

유형	stack	소요 시간	20분				
문제	<p>[문제 설명]</p> <p>문자열 하나가 주어집니다. 이 문자열에 연속한 2개의 같은 문자가 존재하지 않도록 만들고 싶습니다. 연속한 2개의 같은 문자가 존재한다면 이 문자를 지우고 남은 문자열을 이어붙입니다. 이 과정을 연속한 2개의 같은 문자가 없을 때까지 반복하면 목표한 문자열을 얻게 됩니다.</p> <p>주어진 문자열에 대해 위와 같은 과정을 적용해서 나오는 문자열을 구해주세요.</p>						
	<p>[제한 사항]</p>						
	<p>[매개변수 & Return Type]</p>						
	<p>[입력 형식]</p> <p>매개변수 S에 길이 100,000 이하의 문자열이 주어집니다.</p> <p>S는 알파벳 소문자로만 이루어져있습니다.</p>						
	<p>[출력 형식]</p> <p>문제에서 설명된 과정을 거친 문자열을 반환하세요</p>						
매개변수 & Return Type	<p>[입출력 예시 #1]</p> <table><tr><th>S</th><th>출력</th></tr><tr><td>'aacddefg'</td><td>'cefg'</td></tr></table>			S	출력	'aacddefg'	'cefg'
	S	출력					
	'aacddefg'	'cefg'					
	<p>[입출력 예시 #2]</p> <table><tr><th>S</th><th>출력</th></tr><tr><td>'abbac'</td><td>'c'</td></tr></table>			S	출력	'abbac'	'c'
	S	출력					
'abbac'	'c'						
<p>s : string</p> <p>return : string</p>							
초기코드	<pre>/** * @param S {string} * @return {string} */ var solution = function (S) { var answer = ""; return answer; }</pre>						
모범 답안	<pre>/** * @param S {string} * @return {string} */ var solution = function (S) { const stack = []; for (const c of S) {</pre>						

	<pre> if (stack.length && stack[stack.length - 1] === c) { stack.pop(); } else { stack.push(c); } } var answer = ""; for (const c of stack) { answer = answer.concat(c); } return answer; }; var S1 = "aacddefg"; var S2 = "abbac"; console.log(solution(S2)); </pre>	
테스트 케이스	예시케이스(1개)	
	S	출력
	'aacddefg'	'cefg'
	정확성 테스트케이스(5개)	
	S	출력
	'abbac'	'c'
	S	출력
	너무 길기에 별도의 파일로 첨부합니다. 파일명 : problem2-input2.txt	'c'
	S	출력
	너무 길기에 별도의 파일로 첨부합니다. 입력 : problem2-input3.txt, 출력 : problem2-output3.txt	
	S	출력
	'accaaaaaabccbbbcabc'	'cabc'
	S	출력
	'aaaegdaeccdffahddgadbhghddgcbdbcdggcgae-fedeccfcfd'	'aegdaedahgadbhghgcbda bcdcgae-fedefcfd'