Day 58 coding Statement : Bucket Filling

Nejiya has a bucket having a capacity of K liters. It is already filled with X liters of water.

Find the maximum amount of extra water in liters that Nejiya can fill in the bucket without overflowing.

Input Format :

The first and only line of each test case contains two space separated integers K and X - as mentioned in the problem.

Output Format :

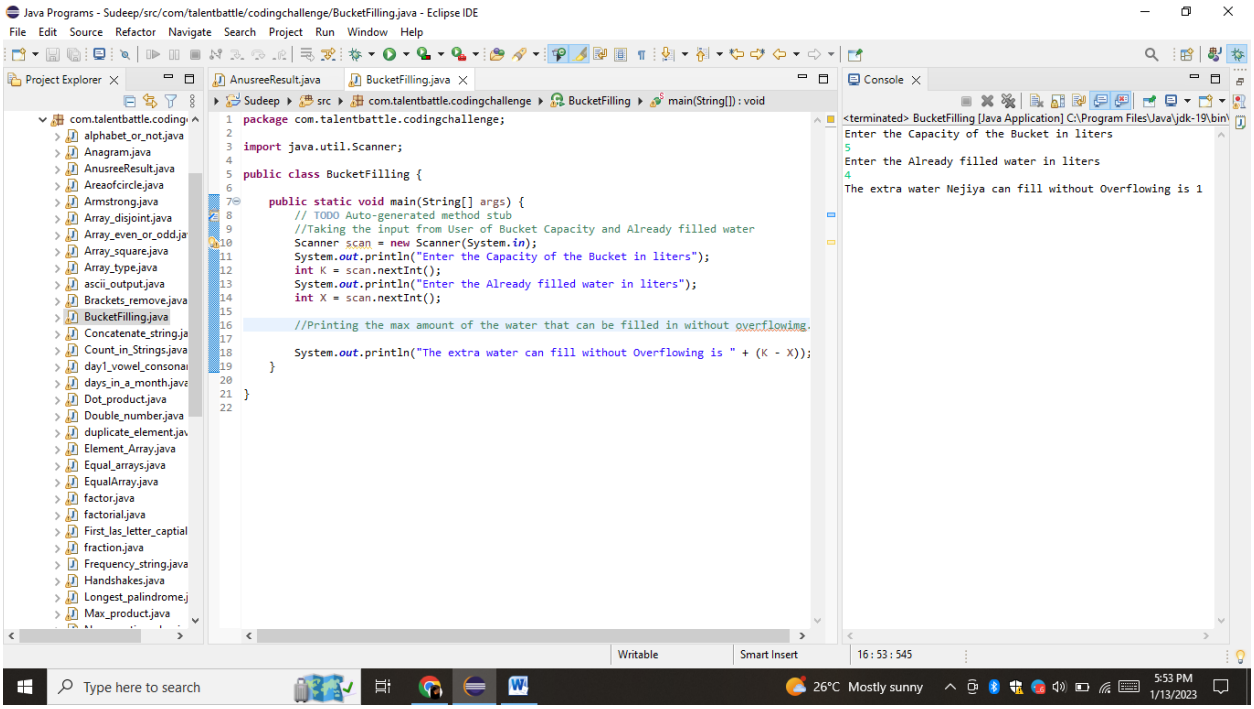
For each test case, output in a single line, the amount of extra water in liters that Nejiya can fill in the bucket without overflowing.

Sample Input : 5 4

Sample Output : 1

Program :

```
package com.talentbattle.codingchallenge;

import java.util.Scanner;

public class BucketFilling {

	public static void main(String[] args) {
		// TODO Auto-generated method stub
		//Taking the input from User of Bucket Capacity and Already filled water
		Scanner scan = new Scanner(System.in);
		System.out.println("Enter the Capacity of the Bucket in liters");
		int K = scan.nextInt();
		System.out.println("Enter the Already filled water in liters");
		int X = scan.nextInt();

		//Printing the max amount of the water that can be filled in without
overflowimg.

		System.out.println("The extra water can fill without Overflowing is " +
(K - X));
	}

}
```




```java
package com.talentbattle.codingchallenge;

import java.util.Scanner;

public class BucketFilling {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        //Taking the input from User of Bucket Capacity and Already filled water
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter the Capacity of the Bucket in liters");
        int K = scan.nextInt();
        System.out.println("Enter the Already filled water in liters");
        int X = scan.nextInt();

        //Printing the max amount of the water that can be filled in without overflowimg.

        System.out.println("The extra water can fill without Overflowing is " + (K - X));
    }

}
```