| **Prepare** > Python > Introduction > Write a function

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

```python
def is_leap(year):
    leap = False
    # Check if the year is divisible by 400 (leap year)
    if (year % 400 == 0):
        leap = True
    # Check if the year is divisible by 100 (not a leap
    elif (year % 100 == 0):
        leap = False
    # Check if the year is divisible by 4 (leap year)
    elif (year % 4 == 0):
        leap = True
    return leap

year = int(input())
print(is_leap(year))
```

Line: 13 Col: 1

⬆ Upload Code as File   Run Code   **Submit Code**

☐ Test against custom input

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**

1 | **1990**

Your Output (stdout)

1 | **False**

Expected Output   **Download**

1 | **False**