Name	**Shubhan Singh**
UID no.	**2022300118**
Experiment No.	**3**

PROBLEM STATEMENT :	*A person is planting a Bomb in an enemy area. He is deciding the date and time of the explosion beforehand. Now calculate how many days are left for the explosion.*
THEORY:	**Date class in Java:** The **java.util.Date** class is a built-in class in Java that represents a specific instant in time with millisecond precision. It provides several methods for working with dates and times, such as comparing dates, converting dates to strings, and calculating the difference between two dates. However, there are some quirks about the year and month handling in the **java.util.Date** class that are worth noting: 1. Year: The year in the **java.util.Date** class starts from 1900. For example, to represent the year 2023, you need to pass 123 as the year parameter while creating the **Date** object. This means that 0 corresponds to the year 1900, 1 corresponds to 1901, and so on. 2. Month: The month in the **java.util.Date** class is 0-based, which means that January is represented by 0, February by 1, and so on. This can lead to some confusion and errors, as programmers are used to thinking of months starting from 1. It's also worth noting that the **java.util.Date** class has some limitations and is considered to be outdated. It doesn't handle time zones or daylight saving time, and some of its methods have been deprecated in favor of the newer **java.time** package introduced in Java 8. If you're working with dates and times in Java, it's recommended to use the **java.time** package instead.

PROGRAM:

```java
// import necessary classes
import java.util.Date;
import java.util.Scanner;

// define a public class named Bomb
public class Bomb {
    public static void main(String[] args) {
        // declare and initialize variables to store the
explosion date
        int exp_day, exp_month, exp_year;

        // prompt the user to enter the explosion date
        System.out.println("Enter Date of explosion (as day,
month and year, seperated by spaces/newlines)");

        // create a Scanner object to read input from the
user
        Scanner sc = new Scanner(System.in);

        // read the input values for the explosion date from
the user and store them in the variables
        exp_day = sc.nextInt();
        exp_month = sc.nextInt();
        exp_year = sc.nextInt();

        // create a new Date object to represent the current
date and time
        Date current = new Date();

        // create a new Date object to represent the
explosion date, adjusting the year and month to account for
the
        // Date class's quirks
        Date explosion = new Date(exp_year - 1900, exp_month
- 1, exp_day);

        // calculate the number of milliseconds between the
current date and time and the explosion date and time
        long current_time = current.getTime();
        long exp_time = explosion.getTime();
        long ms_between_dates = exp_time - current_time;

        // calculate the number of milliseconds in a day and
use it to calculate the number of days between the
        // current date and time and the explosion date and
time
        long ms_in_a_day = 1000 * 60 * 60 * 24;
        long days_between_dates = ms_between_dates /
ms_in_a_day + 1;

        // output the number of days until the explosion to
the user
        System.out.println("The number of days till the
explosion is: " + days_between_dates);
    }
}
```

RESULT:

```
Enter Date of explosion (as day, month and year, seperated by spaces/newlines)
26 5 2023
The number of days till the explosion is: 34

Process finished with exit code 0
```

(Current date was 22/4/23)