**Problem Description**  
Print "Hello World !" in the output.  
**Output Format**

Print in a single line "Hello World !"  
**Example Output**

Hello World !  
Note - You don't have to print "" in the output

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

        System.out.print("Hello World !");

    }

}

**Problem Description**  
Print the following text:

Hello 40

50 World !  
**Output Format**

Print the output specifed in the problem description.  
**Example Output**

Hello 40  
50 World !

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

       System.out.println("Hello 40");

       System.out.print("50 World !");

    }

}

**Problem Description**  
Print the following text in the output:

Hello

World !

Note: There is a empty line you need print between Hello and World !  
**Output Format**

Print the required text in the output.  
**Example Output**

Hello

World !

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

        System.out.println("Hello");

        System.out.println();

        System.out.print("World !");

    }

}

**Problem Description**  
Print the text

Hello

World !

**Output Format**

Hello

World !

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

       System.out.println("Hello");

       System.out.print("World !");

    }

}

**Problem Description**

Print the following pattern in output

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

**Output Format**

Print the following pattern in 5 lines with 5, 4, 3, 2, 1 stars in each line

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

**Example Output**

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

        System.out.println("\*\*\*\*\*");

        System.out.println("\*\*\*\*");

        System.out.println("\*\*\*");

        System.out.println("\*\*");

        System.out.print("\*");

    }

}

**Problem Description**

Print the following pattern in output

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

**Output Format**

Print the following pattern in 5 lines with 4 spaces and 1 star, 3 spaces and 2 stars, 2 spaces and 3 stars, 1 spaces and 4 stars, 0 spaces and 5 stars in respectively lines

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

**Example Output**

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

        System.out.println("    \*");

        System.out.println("   \*\*");

        System.out.println("  \*\*\*");

        System.out.println(" \*\*\*\*");

       System.out.print("\*\*\*\*\*");

    }

}

**Problem Description**  
Perform the following operations on 6 and 3 and print their respective outputs in different lines.

1. Addition  
2. Subtraction  
3. Multiplication  
4. Division

**Output Format**

Print the output of 6+3, 6-3, 6\*3, 6/3 in separate lines.

**Example Output**

9  
3  
18  
2.0

**Example Explanation**

6 + 3 = 9  
6 - 3 = 3  
6 \* 3 = 18  
6 / 3 = 2.0

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

        System.out.println(6+3);

        System.out.println(6-3);

        System.out.println(6\*3);

        System.out.print(6/3.0); or System.out.print((float)6/3);

    }

}

**Problem Description**  
Take two names **A** and **B** as input from the user, print "A says Hi to B" (Without quotations), where **A** and **B** are the names in input.  
  
**Problem Constraints**

1 <= len(A), len(B) <= 15  
Characters in A and B are in lowercase English Alphabets.

**Input Format**

There are two input lines  
The first line has a string A.  
The second line has a string B.

**Output Format**

Print in a single line A says Hi to B.

**Example Input**

Input:-

Ram

Shyam

**Example Output**

Output:-

Ram says Hi to Shyam

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

        Scanner ip = new Scanner(System.in);

        String A=ip.next();

        String B=ip.next();

        System.out.print(A+" says Hi to "+B);

    }

}

**Problem Description**

Print the first five letters of the English alphabet i.e. A, B, C, D and E.

**Output Format**

Print the characters in separate lines.

**Example Output**

A  
B  
C  
D  
E

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

        System.out.println("A");

        System.out.println("B");

        System.out.println("C");

        System.out.println("D");

        System.out.print("E");

    }

}

**Problem Description**

Given two numbers **A** and **B**. Print A+B, A\*B, A-B, A/B in the same order.

**Problem Constraints**

1 <= A, B <= 104

**Input Format**

There are two input lines  
The first line has a single integer **A**.  
The second line has a single integer **B**.

**Output Format**

Print in a single line separated by space A+B, A\*B, A-B, A/B in the same order.

**Example Input**

Input 1:-

4

5

Input 2:-

16

2

**Example Output**

Output 1:-

9 20 -1 0

Output 2:-

18 32 14 8

**Example Explanation**

Explanation 1:-

4 + 5 = 9, 4 \* 5 = 20, 4 - 5 = -1, 4 / 5 = 0

Explanation 2:-

16 + 2 = 18, 16 \* 2 = 32, 16 - 2 = 14, 16 / 2 = 8

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

        Scanner ip=new Scanner(System.in);

        int A=ip.nextInt();

        int B=ip.nextInt();

        System.out.print((A+B)+" "+(A\*B)+" "+(A-B)+" "+(A/B));

    }

}

**Problem Description**

Given two numbers **A** and **B**. Concatenate the two numbers and print it.

**Problem Constraints**

1 <= A, B <= 104

**Input Format**

There are two input lines  
The first line has a single integer **A**.  
The second line has a single integer **B**.

**Output Format**

Print in a single line the concatenated number.

**Example Input**

Input 1:-

4

5

Input 2:-

16

2

**Example Output**

Output 1:-

45

Output 2:-

162

**Example Explanation**

Explanation 1:-

Concatenation of 4 and 5 is 45.

Explanation 2:-

Concatenation of 16 and 2 is 162.

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

        Scanner ip=new Scanner(System.in);

        int A=ip.nextInt();

        int B=ip.nextInt();

        System.out.print(A+""+B);

    }

}

**Problem Description**

You will be given an integer in the input. You need to add 10 to it and print the result to the output.

**Input Format**

One integer value in the input.

**Output Format**

Print a single integer the sum.

**Problem Constraints**

1 <= N <= 1000

**Example Input**

Input 1:-

3

Input 2:-

11

**Example Output**

Output 1:-

13

Output 2:-

21

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

        Scanner ip=new Scanner(System.in);

        int N=ip.nextInt();

        System.out.print(10+N);

    }

}

**Problem Description**

Given an english sentence, take it in the input and print it in the output.

**Input Format**

Single line containing an english sentence.

**Output Format**

Print in a single line the sentence in the input.

**Example Input**

**Input 1:**

Harry loves Hagrid !

**Input 2:**

I am a Scaler

**Example Output**

**Output 1:**

Harry loves Hagrid !

**Output 2:**

I am a Scaler

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

       Scanner ip=new Scanner(System.in);

       String line=ip.nextLine();

       System.out.print(line);

    }

}

**Problem Description**

Take a name **A** as input from the user and print "Hello A", where A is the name in input.

**Problem Constraints**

1 <= len(A) <= 15 Characters in A are in lowercase English Alphabets.

**Input Format**

There is a single input line, which is the string \*\*A\*\*.

**Output Format**

Print in a single line "Hello A" without quotes.

**Example Input**

Input 1:-

Ram

Input 2:-

Shyam

**Example Output**

Output 1:-

Hello Ram

Output 2:-

Hello Shyam

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

       Scanner ip=new Scanner(System.in);

       String A=ip.next();

       System.out.print("Hello "+A);

    }

}

**Problem Description**

Given total bills amount and amount of a single bill. Print number of bills.  
  
**Note :** The total amount is equally splitted in all bills. The number of bills should be an integer value.

**Input Format**

The first line contains a real number N denoting the total budget.

The second line contains an integer M denoting the value of a single bill.

**Output Format**

Print in a single line denoting the total number of bills that can fit in the total budget.

**Problem Constraints**

1 <= N <= 10000  
1 <= M <= 100

**Example Input**

Input:-

126.3

5

**Example Output**

Output:-

25

**Note:** The **problem constraints** mean that when we test your code, the test cases used in the backend can have input values only within those constraints. You need not implement them in your code. You need to make sure your code will work for all such input values!

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

      Scanner ip=new Scanner(System.in);

      double N=ip.nextDouble();

      int M=ip.nextInt();

      System.out.print((int)N/M);

    }

}

**Problem Description**

Given the value of a single bill and the number of bills you received, print the total value of the bills.  
  
**Note:** The value of all the bills are same

**Problem Constraints**

1 <= N <= 100  
1 <= M <= 100

**Input Format**

The first line of the input is an integer N denoting the value of a single bill.  
The second line of the input is an integer M denoting the number of bills.

**Output Format**

Print in a single line denoting the total value of bills.

**Example Input**

Input:-  
12  
10

**Example Output**

Output:-  
120

**Example Explanation**

**Note:** The **problem constraints**mean that when we test your code, the test cases used in the backend can have input values only within those constraints. **You need not implement them in your code.**You need to make sure your code will work for all such input values!

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

        Scanner ip=new Scanner(System.in);

        System.out.print(ip.nextInt() \* ip.nextInt());

    }

}

**Problem Description**

You're going to write some code to help you cook a gorgeous lasagna from your favorite cookbook. According to your cookbook, the Lasagna should be in the oven for 40 minutes. Given the time (in minutes), the lasagna has been in the oven, find how many more minutes the lasagna still needs to bake for.

**Problem Constraints**

0 <= N <= 40

**Input Format**

The only first line contains the integer N, denoting the actual time (in minutes) the lasagna has been in the oven for.

**Output Format**

Print in a single line how many minutes the lasagna needs to bake.

**Example Input**

Input:   
30

**Example Output**

Output:   
10

**Example Explanation**

**Note:** The **problem constraints** mean that when we test your code, the test cases used in the backend can have input values only within those constraints.You need not implement them in your code.You need to make sure your code will work for all such input values!

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

        Scanner ip=new Scanner(System.in);

        System.out.print(40-ip.nextInt());

    }}

**Problem Description**

You'll write some code to help you cook a gorgeous lasagna from your favorite cookbook. Now, you also want to add a few layers to the lasagna. Assume \*\*each layer takes 2 minutes\*\* to prepare. Given the number of layers you want to add to the lasagna, find how many minutes you would spend making them.

**Input Format**

The only first line contains the integer N denoting the number of layers.

**Output Format**

Print in a single line how many minutes are required to prepare N layers.

**Example Input**

Input:-  
2

**Example Output**

Output:-  
4

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

       Scanner ip=new Scanner(System.in);

       System.out.print(2\*ip.nextInt());

    }

}

**Problem Description**

You wrote some code to help you cook a gorgeous lasagna from your favorite cookbook. Now, you want to find the total number of minutes you've been cooking for the sum of your preparation time and the time the lasagna has already spent baking in the oven. The preparation time of one layer is 2 minutes. Given the number of layers added to the lasagna and the number of minutes the lasagna has been baking in the oven, find the total elapsed cooking time (prep + bake) in minutes.

**Problem Constraints**

1 <= N <= 20  
0 <= M <= 40

**Input Format**

There are 2 lines in the input.  
The first line contains the integer N denoting the number of layers.  
The second line contains the integer M denoting the time the lasagna has already spent baking in the oven.

**Output Format**

Print in a single line the total elapsed cooking time.

**Example Input**

Input 1:-  
3  
20  
  
Input 2:-  
1  
29

**Example Output**

Output 1:-  
26  
  
Output 2:-  
31

🡪

import java.lang.\*;

import java.util.\*;

public class Main {

    public static void main(String[] args) {

        Scanner ip=new Scanner(System.in);

        System.out.print(2\*ip.nextInt()+ip.nextInt());

    }

}