**AP CSP CodeX**

|  |  |  |
| --- | --- | --- |
| **MISSION 13 Sounds Fun** | | **Time: 2 class periods** |
| **Project Goal:** In this project students will create a user-friendly graphical user interface for CodeX so that it can act as a race day controller.  **Learning Targets**   * I can explain what a GUI is used for in a computer application. * I can explain the difference between a local and global variable. * I can use a for loop to iterate over a list. * I can distinguish between a blocking and non-blocking function. * I can use the *not* logical operator accurately in a program. | **Key Concepts**   * Variables defined inside of a function are local variables; variables defined outside of functions are global variables. * The soundlib Python module has functions to create different types of tones, as well as playing recorded samples and MP3s. * The for loop is made for looping across a range of numbers, or iterating over lists. * Blocking functions block your code from continuing until they finish. * Non-blocking functions allow code to continue to run without waiting for sound to finish. * Use the *not* logical operator to flip the state of a global variable. | |
| **Assessment Opportunities**   * Mission 13 Assignment * Race\_Control program | **Success Criteria**   * Display a GUI with four menu items * Use a rectangle to show which menu item is selected * Scroll the selection rectangle up and down to the different menu items * Select a menu item using BTN A * Play music in the background * Play music for starting the race * Play music for finishing the race * Play music as a warning sound | |
| **AP CSP Framework**  **AAP-1.C** Represent a list or string using a variable.  **AAP-2.H** Write conditional statements.  **AAP-2.K** Write iteration statements.  **AAP-2.O** Write iteration statements to traverse a list.  **AAP-3.D** Select appropriate libraries or existing code segments to use in creating new programs.  **Computational Thinking Practice 1.B** Determine and design an appropriate method or approach to achieve the purpose.  **Computational Thinking Practice 3.B** Use abstraction to manage complexity in a program. | **Materials**   * Mission 13 Assignment / Answers * [Mission 13 Obj. 1-6 Kahoot Review](https://create.kahoot.it/share/firia-labs-mission-13-obj-1-6/e367f188-3f89-48a2-ae87-453d43d4e612) * [Mission 13 Obj. 7-11 Kahoot Review](https://create.kahoot.it/share/firia-labs-mission-13-obj-7-11/348820b1-3d5f-4f37-a913-d352eed86540) * Solution code for four objectives, the final and the challenge | |
| **Teacher Notes**   * The assignment is best completed digitally. Prepare the assignment for distributing through your LMS. * Encourage the students to do as much code on their own as they can, and use the CodeTrek to check their work, or as a hint when needed. * The assignment adds an extra step to complete after Objective 11, and a challenge. * You can divide the mission into two days. On the first day, stop after Objective 6 and use [Mission 13 Obj. 1-6 Kahoot Review](https://create.kahoot.it/share/firia-labs-mission-13-obj-1-6/e367f188-3f89-48a2-ae87-453d43d4e612). The second day will complete objectives 7-11 and the final steps. Use [Mission 13 Obj. 7-11 Kahoot Review](https://create.kahoot.it/share/firia-labs-mission-13-obj-7-11/348820b1-3d5f-4f37-a913-d352eed86540). If time permits, include the challenge as well. * Another suggestion for assessment is for students to keep a daily journal, or use a reflection form for students to process information they learned and reflect on questions they may still have. * Refer to the Python with CodeX Curriculum Guide or Mission 13 Lesson Prep (found in the l[earning portal](https://learn.firialabs.com/curricula/python-with-codex/teachers-resources/codex-teacher-materials)) for more information. * The teaching guide (below) gives the narration for one way to present the lesson. | | |

**Teaching Guide**

The actual coding part of this Mission is about two normal class periods. The recommendation is to use two days to complete the mission.

**Warm-up (5 minutes)**

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

* **Topic:** In this mission you will use a list. What do you remember about lists? Have a brief discussion. This mission will also use a global variable that is changed in a function. What do you remember about local and global variables? Have a brief discussion.

**Activity – Objectives 1-6 (40 minutes)**

💻 Randomly group students into pairs for pair programming (or they can work individually).

For pair programming, students log in to one computer. Two computers can be used if they want to have the assignment open on one computer and CodeSpace on the other computer.

Students go to [make.firialabs.com](http://make.firialabs.com) and should be at the beginning of Mission 13.

💡 **Teaching tip – Objective 1:**

This objective is fairly involved. Students create a user interface for the race controller. They start with a function that will be called frequently to draw all the rectangles and text for the graphical user interface. It uses a location, or position, on the screen to draw all the elements. You may need to have a short lesson or review the coordinates on the screen. Students will answer one question in their assignment. Students will create a new file, and then follow CodeTrek to code the user interface.

💡 **Teaching tip – Objective 2:**

Start the objective by having students answer the reflection question. The code will use a list for the y-coordinates. It will be used to move the select rectangle. The x-coordinate never changes; it is always 0, so y is the only one that needs to be included in a list. The code will also use a variable for the list index. CodeTrek will use the max() and min() functions for the first time. You may want to review what these functions do, and give the students some practice. Students will answer two more questions in their assignment.

🗝️ **NOTE:** CodeTrek introduces a new variable to keep track of the previous menu selection. CodeTrek doesn’t specifically stop on this line of code (line 33) but there is a comment about it. You might want to point it out.

🗝️ **NOTE:** The code will cause a new error – with a global variable. It is discussed and fixed in the next objective. But students MUST throw the error to meet the goal of the objective.

💡 **Teaching tip – Objective 3:**

Fix another bug! This time it is that the rectangle moves, but the old one stays. So, erase your tracks!

🗝️ **NOTE:** Students need to be careful where they erase the rectangle. It needs to be BEFORE they draw the new rectangle.

💡 **Teaching tip – Objective 4:**

This objective uses the previous selection variable from Objective 2 to “erase” the rectangle before drawing the new rectangle at “menu index”. Review as needed.

💡 **Teaching tip – Objective 5:**

Students will create two more functions. One function takes a parameter for a message and displays it at the top. The other function checks which button is pressed and calls the first function, passing in a message for the parameter.

💡 **Teaching tip – Objective 6:**

Students make the UI more user-friendly by initializing the screen. Uses a global Boolean variable “init” and modifies the if statement in the while True loop.

Solution code for this objective is provided.

**Acity – Objectives 7-11 (40 minutes)**

💻 Continue either in pairs or individually. Students go to [make.firialabs.com](http://make.firialabs.com) and should be on Objective 7 of Mission 13.

💡 **Teaching tip – Objective 7:**

A lot of information here about the soundlib module. It has functions for creating music and sound effects. First a tone will be selected, and then set a pitch and play. Use stop() to quit playing the tone at the given pitch.

Solution code for this objective is provided.

💡 **Teaching tip – Objective 8:**

This objective introduces the for loop as an alternative to the while loop. You might want to practice with your students. Students learned about for loops while preparing for the Create PT. Review as needed.

💡 **Teaching tip – Objective 9:**

This objective discusses blocking and non-blocking functions. You can go back to Mission 5 and look at the code for playing a sound. It is blocking! Here students will learn a different code for non-blocking. In the hint, it shows an argument to add to the “get\_mp3” statement so the music doesn’t start automatically.

This objective also uses a “toggle” Boolean variable. If this is a difficult concept for your students, you will want to review and practice it. CodeTrek does not specifically stop on this line of code (#45) so you may need to specifically point it out.

Solution code for this objective is provided.

💡 **Teaching tip – Objective 10:**

Students create another sound using a loop within a loop. You might want to step through this part of the code using the debugger to track the variables, or work it out on a white board.

💡 **Teaching tip – Objective 11:**

Students create another sound using the glide function.

Solution code for this objective is provided.

💡 **Teaching tip – After Objective 11:**

Students are asked to add two more pieces of code to the program: add a “kill” switch, and change the color of the selection rectangle.

Solution code for the final program is provided.

💡 **Teaching tip – Challenge:**

If students have time, have them try the challenge. This gives them an opportunity to apply their knowledge in a new situation. The challenge is to light up pixels during the program. Students can use their creativity to add functionality. A code solution showing a function and the function calls is provided, but students should not be required to have the same solution.

Solution code for the challenge is provided.

✅ Assignment is complete and ready to turn in.

**Wrap-Up (5 minutes – optional)**

Discussion: You can now make a GUI on CodeX! This is a very useful skill that you can apply to many other programs. What are some programs you already created that could use a GUI?

Students complete the wrap-up questions on the assignment. You can have a short discussion on their answers.

You can use a formative assessment for the wrap-up.

✅ **IMPORTANT!!**

* Remind students to clear their CodeX.

Formative Assessment:

* Daily reflection journal
* [Mission 13 Obj. 1-6 Kahoot Review](https://create.kahoot.it/share/firia-labs-mission-13-obj-1-6/e367f188-3f89-48a2-ae87-453d43d4e612)
* [Mission 13 Obj. 7-11 Kahoot Review](https://create.kahoot.it/share/firia-labs-mission-13-obj-7-11/348820b1-3d5f-4f37-a913-d352eed86540) (in class or individual)
* Exit ticket on lists
* Exit ticket on global and local variables
* Exit ticket on blocking and non-blocking functions