Java Date and Time ★

50 more points to get your next star!

Rank: **421157** | Points: **100/150**

Your Java Date and Time submission got 15.00 points. Share Tweet

You are now 50 points away from the 4th star for your java badge.

Try the next challenge | **Try a Random Challenge**

Problem Submissions Leaderboard Editorial 🔒

The Calendar class is an abstract class that provides methods for converting between a specific instant in time and a set of calendar fields such as YEAR, MONTH, DAY_OF_MONTH, HOUR, and so on, and for manipulating the calendar fields, such as getting the date of the next week.

You are given a date. You just need to write the method, $getDay$, which returns the day on that date. To simplify your task, we have provided a portion of the code in the editor.

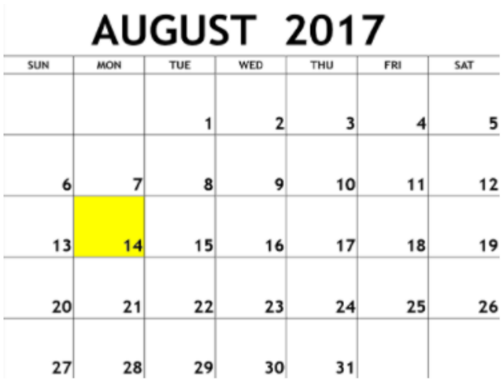
Example

$month = 8$

$day = 14$

$year = 2017$

The method should return $MONDAY$ as the day on that date.



AUGUST 2017

SUN	MON	TUE	WED	THU	FRI	SAT
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Function Description

Complete the findDay function in the editor below.

findDay has the following parameters:

- int: month
- int: day
- int: year

Returns

- string: the day of the week in capital letters

Input Format

A single line of input containing the space separated month, day and year, respectively, in $MM\ DD\ YYYY$ format.

Constraints

- $2000 < year < 3000$

Sample Input

```
08 05 2015
```

Sample Output

```
WEDNESDAY
```

Explanation

The day on August 5^{th} 2015 was `WEDNESDAY`.

Change Theme Language Java 7

```
import java.io.*;
import java.math.*;
import java.security.*;
import java.text.*;
import java.util.*;
import java.util.concurrent.*;
import java.util.regex.*;

class Result {


    public static String findDay(int month, int day, int year) {
        Calendar cal = Calendar.getInstance();
        cal.set(Calendar.MONTH,month-1);
        cal.set(Calendar.DAY_OF_MONTH,day);
        cal.set(Calendar.YEAR,year);
String dayOfWeek = cal.getDisplayName(Calendar.DAY_OF_WEEK, Calendar.LONG, Locale.US).toUpperCase();
        return dayOfWeek;
    }
}

public class Solution {
    public static void main(String[] args) throws IOException {
        BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(System.in));
        BufferedWriter bufferedWriter = new BufferedWriter(new FileWriter(System.getenv("OUTPUT_PATH")));

        String[] firstMultipleInput = bufferedReader.readLine().replaceAll("\\s+$", "").split(" ");

        int month = Integer.parseInt(firstMultipleInput[0]);

        int day = Integer.parseInt(firstMultipleInput[1]);

        int year = Integer.parseInt(firstMultipleInput[2]);

        String res = Result.findDay(month, day, year);
```

Line: 8 Col: 1

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

You have earned 15.00 points!

You are now 50 points away from the 4th star for your java badge.

29% 100/150

Java ★★★

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
```

Input (stdin) Download

```
08 05 2015
```

Expected Output Download

```
WEDNESDAY
```

Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy