

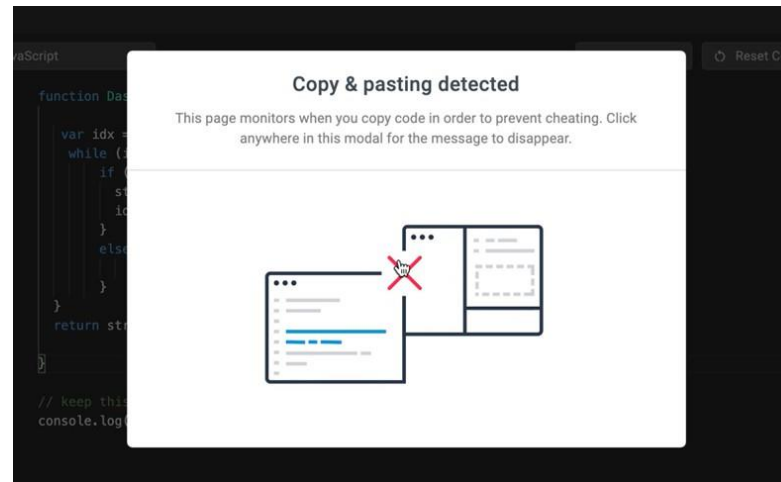
# CoderByte Guideline for Candidate

*FPT Software Academy*



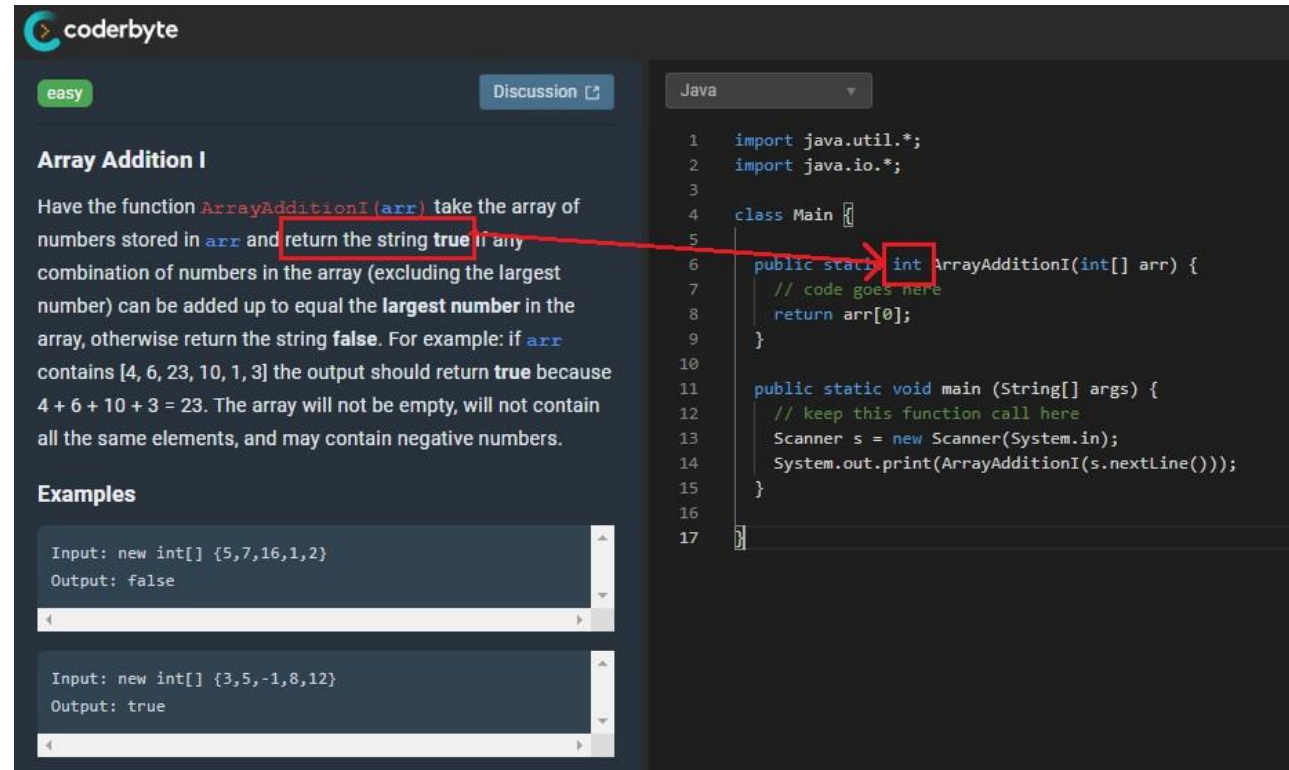
# Before taking the Test

- CoderByte will **record your coding work as a video** during the test, so:
  - Only work on the CoderByte editor, do not use your personal IDE.
  - Avoid opening any other tabs or software applications (chat, email, etc...), the system will monitor and count the number of times.
  - Avoid copying and pasting code from any external sources (such as the internet or personal IDE...). The system will issue a warning and keep a record of any copy/paste actions.



# Before taking the Test

- Review the question requirements carefully. You may need to modify the returned data type to match the required output.



**Array Addition I**

Have the function `ArrayAdditionI(arr)` take the array of numbers stored in `arr` and return the string `true` if any combination of numbers in the array (excluding the largest number) can be added up to equal the largest number in the array, otherwise return the string `false`. For example: if `arr` contains `[4, 6, 23, 10, 1, 3]` the output should return `true` because  $4 + 6 + 10 + 3 = 23$ . The array will not be empty, will not contain all the same elements, and may contain negative numbers.

**Examples**

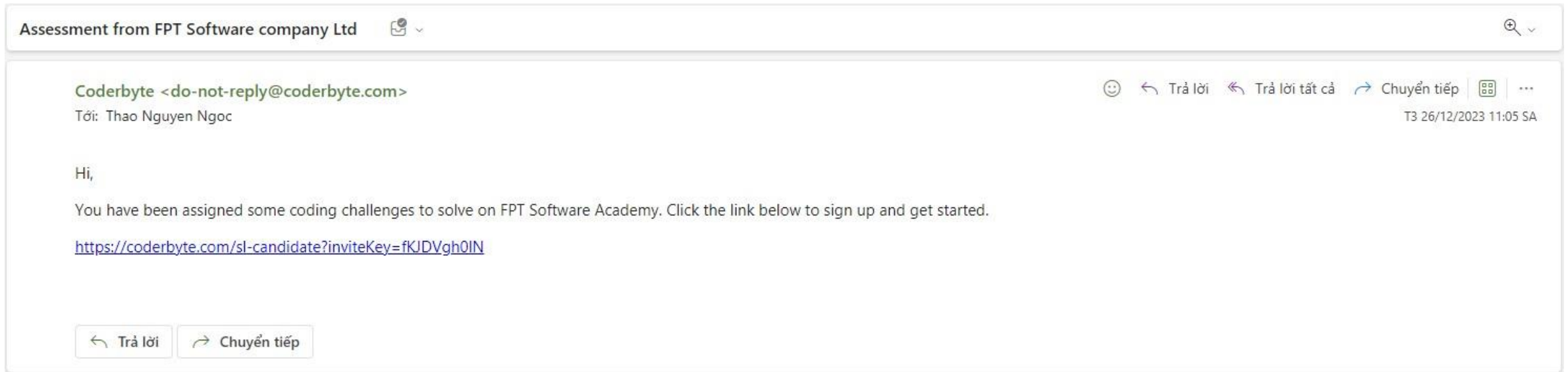
Input: `new int[] {5,7,16,1,2}`  
Output: `false`

Input: `new int[] {3,5,-1,8,12}`  
Output: `true`

```
1 import java.util.*;
2 import java.io.*;
3
4 class Main {
5
6     public static int ArrayAdditionI(int[] arr) {
7         // code goes here
8         return arr[0];
9     }
10
11     public static void main (String[] args) {
12         // keep this function call here
13         Scanner s = new Scanner(System.in);
14         System.out.print(ArrayAdditionI(s.nextLine()));
15     }
16
17 }
```

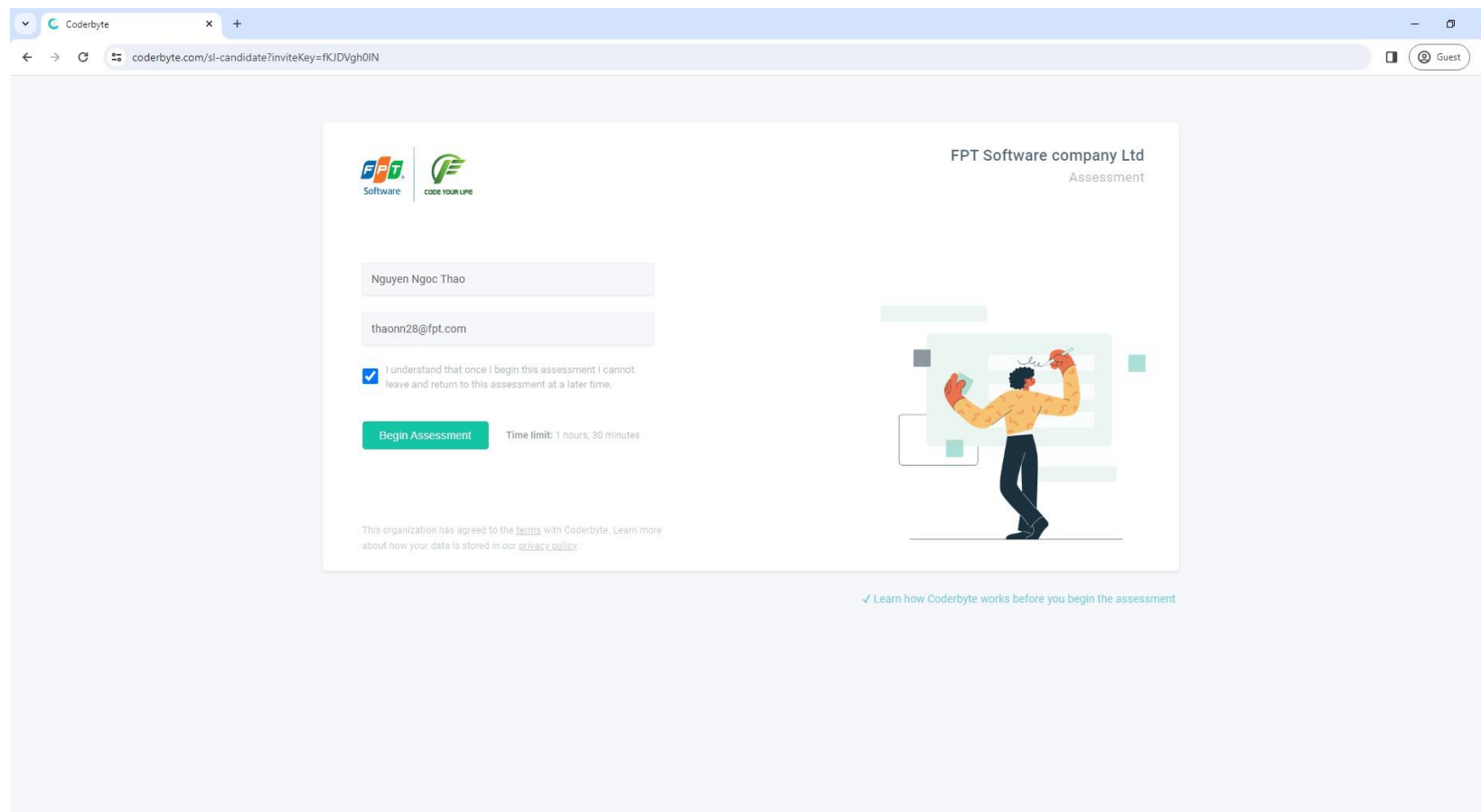
# After being invited

- After being invited, you will receive an email containing a link to the test.



# Signup and get started

- Enter your full name (**note: without accents**) and then click the **"Begin Assessment"** button.

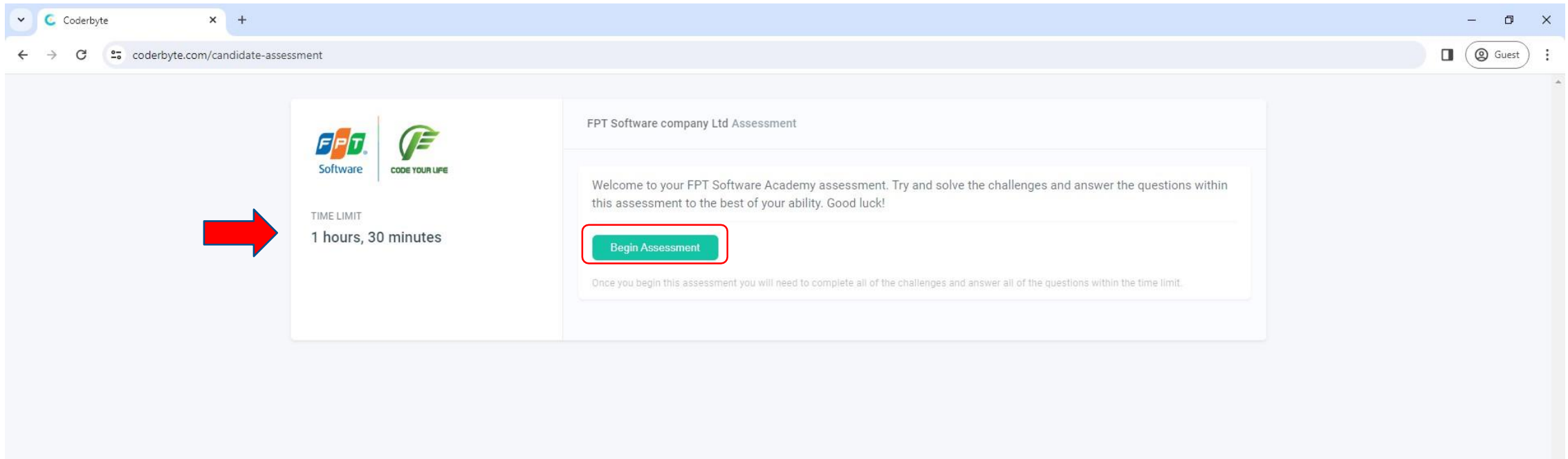


The screenshot shows a web browser window with the URL `coderbyte.com/sl-candidate?inviteKey=fKJDVgh0IN`. The page is titled "FPT Software company Ltd Assessment". It features the FPT Software and "CODE YOUR LIFE" logos. The form contains the following fields and elements:

- Full name field: "Nguyen Ngoc Thao"
- Email field: "thaonn28@fpt.com"
- Terms and conditions checkbox: ☒ "I understand that once I begin this assessment I cannot leave and return to this assessment at a later time."
- "Begin Assessment" button
- Time limit: "Time limit: 1 hours, 30 minutes"
- Illustration of a person interacting with a large screen.
- Footer text: "This organization has agreed to the [terms](#) with Coderbyte. Learn more about how your data is stored in our [privacy policy](#)."
- Link: "✓ Learn how Coderbyte works before you begin the assessment"

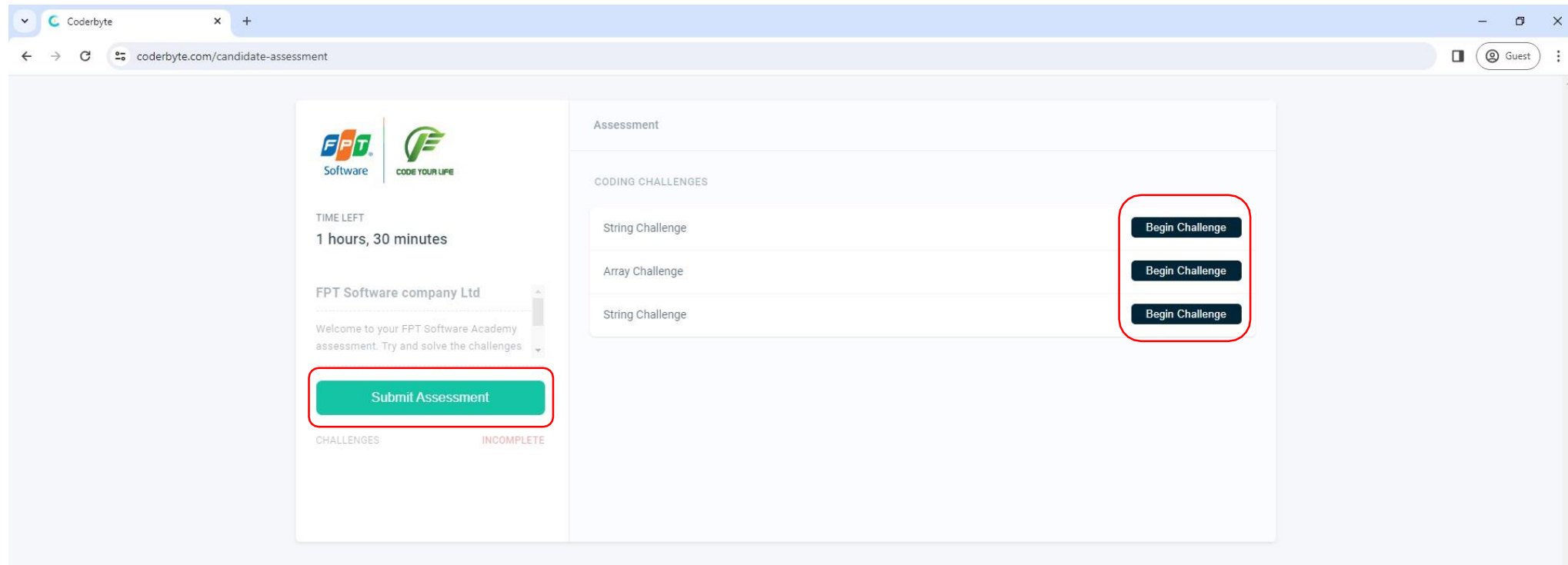
# Test duration information

- Review the test duration, and then press the "**Begin Assessment**" button.



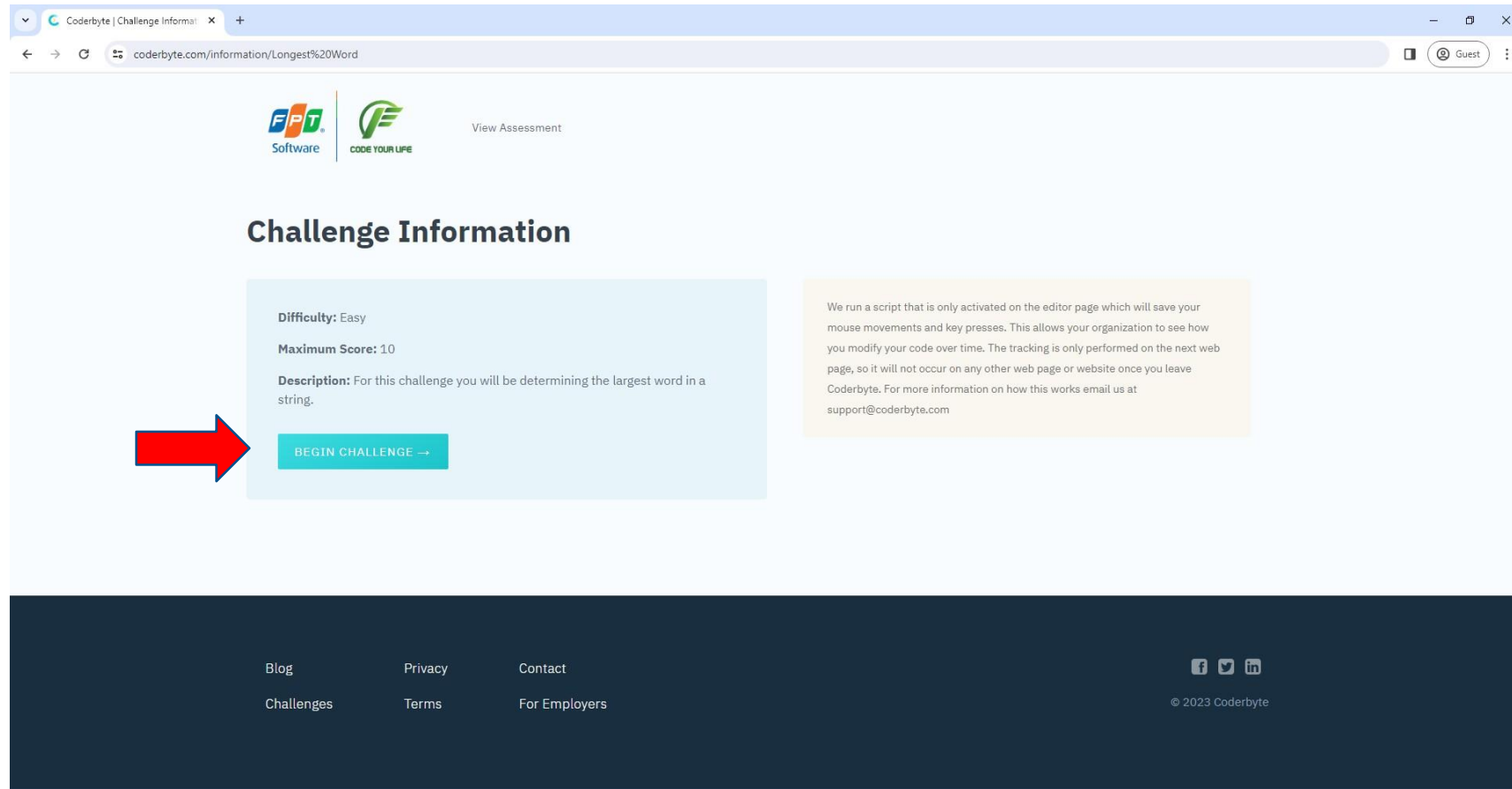
# Test main screen

- A test consists of one or more challenges. Click on the "**Begin Challenge**" button to start answering each challenge individually.
- Click on the "**Submit Assessment**" button to submit the entire Test.



# Challenge information

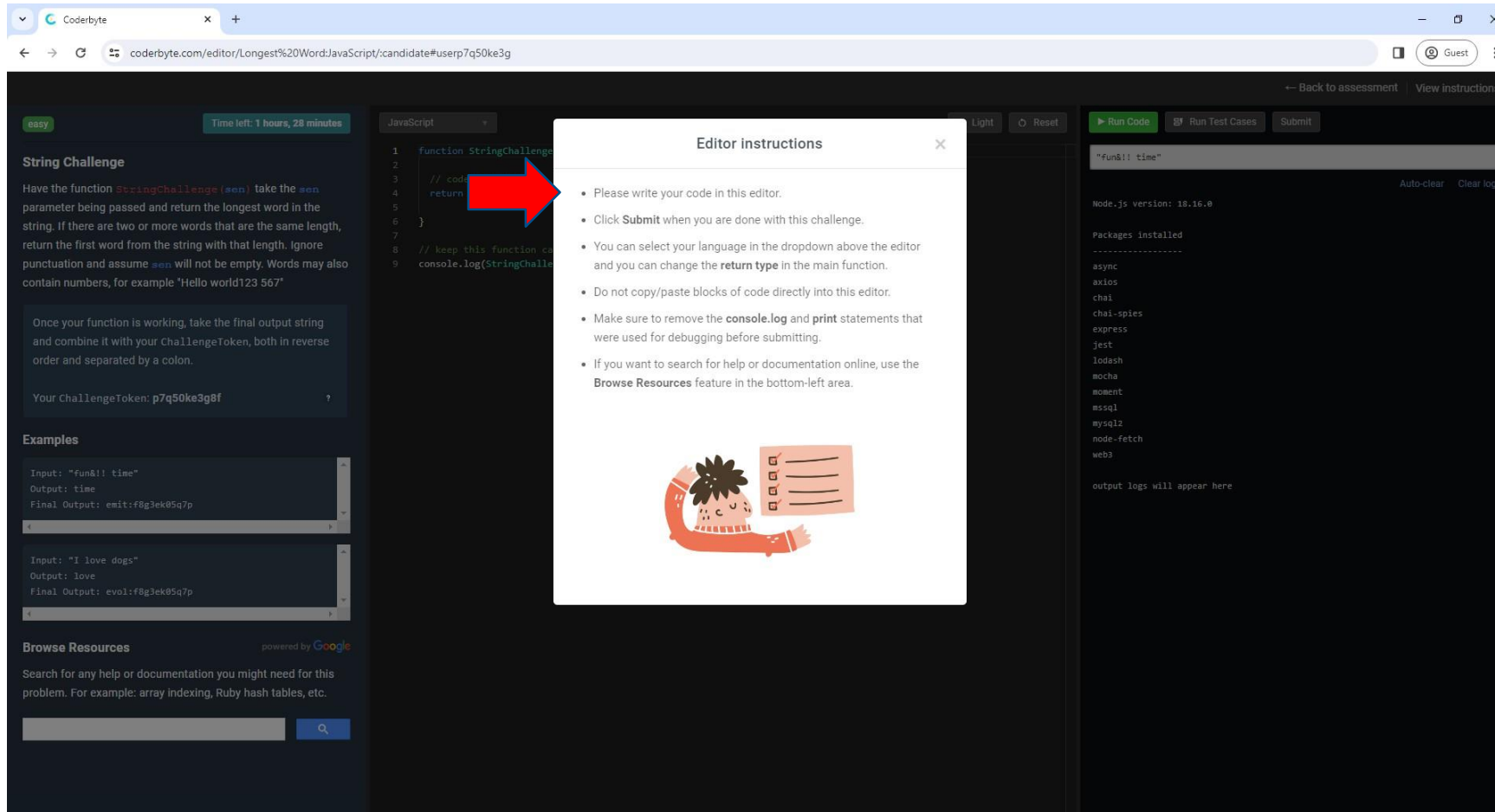
- Review the general information about the challenge then click the **"Begin Challenge"** button to start.





# Editor instructions

- Read the Editor instructions **carefully**.



The screenshot shows the Coderbyte editor interface. On the left, the 'String Challenge' is displayed with its description and examples. In the center, a code editor shows a JavaScript function. On the right, there are buttons for 'Run Code', 'Run Test Cases', and 'Submit'. A modal window titled 'Editor Instructions' is open in the center, listing guidelines for using the editor. A red arrow points from the 'return' statement in the code editor to the modal.

**String Challenge**

Have the function `StringChallenge(sen)` take the `sen` parameter being passed and return the longest word in the string. If there are two or more words that are the same length, return the first word from the string with that length. Ignore punctuation and assume `sen` will not be empty. Words may also contain numbers, for example "Hello world123 567"

Once your function is working, take the final output string and combine it with your ChallengeToken, both in reverse order and separated by a colon.

Your ChallengeToken: p7q50ke3g8f

**Examples**

Input: "fun!! time"  
Output: time  
Final Output: emit:f8g3ek85q7p

Input: "I love dogs"  
Output: love  
Final Output: evol:f8g3ek85q7p

**Editor Instructions**

- Please write your code in this editor.
- Click **Submit** when you are done with this challenge.
- You can select your language in the dropdown above the editor and you can change the **return type** in the main function.
- Do not copy/paste blocks of code directly into this editor.
- Make sure to remove the `console.log` and `print` statements that were used for debugging before submitting.
- If you want to search for help or documentation online, use the **Browse Resources** feature in the bottom-left area.

**Run Code** **Run Test Cases** **Submit**

Node.js version: 18.16.0

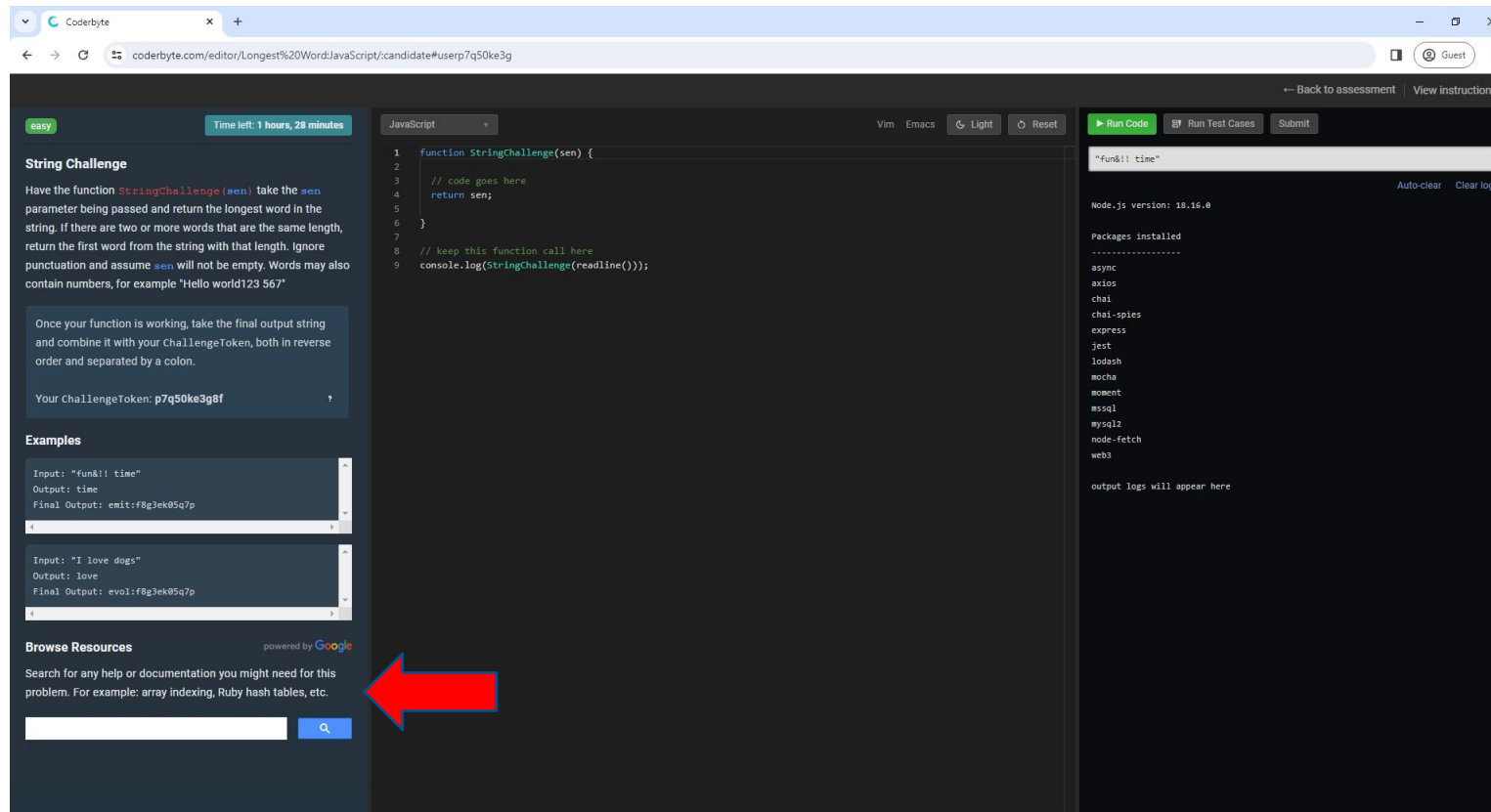
Packages installed

async  
axios  
chai  
chai-spies  
express  
jest  
lodash  
mocha  
moment  
mysql  
mysql2  
node-fetch  
web3

output logs will appear here

# During taking the challenge

- **Do not use** any research on the internet for knowledge or concepts if needed within the "Browse Resources" section.



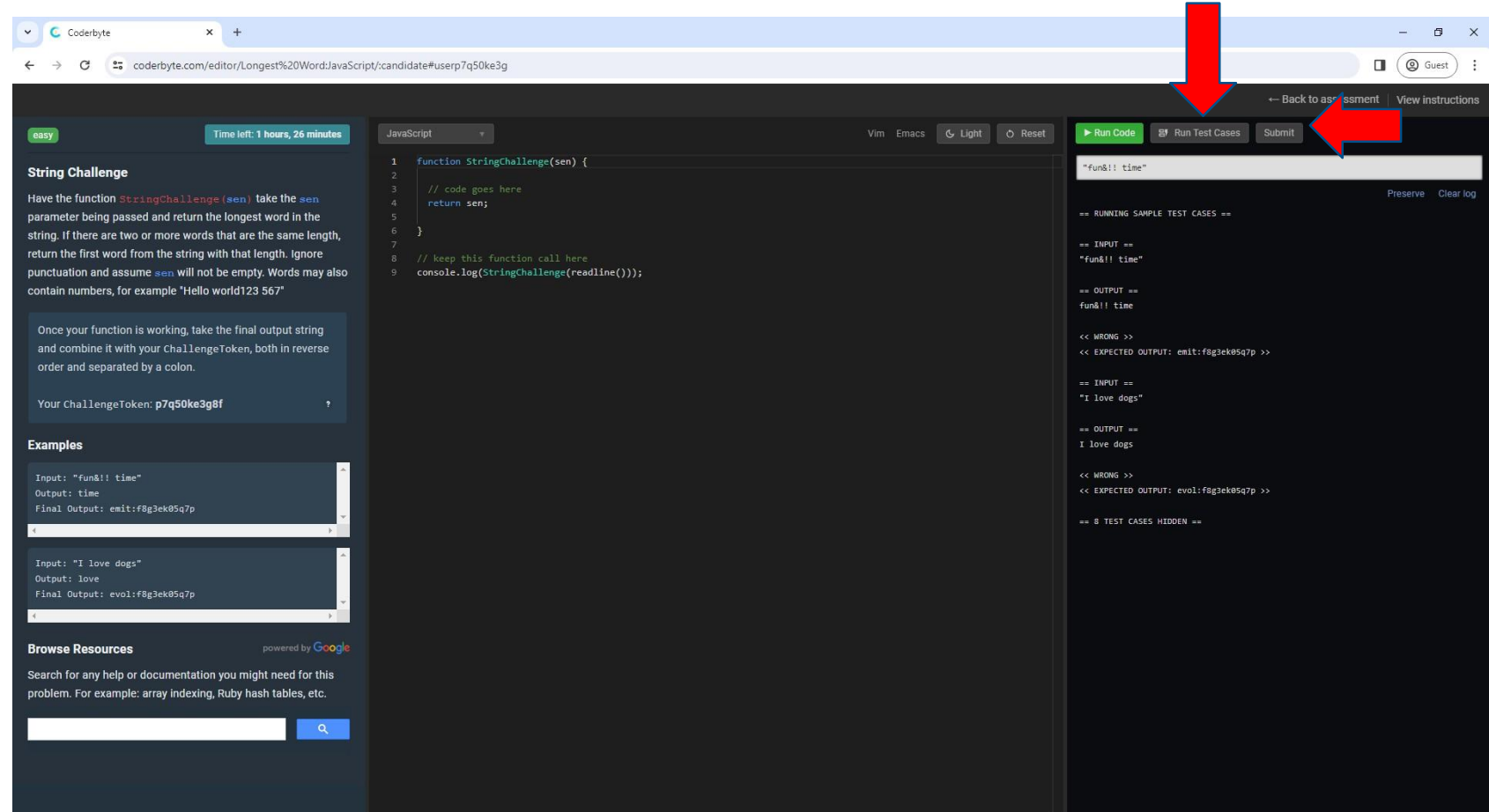
The screenshot shows the Coderbyte challenge interface for a JavaScript challenge titled "String Challenge". The interface is divided into several sections:

- Challenge Description:** "Have the function `StringChallenge(sen)` take the `sen` parameter being passed and return the longest word in the string. If there are two or more words that are the same length, return the first word from the string with that length. Ignore punctuation and assume `sen` will not be empty. Words may also contain numbers, for example 'Hello world123 567'".
- Instructions:** "Once your function is working, take the final output string and combine it with your ChallengeToken, both in reverse order and separated by a colon."
- ChallengeToken:** "Your ChallengeToken: p7q50ke3g0f".
- Examples:** Two examples are provided: "Input: 'fun!! time' Output: time Final Output: emit:f8g3ek05q7p" and "Input: 'I love dogs' Output: love Final Output: evol:f8g3ek05q7p".
- Browse Resources:** A section titled "Browse Resources" with a search bar and a magnifying glass icon. A red arrow points to this section.
- Code Editor:** A central area for writing JavaScript code. The code shown is: 

```
1 function StringChallenge(sen) {  
2  
3   // code goes here  
4   return sen;  
5  
6 }  
7  
8 // keep this function call here  
9 console.log(StringChallenge(readline()));
```
- Run Code:** A button labeled "Run Code" and a "Run Test Cases" button.
- Output:** A section showing the output of the code. It displays "fun!! time" and a list of installed packages: "async, axios, chai, chai-spies, express, jest, lodash, mocha, moment, mysql, mysql2, node-fetch, web3".

# Trying with provided test cases

- Run some provided test cases to validate your solution before submitting your work.
- Click on "**Submit**" to submit your solution.



The screenshot displays the Coderbyte JavaScript challenge interface. The challenge is titled "String Challenge" and has a difficulty level of "easy". The description states: "Have the function `StringChallenge(sen)` take the `sen` parameter being passed and return the longest word in the string. If there are two or more words that are the same length, return the first word from the string with that length. Ignore punctuation and assume `sen` will not be empty. Words may also contain numbers, for example 'Hello world123 567'".

Examples provided:

- Input: "fun!! time"  
Output: time  
Final Output: emit:f8g3ek05q7p
- Input: "I love dogs"  
Output: love  
Final Output: evol:f8g3ek05q7p

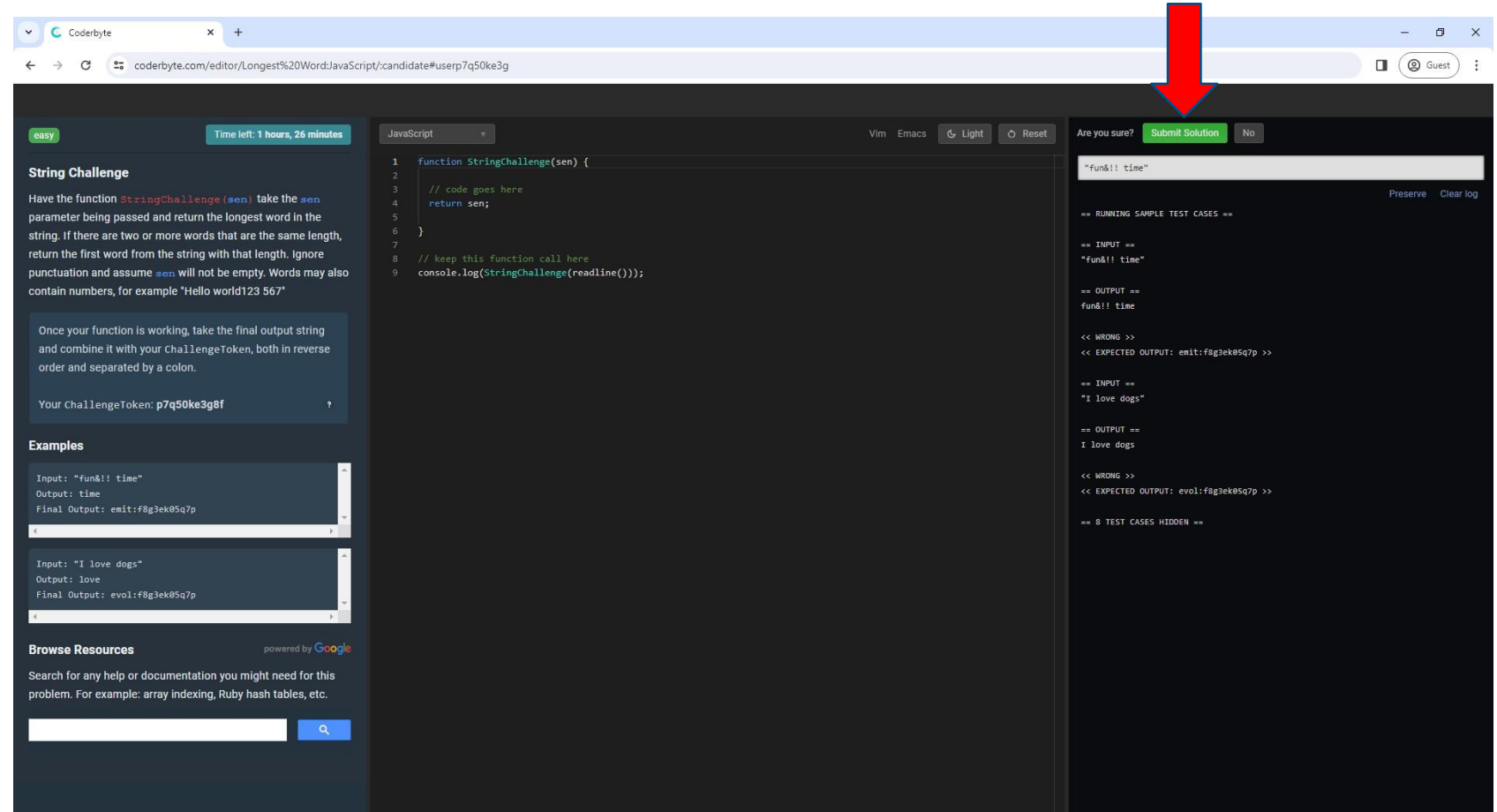
The code editor shows the following JavaScript code:

```
1 function StringChallenge(sen) {  
2   // code goes here  
3   return sen;  
4 }  
5  
6  
7  
8 // keep this function call here  
9 console.log(StringChallenge(readline()));
```

On the right side, there are buttons for "Run Code", "Run Test Cases", and "Submit". A red arrow points to the "Run Test Cases" button, and another red arrow points to the "Submit" button. The test results on the right show two failed test cases with "WRONG" status and "EXPECTED OUTPUT" mismatches.

# Submit your challenge solution

- **Double-check** your solution before confirming the submission.



The screenshot shows the Coderbyte website interface for a JavaScript challenge. The browser address bar shows the URL: `coderbyte.com/editor/Longest%20Word/JavaScript/candidate#userp7q50ke3g`. The challenge is titled "String Challenge" and is marked as "easy". The time left is 1 hour, 26 minutes. The challenge description states: "Have the function `StringChallenge(sen)` take the `sen` parameter being passed and return the longest word in the string. If there are two or more words that are the same length, return the first word from the string with that length. Ignore punctuation and assume `sen` will not be empty. Words may also contain numbers, for example 'Hello world123 567'". Below the description, there is a section for "Examples" with two test cases: "fun!! time" (Output: time) and "I love dogs" (Output: love). The "Submit Solution" button is highlighted with a red arrow. The right sidebar shows the test cases and the output of the submitted solution.

```
1 function StringChallenge(sen) {
2
3   // code goes here
4   return sen;
5
6 }
7
8 // keep this function call here
9 console.log(StringChallenge(readline()));
```

Are you sure? **Submit Solution** No

fun!! time

== RUNNING SAMPLE TEST CASES ==

== INPUT ==  
"fun!! time"

== OUTPUT ==  
fun!! time

<< WRONG >>  
<< EXPECTED OUTPUT: emit:f8g3ek05q7p >>

== INPUT ==  
"I love dogs"

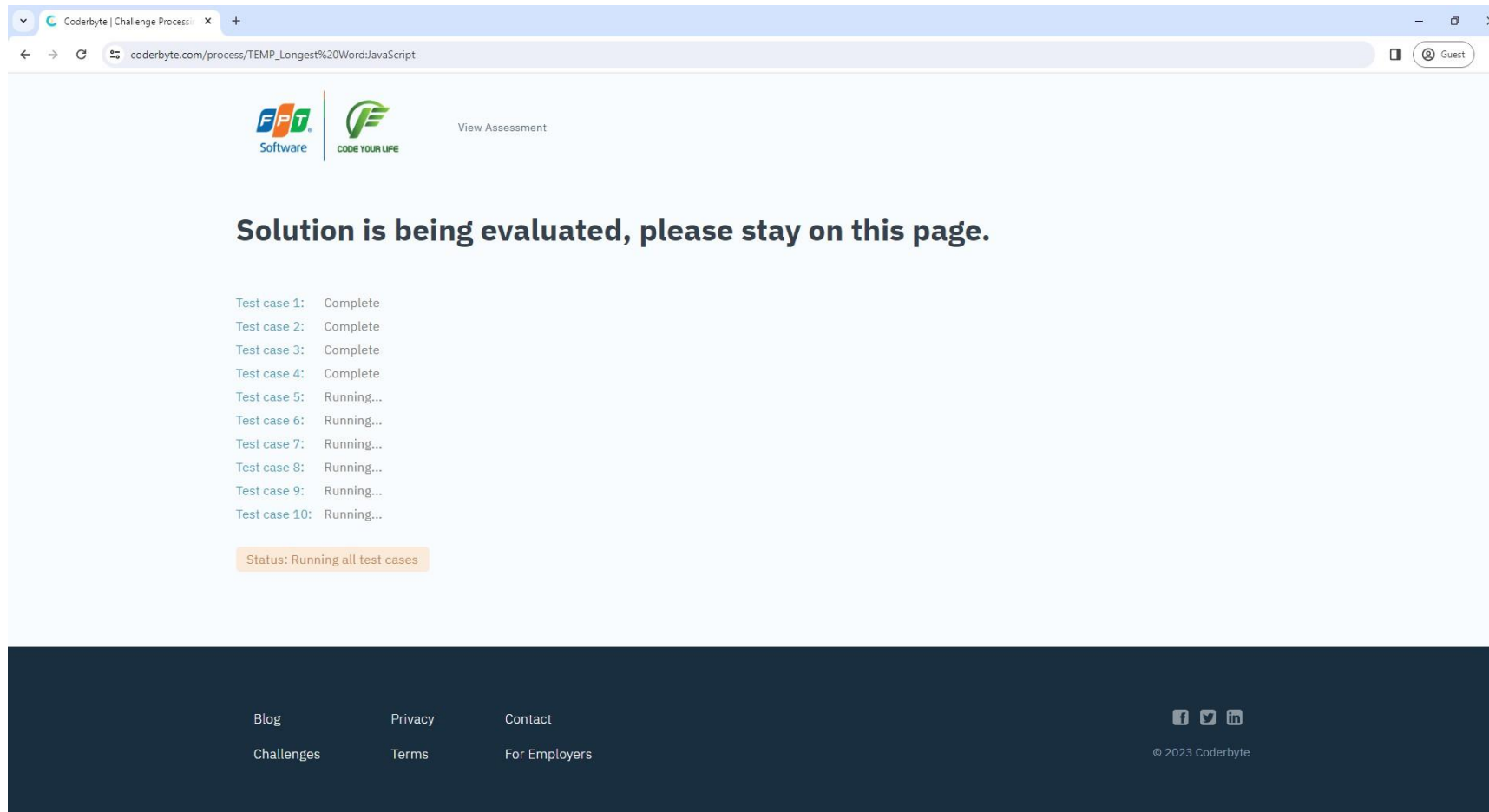
== OUTPUT ==  
I love dogs

<< WRONG >>  
<< EXPECTED OUTPUT: evol:f8g3ek05q7p >>

== 8 TEST CASES HIDDEN ==

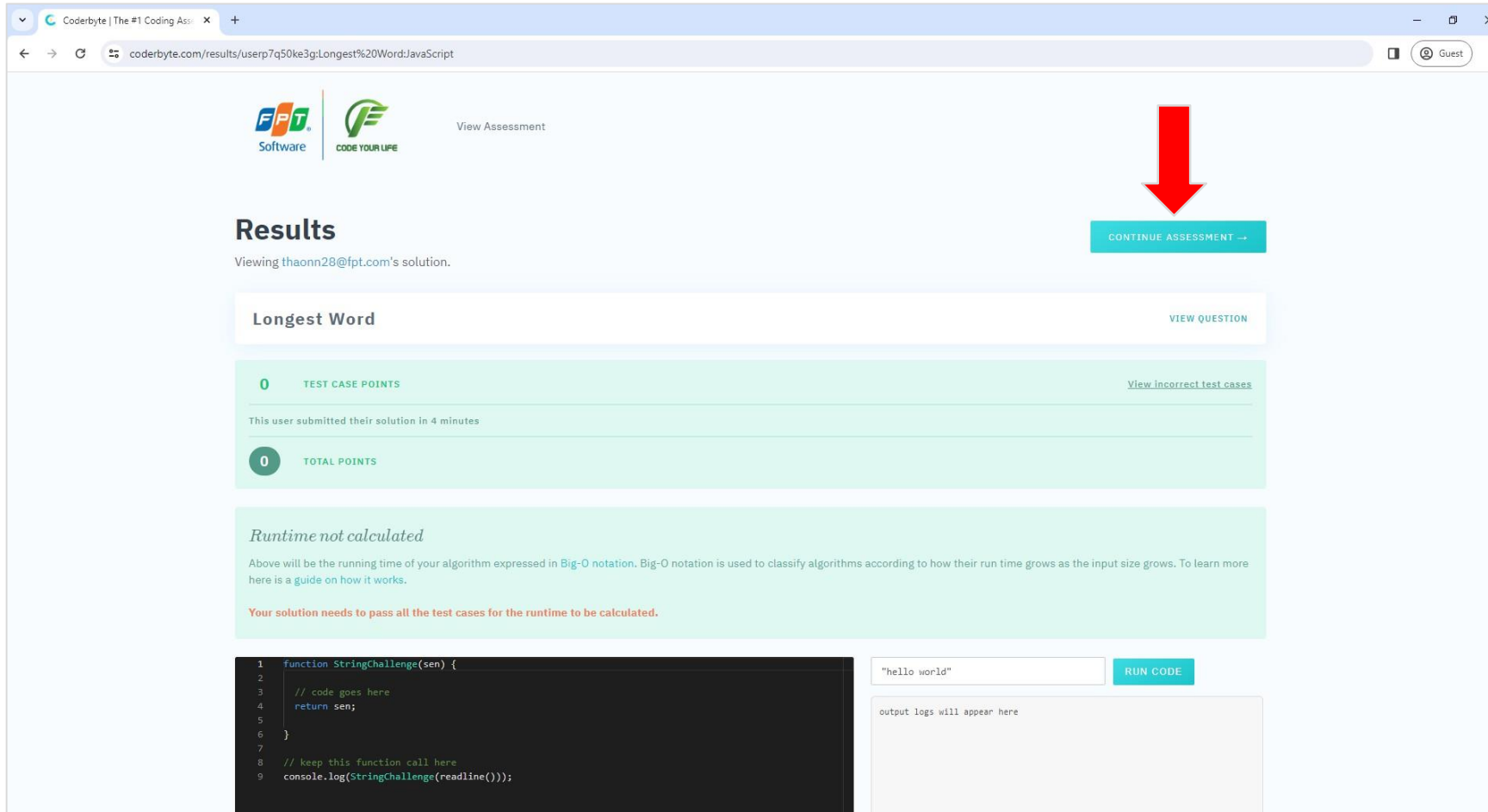
# Grading your challenge submission

- **Wait** for the system to grade your submission.



# Challenge result

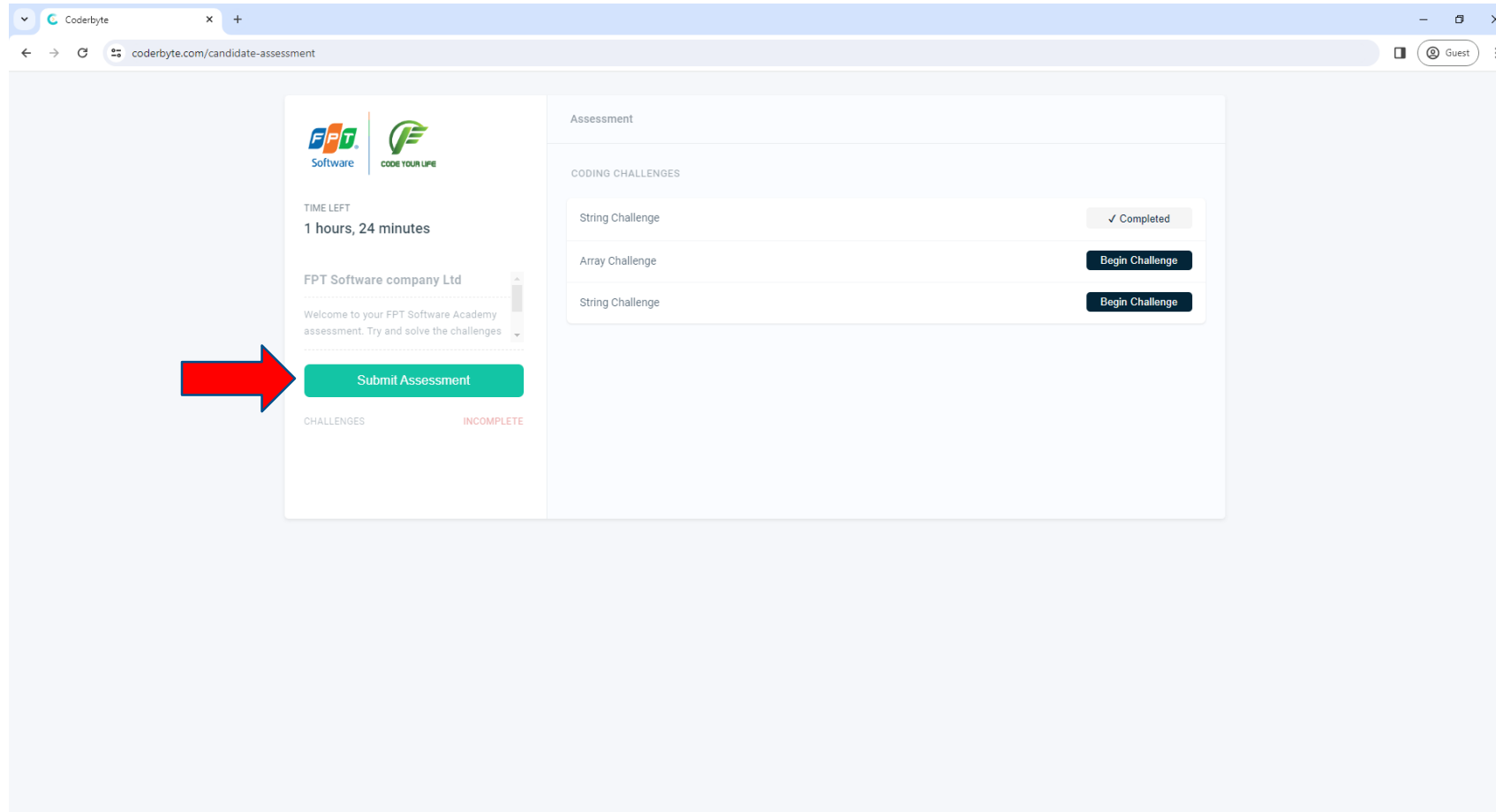
- Click on "**Continue Assessment**" to proceed with the remaining challenges.



The screenshot shows a web browser window with the URL `coderbyte.com/results/userp7q50ke3g:Longest%20Word:JavaScript`. The page header includes the FPT Software and CODE YOUR LIFE logos, and a "View Assessment" link. A large red arrow points to a teal button labeled "CONTINUE ASSESSMENT →". Below this, the "Results" section shows "Viewing thaonn28@fpt.com's solution." and a "VIEW QUESTION" link. The "Longest Word" challenge is displayed with a score of 0 TEST CASE POINTS and 0 TOTAL POINTS. A message states "Runtime not calculated" and "Your solution needs to pass all the test cases for the runtime to be calculated." At the bottom, there is a code editor with a JavaScript function `StringChallenge` and a "RUN CODE" button. The input field contains "hello world" and the output field is empty.

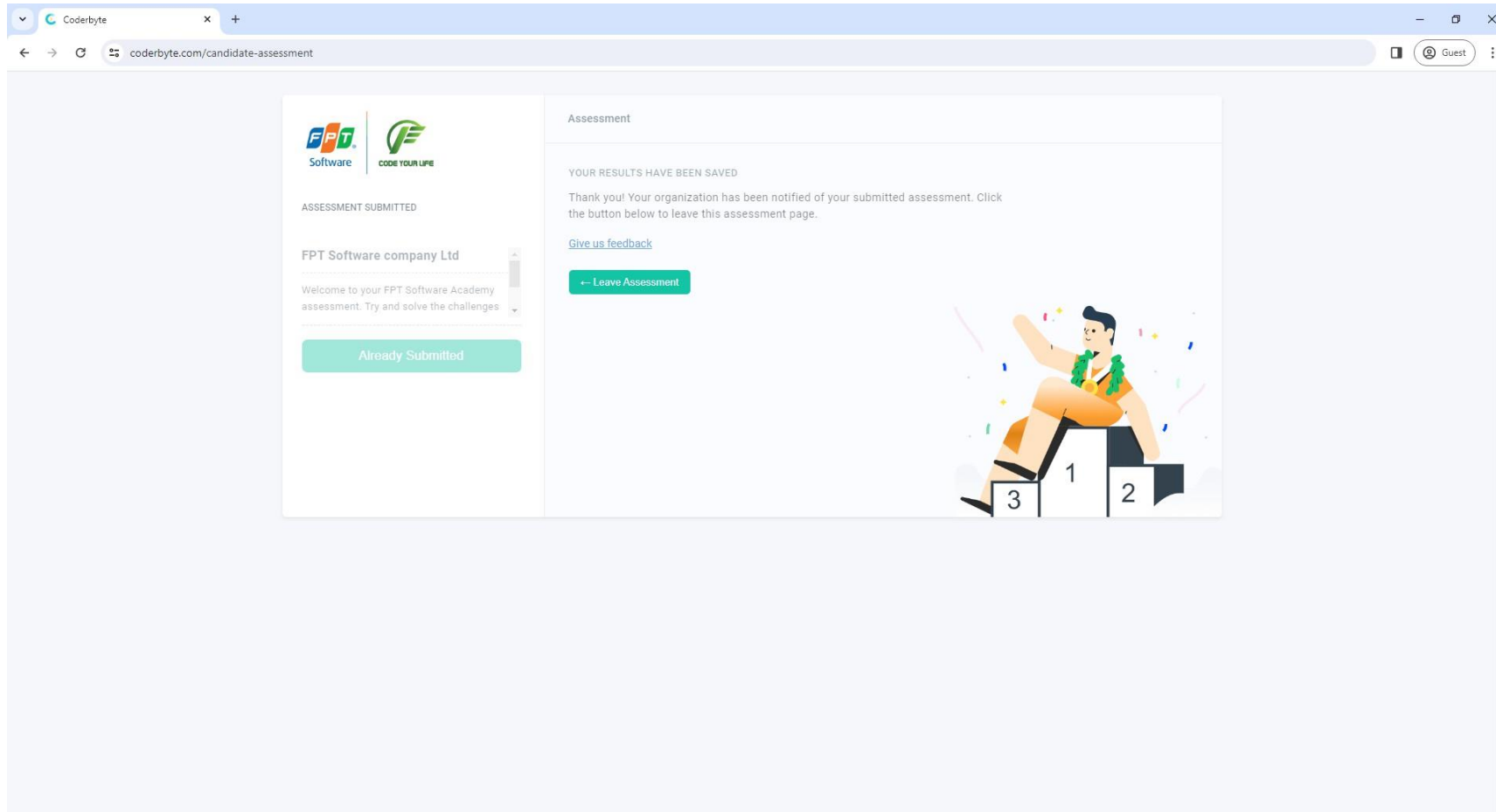
# Submit your entire test

- After completing all the challenges, click on "**Submit Assessment**" to submit the entire test.



# Submit your entire test

- After submitting your entire work, you can close the CoderByte tab.







# Try it out

- Explore various coding challenges at <https://coderbyte.com/challenges>

# THANK YOU!

