**20.Valid Parentheses**

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

class Solution {

/\*TC : O(n)

SC : O(n) as we push all opening brackets onto the stack and in the worst case, we will end up pushing all the brackets onto the stack\*/

public boolean isValid(String s) {

Map<Character,Character> map = new HashMap<>();

map.put(')','(');

map.put(']','[');

map.put('}','{');

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

int index = 0;

for(char c: s.toCharArray())

{

if(map.containsKey(c))

{

if(!stack.empty())

{

char topElement = stack.pop();

if(topElement != map.get(c))

{

return false;

}

}

else

{

return false;

}

continue;

}

stack.push(c);

}

return stack.empty();

}

}