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 , the year to test.

Constraints

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.

Change Theme Language Python 3

```
def is_leap(year):
    leap = False

    # Write your logic here
    if year%400==0:
        leap=True
    elif year%100==0:
        leap=False
    elif year%4==0:
        leap=True
    return leap
year = int(input())
```

int([x]) -> integer int(x, base=10) -> integ
Convert a number or string to an integer, or
If no arguments are given. If x is a number, re
x.__int__(). For floating point numbers, thi
towards zero.
If x is not a number or if base is given, th
string, bytes, or bytearray instance repre
integer literal in the given base. The liter
preceded by '+' or '-' and be surrounded by
The base defaults to 10. Valid bases are 0 a
0 means to interpret the base from the strin
integer literal.

Line: 10 Col: 18

Upload Code as File Run Code Submit Code

Test against custom input

You have earned 10.00 points!

You are now 30 points away from the 2nd star for your python badge.

14% 40/70

Python

Congratulations

You solved this challenge. Would you like to challenge your friends?

Next Challenge

Test case 0

Test case 1

Test case 2

Test case 3

Test case 4

Test case 5

Compiler Message

Success

Hidden Test Case

Unlock this testcase for 5 hackos.

Unlock