Complete the following assignment and push your file to GitHub for reviewing by the instructor. Refer to [GitHub Guide](#) for any inquiries

The homework assignments for the next few lectures will be inter-connected. By the end of JavaScript part 3, you will have developed a fully functional chatbot. These assignments will build on the previous week's assignment, therefore, it is very important that you complete the assignment in a timely manner (i.e. by the due date).

CHATBOT PART III

In this third assignment, you will be adding more features to the chatbot application you created in the previous weeks (in part I & II). In part 3 you will be making two additions to your chatbot application:

1. Update your initial object variable so that the **'input'** property is an array (similar to how we changed the **'output'** property into an array in part 2. In each input array you will include at least 3 similar input statement (synonyms of what you had before). For example, if you object looked like this before:

   ```
   {
           input: 'Hello',
           output: ['Hello', 'Hey', 'Greetings']
   },
   {
            input: 'What is your favourite colour?',
           output: ['I am not sure.', 'There are too many to choose from.', 'I like everyone.']
   },
   {
            input: 'How are you?',
           output: ['Fine', 'Great', 'Not so good']
   }
   ```

   It should now look like this:

   ```
   {
           input: ['Hello', 'Hi', 'Greetings'],
           output: ['Hello', 'Hey', 'Greetings']
   },
   {
           input: ['What is your favourite colour?', 'Who is your favourite HYF instructor?', 'Who is your role model?'],
           output: ['I am not sure.', 'There are too many to choose from.', 'I like everyone.']
   },
   {
           input: ['How are you?', 'How is the weather today?', 'How is Canada doing in the Olympics?'],
   ```

output: ['Fine', 'Great', 'Not so good']
}

Now when the user enters an input, you should search the input arrays for possible answers. HINT: you can iterate over the array, or use the array method include() that we covered in class. For more info on the array include() method, check these links:
- https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/includes
- https://www.w3schools.com/Jsref/jsref_includes_array.asp

2. The second item we will be updating is how the responses are displayed. Currently, you should only have the most recent reply displayed--the text is replaced every time with the new reply. So something like this:

My first chatbot!

Talk to your bot!

Hello   Submit ⦿ Random ◯ Shortest Answer ◯ Longest Answer

Chat history

Hey

My first chatbot!

Talk to your bot!

What is your favourite colour?   Submit ⦿ Random ◯ Shortest Answer ◯ Longest Answer

Chat history

There are too many to choose from.

Change this so that the previous text remains in the text area. This way we build-up a chat history. Hint: remember is you assign a value, it will overwrite the previous value. Ex: reply = "Hey"; // Hey

reply = "There are too many to choose from." //There are too many to choose from.
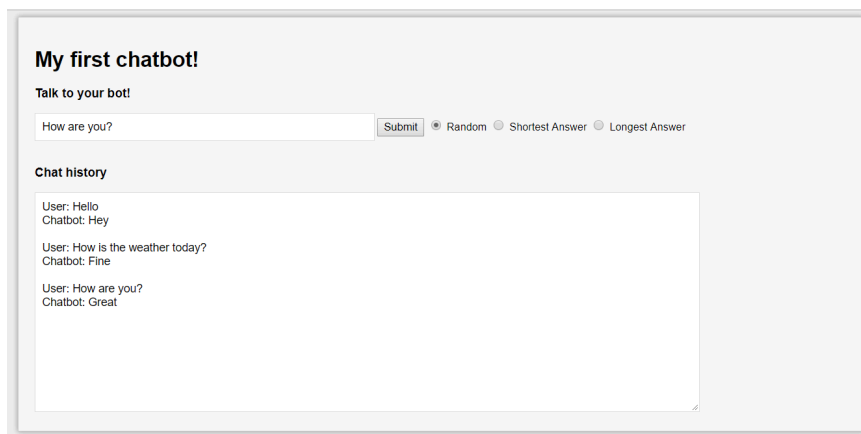
So what you want to do is to keep appending (ADDING) the new replies so that the previous replies also remain. The easiest method to append text is using the '+' operator.

Ex: reply = "Hey"; // Hey

reply = reply + " There are too many to choose from."

//Hey There are too many to choose from

If done correctly, you should get something like this:

My first chatbot!

Talk to your bot!

How are you?   Submit ◉ Random ◯ Shortest Answer ◯ Longest Answer

Chat history

User: Hello
Chatbot: Hey

User: How is the weather today?
Chatbot: Fine

User: How are you?
Chatbot: Great

SUBMISSION

Before submitting your assignment, remember to organize your code according to the best practices for structuring code files, as explained in the class. This means:
- HTML, CSS, and JS code should be in separate files
- All files should be organized into their respective folders

EVALUATION

When evaluating this assignment, we will be looking for:
- Proper code formatting
- Correct structuring of code files
- Descriptive comments within code
- Functionality of the chatbot (if it's working or not)