Clarusway

JS-CC-07 : Bracket Validator

Let's say:

- '(', '{', '[' are called "openers."
- ')', '}', ']' are called "closers."

Write an efficient function that tells us whether or not an input string's openers and closers are properly nested.

> **Examples:**
>
> - " ☐ ( ) }" should return true
> - "{ [( ]) }" should return false
> - "{ [}" should return false

- Simply making sure each opener has a corresponding closer is not enough—we must also confirm that they are correctly ordered.
- For example, "{ [( ]) }" should return false, even though each opener can be matched to a closer.

Learning Outcomes

At the end of the this coding challenge, students will be able to;

- Analyze a problem, identify and apply programming knowledge for appropriate solution.

- Demonstrate their knowledge of algorithmic design principles by using JavaScript and Python effectively.

Problem Statement

- Write a function that takes series of brackets and returns `true` or `false` depending on the rules above.

✍ Happy Coding ⏳