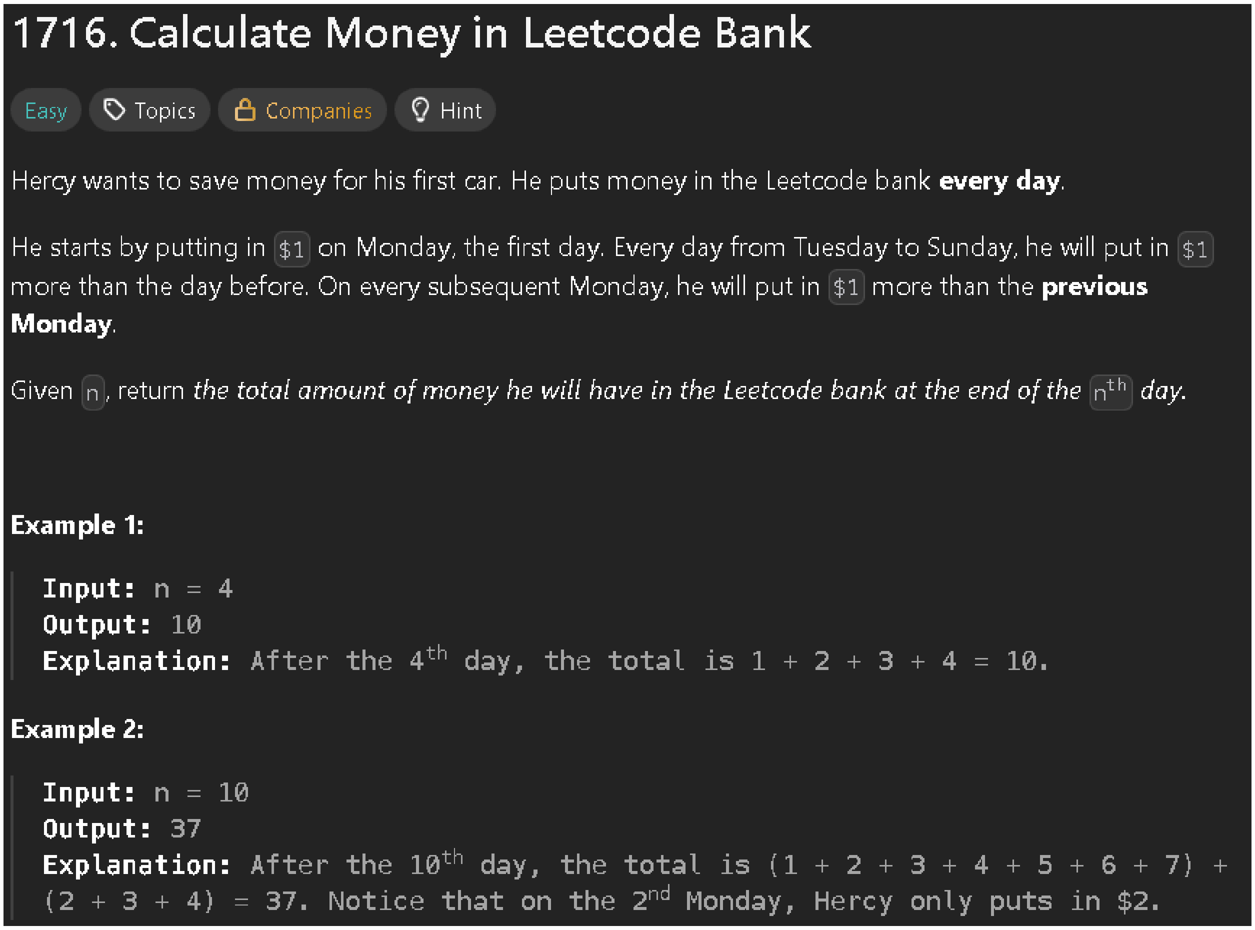


1716. Calculate Money in Leetcode Bank

Hercy wants to save money for his first car. He puts money in the Leetcode bank **every day**.

He starts by putting in `$1` on Monday, the first day. Every day from Tuesday to Sunday, he will put in `$1` more than the day before. On every subsequent Monday, he will put in `$1` more than the **previous Monday**.

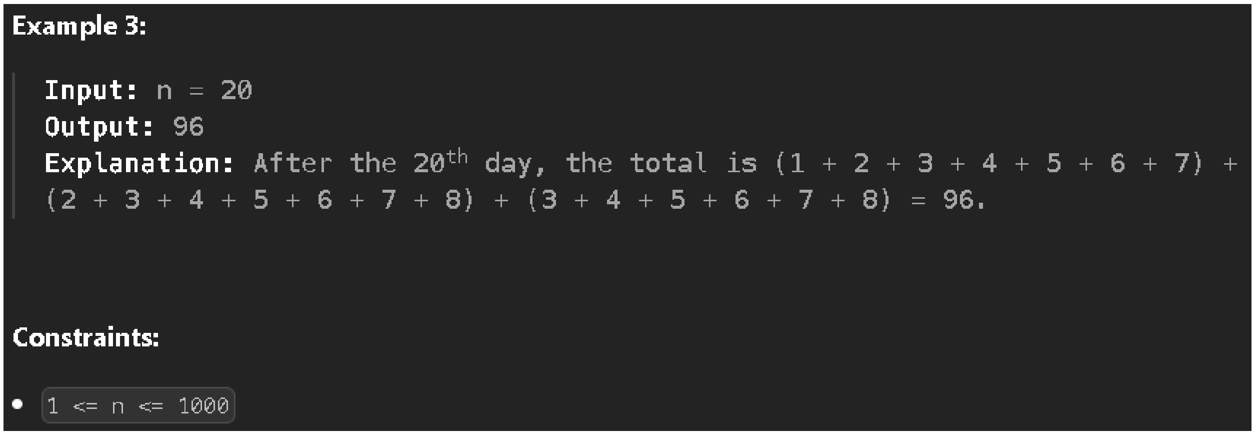
Given `n`, return *the total amount of money he will have in the Leetcode bank at the end of the* `nth` *day.*

Example 1:

```
Input: n = 4
Output: 10
Explanation: After the 4th day, the total is 1 + 2 + 3 + 4 = 10.
```

Example 2:

```
Input: n = 10
Output: 37
Explanation: After the 10th day, the total is (1 + 2 + 3 + 4 + 5 + 6 + 7) +
(2 + 3 + 4) = 37. Notice that on the 2nd Monday, Hercy only puts in $2.
```



Example 3:

```
Input: n = 20
Output: 96
Explanation: After the 20th day, the total is (1 + 2 + 3 + 4 + 5 + 6 + 7) +
(2 + 3 + 4 + 5 + 6 + 7 + 8) + (3 + 4 + 5 + 6 + 7 + 8) = 96.
```

Constraints:

- `1 <= n <= 1000`

Python:

```
class Solution:
    def totalMoney(self, n: int) -> int:
        ans = 0
        monday = 1

        while n > 0:
            for day in range(min(n, 7)):
                ans += monday + day

            n -= 7
            monday += 1

        return ans
```

JavaScript:

```
function totalMoney(n) {
    let ans = 0;        // Variable to store the total money earned
    let monday = 1;     // Represents the amount earned on Monday

    while (n > 0) {     // Continue the loop until 'n' days are exhausted
        for (let day = 0; day < Math.min(n, 7); day++) {  // Iterate for each day of the week or until 'n' days are left
            ans += monday + day;  // Increment 'ans' by the amount earned on the current day (monday + day)
        }

        n -= 7;         // Deduct 7 days (a week) from the total number of days left
        monday++;       // Increment the amount earned on Monday for the next week
    }

    return ans;        // Return the total amount earned over 'n' days
}
```

Java:

```
class Solution {
    public int totalMoney(int n) {
        int ans = 0;
        int monday = 1;

        while (n > 0) {
            for (int day = 0; day < Math.min(n, 7); day++) {
                ans += monday + day;
            }

            n -= 7;
            monday++;
        }

        return ans;
    }
}
```