Task Overview

About the Task: During the task you will write a program about a topic entirely of your choosing. You will have at least 9 class hours to complete the task, at the end of which you'll need to submit:

- A PDF of your program code
- A video of your program running
- A personalized project reference

About this Guide: By completing the course you should already have the skills and knowledge necessary to complete the task. While the Create Task should be a creative opportunity, **you'll want to make sure you use your time wisely and design your program with the task requirements in mind**. To make sure you're able to do that, this guide includes:

- Activities to highlight tricky parts of the task requirements
- Organizers and planning guides to help you think through ideas
- A recommended schedule for using your 9 hours
- Checklists to make sure your final submission meets all the requirements

College Board Resources: This guide is intended to be a companion to resources provided by the College Board, in particular the task directions and scoring guidelines. Those documents are the final authority on the requirements of the project. This guide, however, should help you understand the nuances of those documents and understand what certain terms mean when designing a program in App Lab.

Getting in the Create PT Mindset: You've been preparing for this moment all year! Make lessons have challenged you to independently decide how to program fully working apps. Unit projects have given you more and more freedom to design projects that take multiple days to program. Throughout the year you've built key skills in collaboration and debugging. The Create Task will challenge you to put all of these different pieces together. **You're ready for this moment because you've been putting in the practice!**

What is Required of My Program?

You have a lot of freedom to design any project you like for the Create Task. The task directions include only a few requirements for what you include as your program. This page summarizes those requirements, highlights the "takeaways", and explains what they might look like in App Lab.

Input / Output Requirements

The video of your program requires you to demonstrate input and output in your program. In App Lab the most straightforward way to do this is to build an app with a user interface. Buttons, dropdowns, or essentially anything that requires you to use an event handler (the onEvent block) is user input. Any time your program plays a sound or shows text or images on the screen that is output.

Takeaway 1: Make sure your program has a user interface
with both input and output.

List Requirements

You'll need to create a list somewhere in your program, and later you'll want to access the information stored in that list. This list could be a hard-coded list, filled in by user input, or be information you pull from the data library using the `getColumn` block.

Takeaway 2: Make sure your program creates
and uses a list of information.

Function Requirements

Your function needs to include "at least one procedure that uses one or more parameters to accomplish the program's intended purpose, and that implements an algorithm that includes sequencing, selection, and iteration." Sequencing merely means that there are multiple lines of code that run in order. Selection means that a conditional, or if-statement, selects between two or more portions of your program to run. Iteration simply means repeating some behavior, as in a loop.

Takeaway 3: Make sure your program includes a function that has
a parameter, an if-statement, and a loop.

Acknowledgment Requirements

The task directions ask that you "Include comments or acknowledgments for any part of the submitted program code that has been written by someone other than you and/or your collaborative partner(s)." In theory, you could continue working on a project you started earlier in the year, or you could use code that you found elsewhere. You'll need to be extremely careful in each case that you indicate what is yours and only respond to questions based on the new code you or your partner wrote. You also need to cite any images or sounds that aren't built into App Lab and that you didn't create.

Takeaway 4: Use comments to cite any code, images, sounds, etc. that
you or your partner did not create yourselves during the task.

Function Requirement Activity - Does It Count (10 mins)

This activity should help you better understand how to design the function you will submit. Remember that your function needs to include "at least one procedure that uses one or more parameters to accomplish the program's intended purpose, and that implements an algorithm that includes sequencing, selection, and iteration." Make sure your program includes a function that has a parameter, an if-statement, and a loop.

Be the AP Reader! You are the AP reader trying to determine if the responses below meet the program requirements for a function. Assume each function below was submitted. For each, select whether it meets the requirements and why. No need for detailed explanations, arrows to the code or short bullets are fine.

Example Algorithm 1
 	
``javascript
function repeatWord(word, times){
  var returnWord = "";
  for(var i = 0; i < times; i++){
    returnWord = returnWord + word;
  }
  return returnWord;
}
``
	Meet requirements? Yes / No
	
	Why?

Example Algorithm 2
 	
``javascript
function increaseScore(points){
 	
  score = score + 1;
  if(score > 10){
    endGame();
  }
}
``
	Meet requirements? Yes / No
	
	Why?

Example Algorithm 3
 	
``javascript
function addList(list){
  var total = 0;
  for(var i = 0; i < list.length; i++){
    total = total + list[i];
  }
  return total;
}
``
	Meet requirements? Yes / No
	
	Why?

Example Algorithm 4
``javascript
function addPositives(list){
  var total = 0;
  var currentItem;
  for(var i = 0; i < list.length; i++){
    currentItem = list[i];
    if(currentItem > 0){
      total = total + list[i];
    }
  }
  return total;
}
``
	Meet requirements? Yes / No
	
	Why?

Narrow it Down

Why Narrow It Down: You should assume that you're not going to have enough time to complete the "perfect" project for the Create PT. While 9 hours may seem like a long time, the majority of your score actually is based on your written responses during the end-of-course exam, and it turns out that even many small or simple projects meet all the requirements listed above. You'll be better set up for success if you "narrow down" project ideas.

How to Narrow it Down: Narrowing it down means taking a larger idea for your Create Task and finding the smallest version of it that will still meet the task requirements. Here are some ways to do it.

- **Identify your function and list early:** Start out with a clear idea of how you'll hit the minimum requirements of your list and your function.

- **Pick One Part of a Bigger Idea:** Often your original big idea can be broken down to smaller ones that meet the requirements of the task.

- **Minimal Design Mode - looks don't matter:** Complex visual design work in Design Mode (setting colors, fonts, spacing, etc.) will likely NOT meet any of the requirements for the Create PT. Don't worry about how your app looks until after you already have code that will let you complete the personalized project reference.

Practice Narrowing it Down (10 mins)

Below are three descriptions of potential projects that another CS Principles student is considering. For each write:
- Two or three ways they could narrow down the project using the tips above
- Opportunities to write an algorithm in their project even after it's been narrowed down.

Project 1: Tic-Tac-Toe
"Here's my idea: I want to build a tic-tac-toe game. The user creates an account if they don't already have one and are taken to the main game board. From there the player will play against the computer in either easy, intermediate, or advanced mode, so I will need to write the code for the computer player. When the game is over their lifetime win total is updated. I will also keep track of how long the game took."

Ways to narrow down the project (2 or 3)	List and Function opportunities

Project 2: Health App
"I volunteer at my local health clinic so I want to build a health app. The user can record information about what they eat, how much they sleep, how much they exercise, and information like their blood pressure and weight. Based on the information provided the app will provide recommendations to the user about how they can improve their health for both diet and exercise. Users can also personalize the look of the app with different theme colors."

Ways to narrow down the project (2 or 3)	List and Function opportunities

Project 3: Sports Stats
"I think that I'll build an app that allows the user to quickly record stats during a basketball game. The app will show a picture of the court. The user taps on the court to indicate something happened there. They are presented with a quick menu of options like: shot attempt, foul, steal, rebound, etc. then they select from another list which player did it. At the end of the game it displays a stat sheet for all of the players and the stats for that game."

Ways to narrow down the project (2 or 3)	List and Function opportunities

Choosing a Project Idea

You should now understand how the Create PT works and are ready to start brainstorming projects. While you have at least 9 class hours to complete the task, keep in mind that in those 9 hours you must also make your video and personalized project reference. We recommend budgeting at least 1 hour to complete the video and personalized project reference, and so it is highly recommended that you prepare to do a project in which the programming / coding can be completed in 7-8 hours. You want projects with the following features.

- **Personally Relevant:** Pick projects you are actually interested in building.

- **Clear Purpose:** Aim for a simple program whose purpose can be stated in one sentence. For example:
    - The purpose of my program is _____.
    - My program lets a user _____.

- **Narrowed Down:** Repeat the "Narrow it Down" process with your own ideas. A good rule of thumb is that you'll want to be able to have a first draft of your algorithm within two hours of starting to program.

- **No New Programming Skills:** Make sure you *already* have the programming skills necessary to complete the project. Be flexible. With some creativity you can likely use the skills you've already learned to make many different types of projects. Avoid taking on new programming environments or concepts as part of the Create PT.

Brainstorm Project Ideas (10 mins)

Brainstorm one or two project ideas for the Create PT. Afterwards you'll share ideas with a classmate for feedback.

Project Idea	Classmate Feedback
Purpose: *Ways to Narrow it Down:* *List Opportunities:* *Function Opportunities:* *Do I already have the skills to make this project?*	*Use the list above to give feedback on the idea.*
Purpose: *Ways to Narrow it Down:* *List Opportunities:* *Function Opportunities:* *Do I already have the skills to make this project?*	*Use the list above to give feedback on the idea.*

Create PT Completion Timeline

Before you start, you should think about how you are going to allocate your time for the 9 hours provided for the task. Below is a sample timeline that you can use to plan out how you will complete the Create Performance Task.

Hour	Suggested Activity	Your Plan
1 - 2	Begin building a program for a project you brainstormed. Carefully monitor whether you will finish enough of your project in time. **Goal:** you should be confident after this first round of development that you'll be able to meet the requirements for your list and your function.	
3 - 4	Keep working. Check in after hour 4 once again on whether you are on track to complete the necessary parts of your program. You should ideally know: ● The list you will include in your **personalized project reference** ● The function you will include in your **personalized project reference**	
5 - 8	Finalize all programming. After this point you should avoid adding new features and focus on debugging and cleaning up your existing code.	
9	Record a **video** of your program running. Complete your **personalized project reference**.	

Note: The timeline above is just a guideline. You may complete the performance task on a different schedule. Make sure to leave enough time to complete your computational artifact and write-up.

Create PT Guidelines

1. Program Code

In your program, you must include student-developed program code that contains the following:
- Instructions for input from one of the following:
  - the user (including user actions that trigger events)
  - a device
  - an online data stream
  - a file
- Use of at least one list (or other collection type) to represent a collection of data that is stored and used to manage program complexity and help fulfill the program's purpose
  - IMPORTANT: The data abstraction must make the program easier to develop (alternatives would be more complex) or easier to maintain (future changes to the size of the list would otherwise require significant modifications to the program code).
- At least one procedure that contributes to the program's intended purpose, where you have defined:
  - the procedure's name
  - the return type (if necessary)
  - one or more parameters
  - IMPORTANT: Implementation of built-in or existing procedures or language structures, such as event handlers or main methods, are not considered student-developed.
- An algorithm that includes sequencing, selection, and iteration that is in the body of the selected procedure
- Calls to your student-developed procedure
- Instructions for output (tactile, audible, visual, or textual) based on input and program functionality

Submit one PDF file that contains all of your program code (including comments). Include comments or acknowledgments for any part of the submitted program code that has been written by someone other than you and/or your collaborative partner(s).

Advice: For resources on how to make a PDF of your program code head to https://studio.code.org/s/csp8-2021/lessons/4/levels/1. Here's the most important things to remember:

- **Avoiding Plagiarism:** If you fail to cite the source of any code that you or your partner didn't write, you will automatically get a 0 on the entire task. Make sure you use comments to cite any code that you or your partner did not write. For example:

  ```
  // This algorithm for finding the middle value in a list taken from
  https://stackoverflow.com/questions/38130895/find-middle-of-a-list/38131003

  // The images used in this app came from:
  // [1] bird image - http://name-of-site.com/path/to/image.jpg
  // [2] flower image - http://site.com/path/to/flower.jpg
  ```

- **Making Your PDF:** Use CodePrint to make a PDF of your program. It's designed specifically for the Create PT. You can find it from the link above.

Program Code Checklist

- ❏ Program has a user interface with both input and output.
- ❏ Program creates and uses a list of information.
- ❏ Program includes a function that has a parameter, an if-statement, and a loop.

2. Video

Your video must demonstrate your program running, including:
- input to your program; and
- at least one aspect of the functionality of your program; and
- output produced by your program.

Your video may NOT contain:
- Any distinguishing information about yourself
- Voice narration (though text captions are encouraged)

Your video:

- must be either .mp4, .wmv, .avi, or .mov format; and
- must not exceed 1 minute in length; and
- must not exceed 30MB in file size.

Collaboration is not allowed during the development of your video.

Advice

- **Making Your Video:** Ask your teacher for suggested resources to make a video in your classroom context. Some ideas are provided in the first lesson of this unit as well.
- **Video Runs Continuously:** Your video must run continuously and show your actual code running. It can't just be a series of screenshots.
- **Show One Piece of Functionality:** Your program does NOT need to be complete so long as you can demonstrate one piece of functionality that includes input and output

Video Checklist

- ❏ Video runs continuously (it cannot be a series of screenshots)
- ❏ Video is less than 60 seconds long and less than 30MB in size
- ❏ Video demonstrates input, program functionality, and output

3. Personalized Project Reference

To assist in responding to the written response prompts on exam day, submit required portions of your code by capturing and pasting program code segments you developed during the administration of this task.
- Screen captures should not be blurry, and text should be at least 10-point font size.
- Your code segments should not include any comments.

These code segments will be made available to you on exam day only if this component is submitted as final in the AP Digital Portfolio by the deadline.

Advice

- **Use Tech Tips Level:** Check out this level for guidance on how to capture screenshots of your code based on

the type of device you are using.
- **Rearrange Comments:** Move any comments that you included inside of your function and/or with the code segments involving your list so that they will not be included in the screenshots.

Procedure: Capture and paste two program code segments you developed during the administration of this task that contain a student-developed procedure that implements an algorithm used in your program and a call to that procedure.

IMPORTANT: Built-in or existing procedures and language structures, such as event handlers and main methods, are not considered student-developed.

- The first program code segment must be a student-developed procedure that:
  - Defines the procedure's name and return type (if necessary)
  - Contains and uses one or more parameters that have an effect on the functionality of the procedure
  - Implements an algorithm that includes sequencing, selection, and iteration

Advice

- **Use this Guide to Pick Your Algorithm:** This Survival Guide has lots of tips for making sure that you have the components necessary to choose an algorithm with the correct components. On a high level you need one function that includes a parameter, a loop, and an if-statement.
- **Pick the Function You Are Most Comfortable Explaining:** If you have multiple functions in your program that meet the requirements, include the one that you can best explain to someone else in your own words.
- **Show Just One Function Definition:** Make sure the screenshot you take shows just the code for a single function definition, nothing more.

(Code Segment Goes Here)

- The second program code segment must show where your student-developed procedure is being called in your program.

Advice

- **Show Just One Function Call:** Make sure the screenshot you take shows just the code for a single function definition, nothing more.

(Code Segment Goes Here)

List: Capture and paste two program code segments you developed during the administration of this task that contain a list (or other collection type) being used to manage complexity in your program.
- The first program code segment must show how data have been stored in the list.

Advice

- **Same List Initialized and Processed:** Be very careful that you only include two code segments and that they are both referring to the same list. Even if the processed data is moved to a new list.

(Code Segment Goes Here)

- The second program code segment must show the data in the same list being used, such as creating new data from the existing data or accessing multiple elements in the list, as part of fulfilling the program's purpose.

Advice

- **Same List Initialized and Processed:** Be very careful that you only include two code segments and that they are both referring to the same list. Even if the processed data is moved to a new list.

(Code Segment Goes Here)