**AP CSP CodeX**

|  |  |  |
| --- | --- | --- |
| **MISSION 4 Obj 8-10 & Functions Display Games** | | **Time: 45 minutes** |
| **Project Goal:** Students will program CodeX to get input from the user by pressing buttons to create a game.  **Learning Targets**   * I can program buttons to determine whether they are pressed. * I can write an if:else conditional statement. * I can create and use functions as a form of procedural abstraction. | **Key Concepts**   * Indenting after a colon is very important! * Branching allows segments of code to run when a condition is True, and other segments of code to run when the condition is False. * Functions can be used as procedure abstraction to hide details and focus on specific tasks. | |
| **Assessment Opportunities**   * Mission 4 Obj 8-10 & Functions Assignment * Display program (final through Obj. 10) * Display\_functions program * [Mission 4 Kahoot Review](https://create.kahoot.it/share/firia-labs-ap-csp-mission-4/8dcfdc72-a9ba-415b-ba47-5db29d06003e) | **Success Criteria**   * Use a Boolean condition in an if..then statement * Receive input from the user through a button press * Program a button press to make a fast-click game * Create and use functions in the Display program | |
| **AP CSP Framework**  **AAP-2.H** Write conditional statements. Determine the result of conditional statements.  **AAP-3.A** Write statements to call procedures.  **AAP-3.B** Explain how the use of procedural abstraction manages complexity in a program.  **AAP-3.C** Develop procedural abstractions to manage complexity in a program by writing procedures.  **Computational Thinking Practice 3.B** Use abstraction to manage complexity in a program.  **Computational Thinking Practice 3.C** Explain how abstraction manages complexity. | **Materials**   * Mission 4 Obj 8-10 & Functions Assignment / Answers * Mission 4 Functions slides * [Mission 4 Kahoot Review](https://create.kahoot.it/share/firia-labs-ap-csp-mission-4/8dcfdc72-a9ba-415b-ba47-5db29d06003e) * Solution code for Display (Objective 10) * Solution code for Display\_functions * AP CSP CodeX Vocabulary List * AP CSP CodeX Python Code List * Unit 1 Review Links and Test Questions | |
| **Teacher Notes**   * This is the second part of Mission 4, and it includes additional instructions for creating and using procedures. * The assignment is best completed digitally. Prepare the assignment for distributing through your LMS. * The second half of the assignment starts after the objectives are completed. Open the slides, or have students work through the slides, to create and use functions in the program. * Students should keep the Display program open for the second half, and save it with a new name when following the instructions on the slides. * Students should use the Sandbox for the added instructions. However, if they stay in Mission 4 Objective 10, that is okay. * At the end of the lesson, discuss clearing the CodeX before turning it in. You can use the Clearing CodeX slides. * 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. * You may consider having students (or the class collectively) keep a chart of errors and the ways to fix them. * You can also add vocabulary to a word wall and keep a document or chart of the Python code learned during each mission. * Refer to the Python with CodeX Curriculum Guide or Mission 4 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**

**Warm-up (5 minutes)**

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

Review with students the concepts from Mission 4 completed in the last lesson.

* Assigning a value to a variable using math.
* Boolean variables.
* Branching.
* Indenting.

**Activity – Mission #4 Objectives 8-10 (20 minutes)**

💻Students can get new partners for pair programming, or continue to work with the same partner they had with Mission 4 Obj 1-7. Or they can work individually.

If students are working with a partner, students log in to one computer. Two computers can be used if they want to have the activity guide and slides open on one computer and CodeSpace on the other computer.

Students go to [make.firialabs.com](http://make.firialabs.com) and should be on Objective 8 of Mission 4.

💡 **Teaching tip – Objective 8:**

This Objective does not need CodeX. Students use the 3D simulator to identify the 6 CodeX buttons.

💡 **Teaching tip – Objective 9:**

Students have a question to answer on the activity guide.

💡 **Teaching tip – Objective 10:**

CodeTrek shows code for two of the buttons. Students can choose any two buttons they want. They can even use the same button twice. Then they write code for two more buttons. At this point the instructions in CodeSpace are complete. You can have students turn in the code at this point for assessment, if you want.

**Activity – Mission #4 Functions (20 minutes)**

💻Students continue working with a partner or individually.

💻The “Mission 4 Functions” slides are needed for the rest of the lesson. You can show the slides on a big screen, or have students follow the slides on their computers.

💡 **Teaching tip – Slides 1-6:**

Review functions and abstraction, with an emphasis on procedural abstraction. Students answer questions in the activity guide.

💡 **Teaching tip – Slides 7-14:**

Students save the Mission 4 program Display with a new name: Display\_functions. Then follow the instructions on slides 7-14 to modify their code using functions. Students should use the Sandbox for this part of the lesson, but if they stay in Mission 10 objective 10 that will work fine, too. This part of the lesson reinforces the concepts learned and used in “Defining Functions” after Mission 3.

If students have time, some extension ideas are given to keep students engaged and applying their knowledge. All are optional, but a great opportunity for student learning.

✅ Assignment is complete and ready to turn in. Both students should include their names on the document. Students should turn in their completed program.

**Wrap-Up (5 minutes)**

✅ **IMPORTANT!!**

* Remind students to clear their CodeX.

Formative Assessment:

* Daily reflection journal
* [Mission 4 Kahoot Review](https://create.kahoot.it/share/firia-labs-ap-csp-mission-4/8dcfdc72-a9ba-415b-ba47-5db29d06003e)
* Exit ticket on procedural abstraction
* Group or class discussion on what they learned in the lesson

**SUCCESS CRITERIA:**

* Get input from pressing a button
* Use a branching if:else statement with a Boolean variable
* Create a function for each button press
* Call the functions in a specific order to create a game