

Problem 2758: Next Day

Problem Information

Difficulty: Easy

Acceptance Rate: 0.00%

Paid Only: No

Problem Description

Write code that enhances all date objects such that you can call the

`date.nextDay()`

method on any date object and it will return the next day in the format

YYYY-MM-DD

as a string.

Example 1:

Input:

```
date = "2014-06-20"
```

Output:

```
"2014-06-21"
```

Explanation:

```
const date = new Date("2014-06-20"); date.nextDay(); // "2014-06-21"
```

Example 2:

Input:

`date = "2017-10-31"`

Output:

`"2017-11-01"`

Explanation:

The day after 2017-10-31 is 2017-11-01.

Constraints:

`new Date(date)`

is a valid date object

Code Snippets

JavaScript:

```
/**
 * @return {string}
 */
Date.prototype.nextDay = function() {

}

/**
 * const date = new Date("2014-06-20");
 * date.nextDay(); // "2014-06-21"
 */
```

TypeScript:

```
interface Date {
  nextDay(): string;
}
```

```

Date.prototype.nextDay = function(): string {

}

/**
 * const date = new Date("2014-06-20");
 * date.nextDay(); // "2014-06-21"
 */

```

Solutions

JavaScript Solution:

```

/**
 * Problem: Next Day
 * Difficulty: Easy
 * Tags: string
 *
 * Approach: String manipulation with hash map or two pointers
 * Time Complexity: O(n) or O(n log n)
 * Space Complexity: O(1) to O(n) depending on approach
 */

/**
 * @return {string}
 */
Date.prototype.nextDay = function() {

}

/**
 * const date = new Date("2014-06-20");
 * date.nextDay(); // "2014-06-21"
 */

```

TypeScript Solution:

```

/**
 * Problem: Next Day
 * Difficulty: Easy

```

```
* Tags: string
*
* Approach: String manipulation with hash map or two pointers
* Time Complexity: O(n) or O(n log n)
* Space Complexity: O(1) to O(n) depending on approach
*/

interface Date {
nextDay(): string;
}

Date.prototype.nextDay = function(): string {

}

/**
 * const date = new Date("2014-06-20");
 * date.nextDay(); // "2014-06-21"
 */
```