and in functions
- Test and debug the program so that it runs as expected
- Identify possible values for the parameter and their outcomes