

Problem List

DescriptionEditorSolutionsSubmissions

2777. Date Range Generator Premium

Solved

Medium

Given a start date `start`, an end date `end`, and a positive integer `step`, return a generator object that yields dates in the range from `start` to `end` inclusive.

The value of `step` indicates the number of days between consecutive yielded values.

All yielded dates must be in the string format `YYYY-MM-DD`.

**Example 1:**

**Input:** `start = "2023-04-01", end = "2023-04-04", step = 1`  
**Output:** `["2023-04-01","2023-04-02","2023-04-03","2023-04-04"]`  
**Explanation:**  
`const g = dateRangeGenerator(start, end, step);`  
`g.next().value // '2023-04-01'`  
`g.next().value // '2023-04-02'`  
`g.next().value // '2023-04-03'`  
`g.next().value // '2023-04-04'`

**Example 2:**

**Input:** `start = "2023-04-10", end = "2023-04-20", step = 3`  
**Output:** `["2023-04-10","2023-04-13","2023-04-16","2023-04-19"]`  
**Explanation:**  
`const g = dateRangeGenerator(start, end, step);`  
`g.next().value // '2023-04-10'`  
`g.next().value // '2023-04-13'`  
`g.next().value // '2023-04-16'`  
`g.next().value // '2023-04-19'`

**Example 3:**

**Input:** `start = "2023-04-10", end = "2023-04-10", step = 1`  
**Output:** `["2023-04-10"]`  
**Explanation:**  
`const g = dateRangeGenerator(start, end, step);`  
`g.next().value // '2023-04-10'`

**Constraints:**

- `new Date(start) <= new Date(end)`
- `start` and `end` dates are in the string format `YYYY-MM-DD`
- `0 <=` The difference in days between the start date and the end date `<= 1500`
- `1 <= step <= 1000`

Seen this question in a real interview before? 1/5

Yes No

Accepted 326 | Submissions 404 | Acceptance Rate 80.7%

Discussion (0)

Copyright © 2024 LeetCode All rights reserved

10 0 0

</> Code

TypeScript • Auto

```
1 function* dateRangeGenerator(start: string, end: string, step: number) : Generator<string> {
2
3   while (new Date(start).valueOf() <= new Date(end).valueOf()) {
4     yield start;
5     const nextDate = rangeGenerator(start, step);
6     start = nextDate;
7   }
8
9 };
10
11 function rangeGenerator(start: string, step: number): string {
12
13   return new Date(new Date(start).valueOf() + step * 86400000).toISOString().split("T")[0]
14
15 }
16
17 /**
18  * const g = dateRangeGenerator('2023-04-01', '2023-04-04', 1);
19  * g.next().value; // '2023-04-01'
20  * ...           ..
21  * ...           ..

Ln 7, Col 6


```

Testcase Test Result

Accepted Runtime: 79 ms

Case 1

Case 2

Case 3

Input

"2023-04-01"

"2023-04-04"

1

Output

["2023-04-01","2023-04-02","2023-04-03","2023-04-04"]

[ "2023-04-01", "2023-04-02", "2023-04-03", "2023-04-04" ]