**AP CSP CodeX**

|  |  |  |
| --- | --- | --- |
| **MISSION 4 Obj 1-7 Display Games** | | **Time: 45 minutes** |
| **Project Goal:** Students will learn about variable data types and use display.print() to display multiple messages.  **Learning Targets**   * I can print multiple messages on the screen. * I can convert between integers and strings. * I can identify a variable data type. * I can write an if:else conditional statement. | **Key Concepts**   * Arguments are introduced to students when calling built-in functions * Students write code to perform a calculation. * Values have different data types, like *integer* and *string*. * Indenting after a colon is very important! | |
| **Assessment Opportunities**   * Mission 4 Obj 1-7 Assignment * Display program (through Obj. 7) | **Success Criteria**   * Define and use an argument in a function call * Understand and use variable types, converting types when needed * Use a Boolean condition in an if..then statement | |
| **AP CSP Framework**  **AAP-1.B** Determine the value of a variable as a result of an assignment.  **AAP-2.H** Write conditional statements. Determine the result of conditional statements.  **Computational Thinking Practice 3.A** Generalize data sources through variables.  **Computational Thinking Practice 4.C** Identify and correct errors in algorithms and programs, including error discovery through testing. | **Materials**   * Mission 4 Obj 1-7 Assignment / Answers * AP CSP CodeX Vocabulary List * AP CSP CodeX Python Code List * Unit 1 Review Links and Test Questions | |
| **Teacher Notes**   * Start the lesson by going over the CodeX Mission Reminders slides. * Mission 4 may take a little over a class period, so it is divided up into two lessons. * The assignment is best completed digitally. Prepare the assignment for distributing through your LMS. * If students take longer for the objectives, consider adjusting the assignment to end in one class period, even if you stop before Obj. 7. If students take less time, you can continue in Mission 4. There is additional instructions at the end of Mission 4 for creating and using functions. * 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 3 – variables, RGB and pixels. If you have time, you could use the [Mission 1, 2, 3 Kahoot Review](https://create.kahoot.it/share/firia-labs-ap-csp-mission-1-2-3/5be3baab-3370-49ae-8912-adf30394f2bd), or something similar

**Activity – Mission #4 Objectives 1-7 (40 minutes)**

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

Students log in to one computer. Two computers can be used if they want to have the activity guide open on one computer and CodeSpace on the other computer.

💡 **Teaching tip – Before they start:**

Review the Mission Reminders slides.

Students go to [make.firialabs.com](http://make.firialabs.com) and should be at the beginning of Mission 4

💡 **Teaching tip – Objective 1:**

The term “**argument**” is introduced. Students should click on the wrench to see documentation. The documentation includes positional and keyword arguments. They don’t need to worry about that – only the definition at the beginning of the paragraphs. You may want to expand on the definition (see definition above) and give examples.

* sleep(1) – argument is 1
* sleep(delay) – argument is delay
* pixels.set(0, color) – arguments are 0 and color
* display.show(pics.PLANE) – argument is pics.PLANE

💡 **Teaching tip – Objective 3:**

Students have to add a line of code to calculate num and then use num in the display command. This is a possible source of confusion. They can use CodeTrek if they get frustrated.

Also, they will get an error in their code. The instructions clearly state this, but if students are not reading the instructions they may think they are doing something wrong. Also, students should read the Hint, which discusses the error they will get.

💡 **Teaching tip – Objective 5:**

The code is running **sequentially**, so only “world” shows on the screen. You may want to emphasize this word because it is used on the AP CSP create performance task prompts.

💡 **Teaching tip –Objective 7:**

This is the first time students use a Boolean condition and if statement. The instructions suggest they use the debugger to step through the code and see what it is doing. You may want to guide them through this, or you may believe they are understanding without stepping through. More unplugged practice will be coming with if statements.

✅ Assignment is complete and ready to turn in. Both students should include their names on the document. You can decide if you want students to turn in the program up to this point, or wait until the end of Mission 4.

**Wrap-Up (5 minutes)**

✅ **IMPORTANT!!**

* Remind students to clear their CodeX.

Formative Assessment:

* Daily reflection journal
* [Mission 1, 2, 3 Kahoot Review](https://create.kahoot.it/share/firia-labs-ap-csp-mission-1-2-3/5be3baab-3370-49ae-8912-adf30394f2bd) (in class or individual):
* Exit ticket on data types, Boolean, or branching.
* Group review on data types, Boolean, or branching.

**SUCCESS CRITERIA:**

* Define and use an argument in a function call
* Understand and use variable types, converting types when needed
* Write an if:else statement using a Boolean variable.