Objective

This challenge will help you to learn how to take a character, a string and a sentence as input in C.

To take a single character ***ch*** as input, you can use `scanf("%c", &ch );` and `printf("%c", ch)` writes a character specified by the argument char to stdout

```
char ch;
scanf("%c", &ch);
printf("%c", ch);
```

This piece of code prints the character ***ch***.

You can take a string as input in C using `scanf("%s", s)`. But, it accepts string only until it finds the first space.

In order to take a line as input, you can use `scanf("%[^\n]%*c", s);` where ***s*** is defined as `char s[MAX_LEN]` where ***MAX_LEN*** is the maximum size of ***s***. Here, `[]` is the scanset character. `^\n` stands for taking input until a newline isn't encountered. Then, with this `%*c`, it reads the newline character and here, the used `*` indicates that this newline character is discarded.

Note: The statement: `scanf("%[^\n]%*c", s);` will not work because the last statement will read a newline character, `\n`, from the previous line. This can be handled in a variety of ways. One way is to use `scanf("\n");` before the last statement.

Task

You have to print the character, ***ch***, in the first line. Then print ***s*** in next line. In the last line print the sentence, ***sen***.

Input Format

First, take a character, ***ch*** as input.
Then take the string, ***s*** as input.
Lastly, take the sentence ***sen*** as input.

Constraints

Change Theme Language: C

```c
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>

#define MAX_LEN 100

int main()
{

    /* Enter your code here. Read input from STDIN. Print output to STDOUT */

    char c,s[MAX_LEN],sen[MAX_LEN];

    scanf("%c", &c);
    scanf("%s", s);
    scanf("\n");
    scanf("%[^\n]%*c", sen);

    printf("%c\n", c);
    printf("%s\n", s);
    printf("%s\n", sen);

    return 0;
}
```

Line: 17 Col: 17