Write a function

Problem | Submissions | Leaderboard | Discussions | Editorial

An extra day is added to the calendar almost every four years as February 29, and the day is called a leap day. It corrects the calendar for the fact that our planet takes approximately 365.25 days to orbit the sun. A leap year contains a leap day.

In the Gregorian calendar, three conditions are used to identify leap years:

- The year can be evenly divided by 4, is a leap year, unless:
  - The year can be evenly divided by 100, it is NOT a leap year, unless:
    - The year is also evenly divisible by 400. Then it is a leap year.

This means that in the Gregorian calendar, the years 2000 and 2400 are leap years, while 1800, 1900, 2100, 2200, 2300 and 2500 are NOT leap years. Source

Task

Given a year, determine whether it is a leap year. If it is a leap year, return the Boolean `True`, otherwise return `False`.

Note that the code stub provided reads from STDIN and passes arguments to the `is_leap` function. It is only necessary to complete the `is_leap` function.

Input Format

Read $year$, the year to test.

Constraints

$1900 \leq year \leq 10^5$

Output Format

The function must return a Boolean value (True/False). Output is handled by the provided code stub.

Sample Input 0

```
1990
```

Sample Output 0

```
False
```

Explanation 0

1990 is not a multiple of 4 hence it's not a leap year.

Author shashank21j
Difficulty Medium
Max Score 10
Submitted By 1070250

NEED HELP?

View discussions
View editorial
View top submissions

RATE THIS CHALLENGE

MORE DETAILS

Download problem statement
Download sample test cases
Suggest Edits

Change Theme Language Python 3

```python
def is_leap(year):
    leap = False
    if year%4 == 0:
        if year%100==0 and year%400==0:
            leap = True
        elif year%100==0:
            leap = False
        else:
            leap = True

    # Write your logic here

    return leap

year = int(input())
```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code

Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

Sample Test case 0

Input (stdin) Download

```
1990
```

Your Output (stdout)

```
False
```

Expected Output Download

```
False
```