**Valid Parentheses with Multiple Types**

You are given a string s consisting of different types of parentheses: (), {}, and []. Your task is to determine whether the given string is valid.

A string is considered valid if:

1. Every opening bracket has a corresponding closing bracket of the same type.
2. The brackets are closed in the correct order. This means that a closing bracket must close the most recent unmatched opening bracket.

**Input:**

A string s consisting of characters (, ), {, }, [, and ].

**Output:**

* Return true if the string is valid.
* Return false if the string is invalid.

**Examples:**

* Example 1  
  Input: s = "()"

Output: true  
Explanation: The string contains only one pair of valid parentheses.

**Constraints:**

* 0 ≤ s.length ≤ 10^4
* The string s contains only the characters ()[]{}.

**Test Cases:**

1. Input: s = "()"

Output: true

1. Input: s = "([)]"

Output: false

1. Input: s = "[{()}]"

Output: true

1. Input: s = “”

Output: true

1. Input: s = "{[}"

Output: false

**Edge Cases:**

1. Empty string: If the input string is empty, the output should be true since there are no parentheses to match.
2. Odd length string: If the string has an odd number of characters, it cannot be valid and should return false.
3. Unmatched closing brackets: If the string starts with a closing bracket, it is invalid.