<https://leetcode.com/problems/valid-parentheses/>

Given a string s containing just the characters '(', ')', '{', '}', '[' and ']', determine if the input string is valid.

An input string is valid if:

1. Open brackets must be closed by the same type of brackets.
2. Open brackets must be closed in the correct order.
3. Every close bracket has a corresponding open bracket of the same type.

**Example 1:**

Input: s = "()"

Output: true

**Example 2:**

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

Output: true

**Example 3:**

Input: s = "(]"

Output: false

**Constraints:**

* 1 <= s.length <= 104
* s consists of parentheses only '()[]{}'.

**Attempt 1: 2022-11-26**

**Solution 1:  Stack (10min)**

class Solution {

public boolean isValid(String s) {

Stack<Character> stack = new Stack<Character>();

char[] chars = s.toCharArray();

for(char c : chars) {

if(c == '{' || c == '(' || c == '[') {

stack.push(c);

} else if(c == '}' && !stack.isEmpty() && stack.peek() == '{') {

stack.pop();

} else if(c == ')' && !stack.isEmpty() && stack.peek() == '(') {

stack.pop();

} else if(c == ']' && !stack.isEmpty() && stack.peek() == '[') {

stack.pop();

} else {

// Don't forget, test out by input as "]"

return false;

}

}

return stack.isEmpty();

}

}