

# 프로그래머스 자바스크립트• 369게임

≡ 태그

## 문제 설명

머쓱이는 친구들과 369게임을 하고 있습니다. 369게임은 1부터 숫자를 하나씩 대며 3, 6, 9가 들어가는 숫자는 숫자 대신 3, 6, 9의 개수만큼 박수를 치는 게임입니다. 머쓱이가 말해야하는 숫자 order가 매개변수로 주어질 때, 머쓱이가 쳐야할 박수 횟수를 return 하도록 solution 함수를 완성해보세요.

입출력 예 설명

입출력 예 #1

3은 3이 1개 있으므로 1을 출력합니다.

입출력 예 #2

29423은 3이 1개, 9가 1개 있으므로 2를 출력합니다.

제한사항

$1 \leq \text{order} \leq 1,000,000$

참고: 이 솔루션은 숫자를 문자열로 변환하고 해당 숫자를 반복합니다. 이렇게 하면 숫자가 여러 자리인 경우에도 각 자리를 개별적으로 처리할 수 있습니다. 또한 `filter()` 메서드를 사용하여 숫자가 숫자에 나타나는 횟수를 계산합니다. `split('')` 메서드는 문자열을 문자 배열로 분할하는 데 사용됩니다. 이 경우 `filter()` 메서드를 적용할 수 있도록 숫자 문자열을 문자 배열로 변환하는 데 사용됩니다.

`filter()` 메서드는 콜백 함수를 인수로 받아 콜백 함수에 의해 구현된 테스트를 통과한 요소로 새로운 배열을 반환하는 고차 함수입니다. 이 경우 콜백 함수는 각 문자가 3, 6, 9인지 확인하고 있으면 `true`, 아니면 `false`를 반환합니다. 그런 다음 `filter()` 메서드는 이 테스트를 통과하는 요소(숫자 문자열의 3, 6 또는 9)로 새 배열을 만듭니다. 그런 다음 이 배열의 길이 속성을 사용하여 숫자 문자열에서 3, 6 또는 9의 수를 계산합니다.

예를 들어 숫자 문자열이 "29423"인 경우 `split('')` 메서드는 이를 `['2', '9', '4', '2', '3']` 배열로 변환하고 필터는 `()` 메서드는 테스트를 통과한 요소로 새 배열 `['9', '3']`을 만듭니다. 이 배열의 길이 속성은 숫자 문자열에서 3과 9의 수인 2입니다.

이게 도움이 되길 바랍니다! 다른 질문이 있으면 알려주세요.

```
function solution(order) {
    let claps = 0;

    for (const digit of order.toString()) {
        if (digit === '3' || digit === '6' || digit === '9') {
            claps += digit.split('').filter(d => d === '3' || d === '6' || d === '9').length;
        }
    }
}
```

```

    return claps;
}

```

```

function solution(order) {
    var answer = [...order.toString().matchAll(/[3|6|9]/g)].length;
    return answer;
}

function solution(order) {
    return (''+order).split(/[369]/).length-1;
}

```

```

class Solution {
    public int solution(int order) {
        int answer = 0;
        int count = 0;
        while(order != 0)
        {
            if(order % 10 == 3 || order % 10 == 6 || order % 10 == 9)
            {
                count++;
            }
            order = order/10;
        }
        answer = count;
        return answer;
    }
}

```

모듈로 연산자(%)를 사용하여 숫자의 마지막 자리를 추출하고 나누기 연산자(/)를 사용하여 숫자에서 마지막 자리를 제거합니다. 이렇게 하면 숫자가 여러 자리인 경우에도 각 자리를 개별적으로 처리할 수 있습니다.

```

class Solution {
    public int solution(int order) {
        int answer = 0;

        String str = order+"";

        for(int i=0; i<str.length(); i++){
            char c = str.charAt(i);
            if(c=='3' || c=='6' || c=='9') answer++;
        }

        return answer;
    }
}

```

```
}  
}
```