Java's System.out.printf function can be used to print formatted output. The purpose of this exercise is to test your understanding of formatting output using printf.

To get you started, a portion of the solution is provided for you in the editor; you must format and print the input to complete the solution.

Input Format

Every line of input will contain a String followed by an integer.
Each String will have a maximum of 10 alphabetic characters, and each integer will be in the inclusive range from 0 to 999.

Output Format

In each line of output there should be two columns:
The first column contains the String and is left justified using exactly 15 characters.
The second column contains the integer, expressed in exactly 3 digits; if the original input has less than three digits, you must pad your output's leading digits with zeroes.

Sample Input

```
java 100
cpp 65
python 50
```

Sample Output

```
================================
java           100
cpp            065
python         050
================================
```

Explanation

Each String is left justified with trailing whitespace through the first 15 characters. The leading digit of


```java
import java.util.Scanner;

public class Solution {

    public static void main(String[] args) {
            Scanner sc=new Scanner(System.in);
            System.out.println("================================");
            for(int i=0;i<3;i++){
                String s1=sc.next();
                int x=sc.nextInt();
                System.out.printf("%-15s%03d\n", s1, x);
            }
            System.out.println("================================");

    }
}
```


Java's System.out.printf function can be used to print formatted output. The purpose of this exercise is to test your understanding of formatting output using printf.

To get you started, a portion of the solution is provided for you in the editor; you must format and print the input to complete the solution.

Input Format

Every line of input will contain a String followed by an integer.
Each String will have a maximum of 10 alphabetic characters, and each integer will be in the inclusive range from 0 to 999.

Output Format

In each line of output there should be two columns:
The first column contains the String and is left justified using exactly 15 characters.
The second column contains the integer, expressed in exactly 3 digits; if the original input has less than three digits, you must pad your output's leading digits with zeroes.

Sample Input

```
java 100
cpp 65
python 50
```

Sample Output

```
================================
java           100
cpp            065
python         050
================================
```

Explanation

Each String is left justified with trailing whitespace through the first 15 characters. The leading digit of


```java
import java.util.Scanner;

public class Solution {

    public static void main(String[] args) {
            Scanner sc=new Scanner(System.in);
            System.out.println("================================");
            for(int i=0;i<3;i++){
                String s1=sc.next();
                int x=sc.nextInt();
                System.out.printf("%-15s%03d\n", s1, x);
            }
            System.out.println("================================");

    }
}
```


Java's System.out.printf function can be used to print formatted output. The purpose of this exercise is to test your understanding of formatting output using printf.

To get you started, a portion of the solution is provided for you in the editor; you must format and print the input to complete the solution.

Input Format

Every line of input will contain a String followed by an integer.
Each String will have a maximum of 10 alphabetic characters, and each integer will be in the inclusive range from 0 to 999.

Output Format

In each line of output there should be two columns:
The first column contains the String and is left justified using exactly 15 characters.
The second column contains the integer, expressed in exactly 3 digits; if the original input has less than three digits, you must pad your output's leading digits with zeroes.

Sample Input

```
java 100
cpp 65
python 50
```

Sample Output

```
================================
java           100
cpp            065
python         050
================================
```

Explanation

Each String is left justified with trailing whitespace through the first 15 characters. The leading digit of


```java
import java.util.Scanner;

public class Solution {

    public static void main(String[] args) {
            Scanner sc=new Scanner(System.in);
            System.out.println("================================");
            for(int i=0;i<3;i++){
                String s1=sc.next();
                int x=sc.nextInt();
                System.out.printf("%-15s%03d\n", s1, x);
            }
            System.out.println("================================");

    }
}
```


Java's System.out.printf function can be used to print formatted output. The purpose of this exercise is to test your understanding of formatting output using printf.

To get you started, a portion of the solution is provided for you in the editor; you must format and print the input to complete the solution.

Input Format

Every line of input will contain a String followed by an integer.
Each String will have a maximum of 10 alphabetic characters, and each integer will be in the inclusive range from 0 to 999.

Output Format

In each line of output there should be two columns:
The first column contains the String and is left justified using exactly 15 characters.
The second column contains the integer, expressed in exactly 3 digits; if the original input has less than three digits, you must pad your output's leading digits with zeroes.

Sample Input

```
java 100
cpp 65
python 50
```

Sample Output

```
================================
java           100
cpp            065
python         050
================================
```

Explanation

Each String is left justified with trailing whitespace through the first 15 characters. The leading digit of


```java
import java.util.Scanner;

public class Solution {

    public static void main(String[] args) {
            Scanner sc=new Scanner(System.in);
            System.out.println("================================");
            for(int i=0;i<3;i++){
                String s1=sc.next();
                int x=sc.nextInt();
                System.out.printf("%-15s%03d\n", s1, x);
            }
            System.out.println("================================");

    }
}
```