**AP CSP CodeX**

|  |  |  |
| --- | --- | --- |
| **MISSION 8 Answer Bot** | | **Time: 45 minutes** |
| **Project Goal:** Students will use a list and a while loop to create a playlist or a bank of random answers.  **Learning Targets**   * I can create a list to use abstraction in my code. * I can apply properties of lists in my program. * I can use randomization with lists. * I can create and call a function. | **Key Concepts**   * Random number generators are crucial for many computer applications. * Lists can be used for a variety of purposes and uses in a program. | |
| **Assessment Opportunities**   * Mission 8 Assignment * Answer\_bot program (with or without additions) | **Success Criteria**   * Program a button to print a random number * Select a random item from a list and print it * Select a random color from COLOR\_LIST and use it to light up pixels * Create a function and call it in the main program | |
| **AP CSP Framework**  **AAP-1.C** Represent a list or string using a variable.  **AAP-1.D** Develop data abstraction using lists to store multiple elements, and explain how the use of data abstraction manages complexity in program code.  **AAP-2.N** Write expressions that use list indexing and list procedures.  **AAP-3.E** Write expressions to generate possible values, and evaluate expressions to determine the possible results.  **Computational Thinking Practice 3.A** Generalize data sources through variables.  **Computational Thinking Practice 3.B** Use abstraction to manage complexity in a program. | **Materials**   * Mission 8 Assignment / Answers * AP CSP CodeX Vocabulary List * AP CSP CodeX Python Code List * Unit 2 Review Links and Test Questions * [Mission 8 Kahoot Review](https://create.kahoot.it/share/firia-labs-mission-8/6df93bf9-a83a-444e-929d-65b187437f64) * Solution code   + Answer\_bot (after objective 7)   + Answer\_bot\_additions   + Answer\_bot\_challenge | |
| **Teacher Notes**   * The assignment is best completed digitally. Prepare the assignment for distributing through your LMS. * This lesson introduces random numbers: different ways to generate them and use them in code. * The mission includes a lot of code review, as well as new code. 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. * Objective 5 has the students create a list. Encourage students to use their own lists, and not the one in CodeTrek. Also encourage them to have more than three items. * After Obj 7, students will practice programming skills by creating a function, adding a kill switch, and clearing the screen and pixels at the end of the program. If students don’t get all of the additions complete, that is okay. But they should try. * An additional challenge is given for any students finishing the program and having extra time. It is not required, but good for extra practice with iteration, which is required for the Create PT. * If you have time at the end of the lesson, use the [Mission 8 Kahoot Review](https://create.kahoot.it/share/firia-labs-mission-8/6df93bf9-a83a-444e-929d-65b187437f64). * 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 8 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:** This is a project with *practical* applications. In our previous program, the programmer or the person running the program had control over when something happened or what image appeared. However, in real life and computer applications, we might have a lot of randomness
* What are some examples of when we need something randomized?

**DISCUSS REAL WORLD APPLICATIONS:**

* Video games
* Secure password encryption
* Real-world simulator trainers
* Scientific statistical sampling

**Activity – Mission #8 (30 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.

Students go to [make.firialabs.com](http://make.firialabs.com) and should be at the beginning of Mission 8.

💡 **Teaching tip – Objective 1:**

The code in this objective will cause an error. The instructions clearly state this, but students aren’t always good at reading the instructions. The assignment document asks them if they know what the error is and how to fix it. They will do this on the next objective, but hopefully they remember before the instructions help them.

💡 **Teaching tip – Objective 3:**

This objective shows students an additional parameter to display.print(). I recommend using the keyword “parameter” so they become familiar with it. The instructions have them use scale=3. They can try different numbers for the scale. When the scale makes the text too big for the screen, it will turn into strange lines and not be readable. When this happens, they need to adjust the scale smaller.

💡 **Teaching tip – Objective 5:**

Students create a list. The list can contain any text that is school appropriate. The example is for food, but it can be anything they are interested in (sports, fashion, food, games, etc.)

💡 **Teaching tip – Objective 6:**

Students will use another list, but they DO NOT need to create it. The instructions show the list, but they do not need to type it. The list used has all the built-in colors for the neoPixels, and the list is already created. Students simply need to access it: COLOR\_LIST.

**Activity – After Objective 7 (10 minutes)**

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

With this activity, students have a chance to practice some of the programming skills they have been learning.

Spend the time you have on these additions, but if students don’t do them all, that is okay. They can be done in any order.

💡 **Teaching tip – Addition #1:**

Clear the screen before printing an item in the list. Just add “display.clear() before the display.print() statement.

💡 **Teaching tip – Addition #2:**

Create a function to change the pixel colors. The name of the function should be descriptive. Then students can cut and paste the code from the main program to the function, and call the function in the main program.

💡 **Teaching tip – Addition #3:**

Add a kill switch. Students have done this a couple times already, so they should be able to do this without a problem.

💡 **Teaching tip – Addition #4:**

After the program ends (pressing the kill switch), clear the screen and the pixels. One way is to use display.clear() and pixels.off(). This code is NOT inside the while True: loop, or indented.

Example solutions are given for the program as given in the mission, with additions, and with the challenge.

✅ Assignment is complete and ready to turn in.

**Wrap-Up (5 minutes – optional)**

Use a formative assessment for the wrap-up.

✅ **IMPORTANT!!**

* Remind students to clear their CodeX.

Formative Assessment:

* Daily reflection journal
* Mission 8 Kahoot Review (in class or individual)
* Exit ticket or group review on generating random numbers

**SUCCESS CRITERIA:**

* Program a button to print a random number
* Select a random item from a list and print it
* Select a random color from COLOR\_LIST and use it to light up pixels
* Create a function and call it in the main program