**AP CSP CodeX**

|  |  |  |
| --- | --- | --- |
| **MISSION 6 Obj 1-7 Heartbeat** | | **Time: 45 minutes** |
| **Project Goal:** Students will use images in an infinite loop to create a beating heart animation.  **Learning Targets**   * I can create an infinite loop to make my code more efficient. * I can code a “kill switch”. | **Key Concepts**   * Infinite loops – when a *condition* is always **True**. * A function can be used anywhere, including the code in an infinite loop. * Readability is very important in coding. Whitespace can help, and comments are essential. | |
| **Assessment Opportunities**   * Mission 6 Obj 1-7 Assignment * Heart2\_functions program (through Obj. 7) | **Success Criteria**   * Create a program that shows a beating heart using an infinite loop * Create a function for the code that shows a beating heart | |
| **AP CSP Framework**  **AAP-1.J** Write iteration statements. Determine the result or side effect of iteration statements.  **Computational Thinking Practice 3.B** Use abstraction to manage complexity in a program.  **Computational Thinking Practice 4.C** Identify and correct errors in algorithms and programs, including error discovery through testing. | **Materials**   * Mission 6 Obj 1-7 Assignment / Answers * AP CSP CodeX Vocabulary List * AP CSP CodeX Python Code List * Unit 2 Review Links and Test Questions * [Mission 6 Kahoot Review (Questions 1-9)](https://create.kahoot.it/share/firia-labs-mission-6/7bf069b2-892b-4db9-89f7-10738cbdbc63) | |
| **Teacher Notes**   * Start the lesson by going over the CodeX Mission Reminders slides. * Mission 6 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. * After Objective 7, students continue with the code to create and use a function. This is extended beyond the instructions in CodeSpace. Students can stay in Obj. 7 or move to the sandbox. Before creating the function, a recommendation is given to save the program with a new name to keep the original code. This step is not vital to the program. * At the end of the lesson, discuss clearing the CodeX before turning it in. * If you have time at the end of the lesson, use the [Mission 6 Kahoot Review](https://create.kahoot.it/share/firia-labs-mission-6/7bf069b2-892b-4db9-89f7-10738cbdbc63). The first nine questions are covered in Objectives 1-7. You can hold off on the rest of the review until after Mission 6 is completed. * 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 6 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.

* Topic: Ask students if they’ve seen flashing road signs. Can they think of other continuously blinking or repeating indicators? How would you make something repeat forever with code?

**Activity – Mission #6 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 6.

💡 **Teaching tip – Objective 2:**

CodeTrek will remind students about adding the sleep command, but they still need to replace the #TODO with the actual code for displaying a small heart. It doesn’t show the code for that.

💡 **Teaching tip – Objective 3:**

At this point, we suggest introducing the students to features of the Editor they may not be aware of, such as **cut**, **copy**, **paste**, and **undo**. There is a Toolbox tool covering these “Editor Shortcuts”, and they’re also explained in a video at <https://youtu.be/WItk_kpkGiU>. This video does use an older version of CodeSpace. A couple of other handy editor shortcuts include undo, indent and comment/uncomment. These shortcuts are on the assignment. The comment / uncomment is extremely helpful in commenting out large sections of code, especially during testing.

💡 **Teaching tip – Objective 4:**

You can demonstrate this keyboard shortcut: to quickly indent code that is already typed, you can highlight the code and press TAB, and to un-indent, shift-TAB.

Remind students: You only need ONE heartbeat cycle in the loop, so remember to delete all the extra code they added for Objective #3.

💡 **Teaching tip – Objective 7:**

This objective has students use the debugger. You may need to demonstrate this or remind them how to use it.

💡 **Teaching tip – After Objective 7:**

For extra practice, students should create a function for the code inside the while True: loop (not the button press). Then call the function in the infinite loop. Students should be able to do this step on their own. I recommend doing a “Save-As” and renaming the file Heart2\_functions. But if they don’t do it, that is okay.

After creating the function, students are also asked to change the kill switch from BTN\_A to BTN\_U. This is because in the next part of Mission 6, students will program BTN\_A to do something else, and students will want to keep the kill switch.

Students should test their code and debug any errors before submitting and going to the next lesson.

✅ Assignment is complete and ready to turn in. You can decide if you want students to turn in the program up to this point, or wait until the end of Mission 6.

**Wrap-Up (5 minutes)**

Use a formative assessment for the wrap-up.

✅ **IMPORTANT!!**

* Remind students to clear their CodeX.

Formative Assessment:

* Daily reflection journal
* [Mission 6 Kahoot Review](https://create.kahoot.it/share/firia-labs-mission-6/7bf069b2-892b-4db9-89f7-10738cbdbc63) – questions 1-9 (in class or individual)
* Exit ticket on loop, while loop or infinite loop.
* Group review on loops.

**SUCCESS CRITERIA:**

* Create a program that shows a beating heart using an infinite loop
* Include a “kill switch” to stop the program (BTN\_U)
* Create and call a function for the beating heart animation