

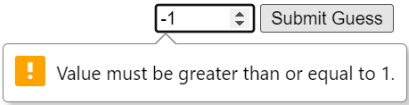
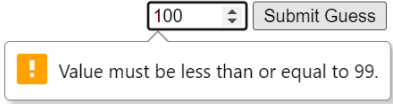
TEST SCRIPT FOR GUESSING GAME	
NAME OF TESTER:	Saul Castillo
DATE – START AND FINISH:	January 9 th , 2022, 20:36 P.M to 22:30 P.M
VERSION OF SPECIFICATION TESTED AGAINST:	Readme.MD - December 17 th , 2022, 18:51 P.M
VERSION OF CODE TESTED:	1.1 – Forked version with comments.
SYSTEM TESTED ON:	<ul style="list-style-type: none"> PC O.S: Windows 11 CROME BROWSER VERSION: 108.0.5359.125 (Official Build) (64-bit)

TEST SCRIPT FOR GUESSING GAME					
STEP	REQUIREMENT	ACTION	PASS/FAIL	COMMENTS	FIX
1	No errors in the browser console.	Open app in browser with console open.	Failed	Uncaught SyntaxError: Unexpected token '=' at index.js:63:19: <ul style="list-style-type: none"> If condition using wrong comparison operator ("====") 	Fixed comparison operator from ("====") to ("===").
			Failed	Uncaught SyntaxError: Unexpected identifier 'setup' (at index.js:79:9): <ul style="list-style-type: none"> Misspelled "funtion" keyword is being used. 	Fixed misspelled "funtion" keyword to "function".
			Failed	Uncaught TypeError: Assignment to constant variable. at setup (index.js:85:23) at index.js:98:1: <ul style="list-style-type: none"> maxNumberOfAttempts is being declared as constant but it is being assigned a new value on setup function. 	Changed maxNumberOfAttempts declaration from const to let.
			Failed	Uncaught TypeError: Cannot read properties of undefined (reading 'style') at hideAllMessages (index.js:75:28) at setup (index.js:91:3) at index.js:98:1: <ul style="list-style-type: none"> For index finishing on undefined nodeList position. 	Fixed for counter on line 74: counter was finishing on " <code><= messages.length</code> ". Changed the operator to " <code>< messages.length</code> "
			Passed	No errors on JS console when opening the game.	-
2	At the start, the "reset" button and all of the "messages" should be hidden.	Open app in browser.	Passed	Reset button and messages are hidden.	Fixed by previous tests and fixes.

3	You should be able to type a number into the input field and click "Submit Guess" to submit the form.	Type a number into the input field.	Passed	Game lets the user input a number into the guess number input field.	-
		Scroll the guess number input field to check limits from 1 to 99	Passed	Guess number input field respects the limits of the game: starts on 1 and finishes on 99.	-
		Type a number out of the limits from 1 to 99	Failed	<p>The guess number input field lets the user type a number out of the limits from 1 to 99.:</p> <ul style="list-style-type: none"> A form is required to wrap the input controls in order to let the validations within the inputs work. 	<ul style="list-style-type: none"> Index.html:18: Wrapped inside a form the input controls. Form has id="guessNumberForm". Index.js:3: Changed <i>submitButton</i> declaration to <i>guessNumberForm</i>. Made the <i>guessNumberForm</i> assigned to the <i>guessNumberForm</i> form HTML element. Added on the next line a <i>submitButton</i> declaration. Index.js checkGuess function: added <code>event.preventDefault();</code>. Index.js:88: Changed the variable name to <i>guessNumberForm</i>. Also corrected "disabled" property that was misspelled. Index.js:95: Assigned the event listener to the <i>guessNumberForm</i> variable. Changed the event to 'submit'.
			Passed	Since the input controls are now inside a form, the validations are working as expected before submitting a number.	-
		Submit the guess number	Failed	<p>Uncaught TypeError: Assignment to constant variable.</p> <p>at HTMLButtonElement.checkGuess (index.js:30:12):</p> <ul style="list-style-type: none"> <i>attempts</i> is being declared as <i>constant</i> but it is being assigned a new value on <i>checkGuess</i> function. 	Changed <i>attempts</i> declaration from <i>const</i> to <i>let</i> .
			Failed	<p>Uncaught TypeError: Cannot read properties of null (reading 'style')</p> <p>at HTMLButtonElement.checkGuess (index.js:58:28):</p> <ul style="list-style-type: none"> <i>numberOfGuessesMessage</i> declaration is being assigned an undefined HTML element ID "<i>num-of-guesses</i>". 	Changed <i>numberOfGuessesMessage</i> declaration to store the correct HTML element ID to " <i>number-of-guesses</i> ".
			Passed	Game now lets the user input a number on the guess number input and lets submit the guess number.	-

4.	<p>When the form is submitted, you should see the following:</p> <ul style="list-style-type: none"> ○ A message displaying the number that was entered. ○ A message displaying how many tries you have left (starts at 5 and decrements by 1). ○ A message describing the guess (low, high, etc.). ○ A “reset” button that restarts the game. 	Submit the guess number.	Failed	<p>Message displaying how many tries are left displays the wrong guesses remaining:</p> <ul style="list-style-type: none"> ○ index.js:85 shows that the code is resetting the <i>maxNumberOfAttempts</i> to 0 instead of the <i>attempts</i> variable. 	Fixed line 85 in order to reset the <i>attempts</i> variable instead of the <i>maxNumberOfAttempts</i> variable.
			Failed	<p>Message displaying if the guess number is “lower or higher” is not displaying the correct message depending on the guess number input:</p> <ul style="list-style-type: none"> ○ index.js:53 shows that the code is displaying the <i>tooLowMessage</i> when the guess number is higher instead of showing the <i>tooHighMessage</i>. 	Fixed line 53 in order to show the correct message (<i>tooHighMessage</i>) instead of <i>tooLowMessage</i> .
			Passed	<p>Game now shows the expected messages and button when submitting a guess number.</p>	-
5	If the guessed number is BELOW the target, the message should say “too low”	Submit a guess number below the target.	Passed	<ul style="list-style-type: none"> ○ Target number: 40. ○ Guess number: 4. ○ Result: <i>You guessed to low. Try again.</i> 	Fixed by previous steps.
6	If the guessed number is ABOVE the target, the message should say “too high”.	Submit a guess number above the target.	Passed	<ul style="list-style-type: none"> ○ Target number: 40 ○ Guess number: 50 ○ Result: <i>You guessed too high. Try again.</i> 	Fixed by previous steps.
7	<p>If the guessed number is the SAME as the target, then:</p> <ul style="list-style-type: none"> ○ The input field and “Submit Guess” button should be disabled. ○ The message should say “guessed correctly”. 	Submit the same number as the target number.	Passed	<ul style="list-style-type: none"> ○ Target number: 48 ○ Guess number: 48 ○ Result: <i>Congratulations, You guessed correctly! Would you like to play again?</i> ○ <i>Input field</i> and <i>Submit Guess</i> button are disabled. ○ Message is being displayed correctly. 	Fixed by previous steps.

8	<p>If the guessed number is not the same AND all 5 tries have been used, then:</p> <ul style="list-style-type: none"> The input field and "Submit Guess" button should be disabled. The message should say "0 guesses remaining". 	Make 5 wrong attempts on guessing the target number.	Passed	<ul style="list-style-type: none"> <i>Input field</i> and <i>Submit Guess</i> button are disabled. The message displays the expected output of "0 guesses remaining". 	Fixed by previous steps.
9	When you click the "reset" button, the form should return to its initial state (not disabled).	Click <i>reset</i> button.	Passed	<ul style="list-style-type: none"> <i>Reset</i> button returns the game to its initial state (not disabled). 	Fixed by previous steps.

TEST SCRIPT FOR GUESSING GAME – STRETCH GOALS					
STEP	REQUIREMENT	ACTION	PASS/FAIL	COMMENTS	FIX
1	You should not be able to submit a guessed number lower than 1.	Input a number lower than 1	Passed	<p>Input control automatically validates the limits from 1 to 99.</p>  <p>This can also be fixed by adding a condition on <i>checkGuess</i> function that checks if the number is lower than 1:</p> <ul style="list-style-type: none"> If the condition is true, then use one of the <p> messages to display the error styled with a red font. Do not increment the attempts, do not show the remaining attempts, do not show if the number is below or above the target number and do not show the reset button. 	Fixed on previous steps by wrapping the input controls inside a form.
2	You should not be able to submit a guessed number higher than 99.	Input a number higher than 99.	Passed	<p>Input control automatically validates the limits from 1 to 99.</p>  <p>This can also be fixed by adding a condition on <i>checkGuess</i> function that checks if the number is higher than 99:</p> <ul style="list-style-type: none"> If the condition is true, then use one of the <p> messages to display the error styled with a red font. Do not increment the attempts, do not show the remaining attempts, do not show if the number is below or above the target number and do not show the reset button. 	Fixed on previous steps by wrapping the input controls inside a form.

3	If there is only one guess left, it should say "guess" (singular) instead of "guesses" (plural).	Attempt 4 guesses. Stop at 1 guess remaining.	Failed	When 1 attempt is left, the message shows "1 <i>guesses remaining</i> " instead of "1 <i>guess remaining</i> ".	On index.js, after line 60, it's required to add the following condition: if the remaining attempts are lower or equal to 1, then show <i>guess</i> . Else, show <i>guesses</i> .
			Passed	Successfully fixed the issue by adding the previous condition: <div> <input type="text"/> <input type="button" value="Submit Guess"/> </div> You guessed 1. 1 guess remaining	-