

프로그래머스 자바스크립트• 중복된 문자 제거

≡ 태그

문제 설명

문자열 `my_string`이 매개변수로 주어집니다. `my_string`에서 중복된 문자를 제거하고 하나의 문자만 남긴 문자열을 `return`하도록 `solution` 함수를 완성해주세요.

제한사항

$1 \leq \text{my_string} \leq 110$

`my_string`은 대문자, 소문자, 공백으로 구성되어 있습니다.

대문자와 소문자를 구분합니다.

공백(" ")도 하나의 문자로 구분합니다.

중복된 문자 중 가장 앞에 있는 문자를 남깁니다.

입출력 예 #1

"people"에서 중복된 문자 "p"와 "e"을 제거한 "peol"을 `return`합니다.

입출력 예 #2

"We are the world"에서 중복된 문자 "e", " ", "r" 들을 제거한 "We arthwold"을 `return`합니다.

```
function solution(my_string) {
  let result = "";
  for (const char of my_string) {
    if (!result.includes(char)) {
      result += char;
    }
  }
  return result;
}
```

이 문제를 해결하기 위해 새 문자열을 만들고 `my_string`의 문자를 반복할 수 있습니다. 각 반복에서 현재 문자가 이미 새 문자열에 있는지 확인할 수 있습니다. 새 문자열에 추가할 수 있습니다. 그렇지 않은 경우 건너뛸 수 있습니다.

```
function solution(my_string) {
  return [...new Set(my_string)].join('');
}
```

Set 개체는 JavaScript에서 고유한 값의 모음입니다. 값 집합을 저장하고 조작할 수 있으며 집합에 값이 있는지 추가, 삭제 및 확인하는 데 유용한 방법을 제공합니다.

Set 개체를 사용하여 집합에 고유한 문자를 저장한 다음 확산 연산자(...)를 사용하여 집합을 문자 배열로 변환하여 문자열에서 중복 문자를 제거할 수 있습니다. 이것은 문자열의 고유한 문자를 Set 개체에 저장하고 확산 연산자를 사용하여 Set 개체를 문자 배열로 변환한 다음 join() 메서드를 사용하여 배열의 요소를 문자열로 연결합니다. 결과 문자열에는 중복 문자가 포함되지 않습니다.

참고: Set 개체는 정렬되지 않은 컬렉션이므로 집합을 배열로 변환할 때 집합의 요소 순서가 유지되지 않을 수 있습니다. 요소의 순서를 유지하려면 이전 솔루션에 표시된 것처럼 배열과 includes() 메서드를 사용하여 고유한 문자만 저장할 수 있습니다.

```
const myString = "hello world";
const uniqueChars = new Set(myString);
const charArray = [...uniqueChars];
const result = charArray.join('');
console.log(result); // should print "helo wrd"
```

이 솔루션은 JavaScript의 문자열에서 중복 문자를 제거하는 간결하고 효율적인 방법입니다. Set 개체를 사용하여 문자열에 고유한 문자를 저장하고 확산 연산자(...)를 사용하여 Set 개체를 문자 배열로 변환합니다. 그런 다음 join() 메서드를 사용하여 배열의 요소를 문자열로 연결합니다.

```
import java.util.stream.Collectors;
class Solution {
    public String solution(String my_string) {
        return my_string.chars()
            .mapToObj(Character::toString)
            .distinct()
            .collect(Collectors.joining());
    }
}
```

```
import java.util.*;
class Solution {
    public String solution(String my_string) {
        String[] answer = my_string.split("");
        Set<String> set = new LinkedHashSet<String>(Arrays.asList(answer));

        return String.join("", set);
    }
}
```