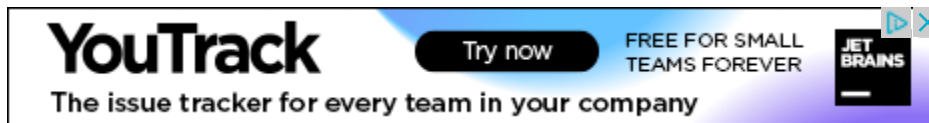


Java Leap Year Program

How to write Java Leap Year Program using If Statement, Nested If Statement, and Else If Statement in Java Programming with example?. Before we get into Java Leap Year Program, Let us see the logic and definition behind the Leap Year



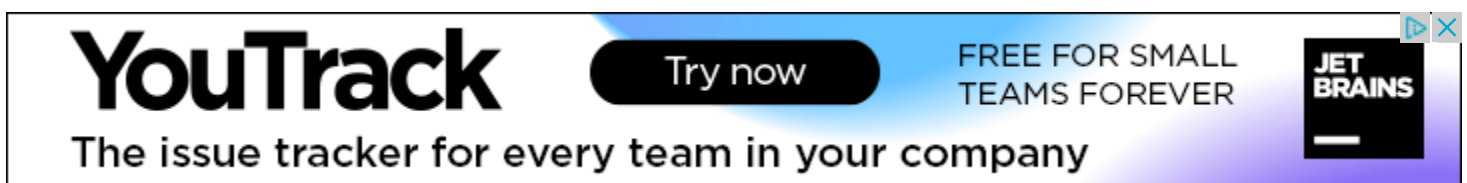
Java Leap year


The normal year contains 365 days, but the leap year contains 366 days. This extra day added to the February month, that is why we get February 29. As per Mathematics, except century years, Years perfectly divisible by four are called Leap years. Century year's means they end with 00 such as 1200, 1300, 2400, 2500 etc (Obviously they are divisible by 100). For these century years, we have to calculate further to check the Leap year.

- If the century year is divisible by 400, then that year is a Leap year
- If the century year is not divisible by 400, then that year is not a Leap year.

Java Leap Year Program using If Statement

This **Java program** for leap year allows the user to enter any year. Then this Java program will check whether the user entered year is Leap year or not using the **Java If Else statement**





Download Now | Open PDF

Download Newest Version Now | Fall in love with PDF again

Open PDF

[Open >](#)



Every Device

WATCH DEMO NOW

```
// Java Leap Year Program using If Statement

package DatePrograms;

import java.util.Scanner;


public class LeapYearUsingIf {
    private static Scanner sc;

    public static void main(String[] args) {
        int year;
        sc = new Scanner(System.in);

        System.out.println("\n Please Enter any year you wish: ");
        year = sc.nextInt();

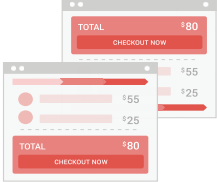

        if (( year%400 == 0)|| (( year%4 == 0 ) &&( year%100 != 0))) {
            System.out.format("\n %d is a Leap Year. \n", year);
        }
        else {
            System.out.format("\n %d is NOT a Leap Year. \n", year);
        }
    }
}
```

OUTPUT

 Optimize

Google Optimize makes A/B testing as easy as 1-2-3.

[Test for free](#)



Let us check for a normal year (Not Leap Year)

ANALYSIS

In this **Java** leap year program, Since we have to check multiple conditions within one If Statement, we used **Logical AND** and Logical OR operators. Let us divide the condition to understand it better

1: (year % 400 == 0)||

2: (year % 4 == 0) &&

3: (year % 100 != 0))



The first condition (year%400 == 0) will check whether the (year%400) reminder is precisely equal to 0 or not. As per the algorithm, any number that is divisible by 400 is a Leap year.

OR

[X](#)

The second condition holds two statements, so they both have to be TRUE.

then the given number is not a century number.

- Any number that is divisible by four but not by 100, then that number is Leap Year.

TIP: Please refer to [Java Logical Operators](#) to understand the Logical And and Logical Or.

Java Leap Year Program using Else If Statement

This Java leap year [program](#) allows the user to enter any year. Then the Java program will check whether the user entered year is Leap year or not using the [Java Else If statement](#).

```
public class LeapyearUsingElself {
    private static Scanner sc;

    public static void main(String[] args) {
        int year;
        sc = new Scanner(System.in);

        System.out.println("\n Please Enter any year you wish: ");
        year = sc.nextInt();

        if ( year % 400 == 0) {
            System.out.format("\n %d is a Leap Year. \n", year);
        }
        else if (year%100 == 0) {
            System.out.format("\n %d is NOT a Leap Year. \n", year);
        }
        else if(year%4 == 0) {
            System.out.format("\n %d is a Leap Year. \n", year);
        }
        else {
            System.out.format("\n %d is NOT a Leap Year. \n", year);
        }
    }
}
```

OUTPUT

ANALYSIS

In this Java leap year program, the following statements ask the user to enter any year to check whether that year is Leap year or Not. Next, we are going to assign the user entered value to year variable

```
System.out.println("\n Please Enter any year you wish: ");  
year = sc.nextInt();
```

As per the algorithm, any number that is not divisible by 400 but by 100 is Not a Leap year (Century Year). We checked (year % 400) in the First If statement. Because it failed, so it came to the second condition. If both the first and second condition Fails, then it will go to the third condition.

- The third condition will check whether year mod 4 is equal to 0. If this condition is True, then given year is Leap year because We already checked for the century years in the previous condition. If all the statements Fail, then it will go the Else statement at the end.
- If all the above statements fail, then it is not a Leap year in Java.

Java Leap Year Program using Nested If Statement

This leap year program in Java helps the user to enter any year. Then the Java program will check whether the user entered year is Leap year or not using the [Nested If in Java](#)


```
private static Scanner sc;

public static void main(String[] args) {
    int year;
    sc = new Scanner(System.in);

    System.out.println("\n Please Enter any year you wish: ");
    year = sc.nextInt();

    if ( year % 4 == 0) {
        if (year%100 == 0) {
            if(year%400 == 0) {
                System.out.format("\n %d is a Leap Year. \n",
year);
            }
            else {
                System.out.format("\n %d is NOT a Leap Year. \n",
year);
            }
        }
        else {
            System.out.format("\n %d is a Leap Year. \n", year);
        }
    }
    else {
        System.out.format("\n %d is NOT a Leap Year. \n", year);
    }
}
```

OUTPUT

ANALYSIS

In this Java leap year program, the User will enter any year to check whether that year is Leap year or Not. The first If condition will check whether the remainder of the (year%4) is precisely equal to 0 or not.

- If the condition is False, then the given number is not the Leap year.
- If the condition is True, then we have check further for the century year. So the Javac

compiler will go to the Nested If condition.

The second If condition will check (year%100) reminder is exactly equal to 0 or Not.

In this condition, the Javac will check whether the remainder of the (year%400) is exactly equal to 0 or not.

- If the condition is False, then the given number is not the Leap year.
- If the condition is True, then the given number is Leap Year in java

Java Leap Year Program using Oops

This **program** for java leap year allows entering any positive integer (year). Then the Java program checks whether the given year is a leap year or not. In this example, we are dividing the code using the Object-Oriented Programming.

To do this, first, we will create a class that holds a method to reverse an integer recursively.

```
package DatePrograms;

public class LeapYear {
    public int CheckLeapYear(int year) {
        if (( year%400 == 0)|| (( year%4 == 0 ) && ( year%100 != 0))) {
            return year;
        }
        else {
            return 0;
        }
    }
}
```

Within the Main program of the leap year program in java, we will create an instance of the above-specified class and call the methods.

```
sc = new Scanner(System.in);
System.out.println(" Please Enter any year you wish: ");
year = sc.nextInt();

LeapYear ly = new LeapYear();
leap = ly.CheckLeapYear(year);

if(leap != 0) {
    System.out.format("\n %d is a Leap Year. \n", year);
}
else {
    System.out.format("\n %d is NOT a Leap Year. \n", year);
}
}
```

OUTPUT

ANALYSIS

LeapYear Class Analysis:

In this java leap year program, first, we declared an integer function **CheckLeapYear** with one argument. Within the function, we used the If statement to check whether the given

First, we created an instance / created an Object of the **LeapYear** Class

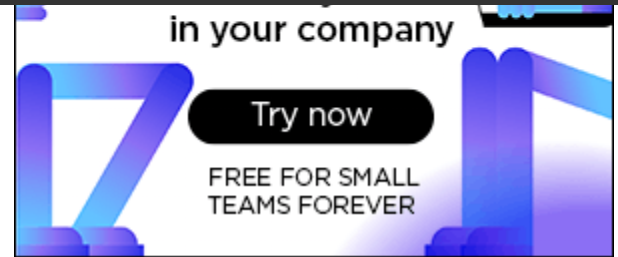
```
LeapYear ly = new LeapYear();
```


Next, we are calling the **CheckLeapYear** method.

```
leap = ly.CheckLeapYear(year);
```

Lastly, **System.out.println** statement will print the output.


Placed Under: [Java Programs](#)

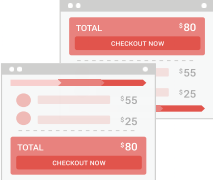
[C](#)[C#](#)[Java](#)[R](#)[Python](#)[SQL](#)[MySQL](#)[Js](#)[BI Tools](#)[More](#)

 Optimize

Google Optimize makes A/B testing as easy as 1-2-3.

[Test for free](#)





The visualizations show two overlapping browser window mockups. The top window shows a checkout page with a 'TOTAL' of \$80 and a 'CHECKOUT NOW' button. The bottom window shows a similar page but with a 'TOTAL' of \$55 and a 'CHECKOUT NOW' button. To the right of the windows are two vertical progress bars: the top one is at 55% and the bottom one is at 25%.

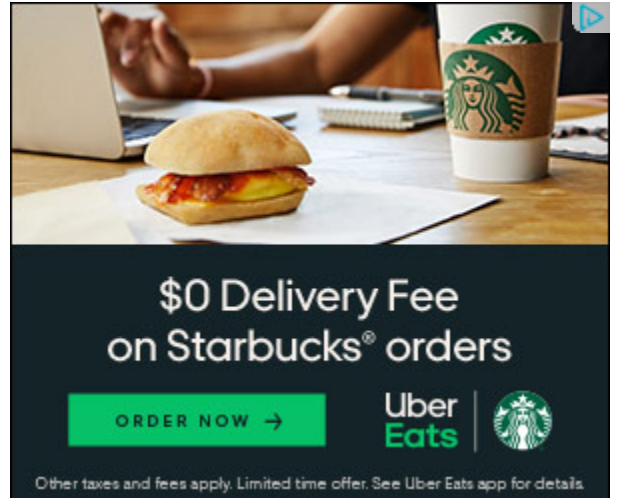
Builder

[Ad](#)

TRENDING POSTS

[SQL Concatenate Rows into String](#)[Python Program to find ASCII Value of a Character](#)[C Program to Print Even Numbers from 1 to N](#)[SQL ISNUMERIC Function](#)[JavaScript setMinutes Function](#)[C Program to find Power of a Number](#)[Update Strategy in Informatica](#)[MySQL Cross Join](#)[SQL QUOTENAME Function](#)[How to Replace String in SQL Server](#)

Create and Launch a Beautiful eCommerce Website Quickly with Weebly.


[LEARN MORE](#)

Ad



You Make It. Weebly Helps
You Make it a Business.
Create Your Online Store
Now.


[LEARN MORE](#)

[C](#)[C#](#)[Java](#)[R](#)[Python](#)[SQL](#)[MySQL](#)[Js](#)[BI Tools](#)[More](#)


\$0 Delivery Fee
on Starbucks® orders

[ORDER NOW →](#)

Uber
Eats




Other taxes and fees apply. Limited time offer. See Uber Eats app for details.



\$0 Delivery Fee
on Starbucks® orders

[ORDER NOW →](#)

Uber
Eats



Other taxes and fees apply. Limited time offer. See Uber Eats app for details.

[C](#)[C#](#)[Java](#)[R](#)[Python](#)[SQL](#)[MySQL](#)[Js](#)[BI Tools](#)[More](#)[C PROGRAMS](#)[JAVA PROGRAMS](#)[SQL FAQ'S](#)[PYTHON PROGRAMS](#)[SSIS](#)[TABLEAU](#)[JAVASCRIPT](#)

COPYRIGHT © 2020 | TUTORIAL GATEWAY · ALL RIGHTS RESERVED BY SURESH
[HOME](#) | [ABOUT US](#) | [CONTACT US](#) | [PRIVACY POLICY](#)