



# Overloading Ostream Operator ☆

19/44 challenges solved

Rank: 22572 | Points: 270 ⓘ



Your Overloading Ostream Operator submission got 20.00 points.

[Try the next challenge](#) | [Try a Random Challenge](#)



## Problem

[Submissions](#)[Leaderboard](#)[Editorial](#) ⓘ

The task is to overload the << operator for Person class in such a way that for p being an instance of class Person the result of:

```
std::cout << p << " " << <some_string_value> << std::endl;
```

produces the following output:

```
first_name=<first_name>,last_name=<last_name> <some_string_value>
```

where:

- <first\_name> is the value of p's first\_name\_
- <last\_name> is the value of p's last\_name\_
- <some\_string\_value> is an arbitrary std::string value

## Input Format

The input is read by the provided locked code template. In the only line of the input there are 3 space-separated strings first\_name, last\_name, event. The values of first\_name and last\_name will be used to create an object p of type Person. The value of event will be used by the provided code to produce the output.

## Constraints

- Each word in the input contains only English letters and is no longer than 15 characters

## Output Format

The output should be produced by the provided locked code template. This code will use the implementation of Person public methods and the overloaded << operator to produce the output. Specifically, the output will be produced by the following code:

```
cout << p << " " << event << endl;
```

## Sample Input 0

```
john doe registered
```

## Sample Output 0

```
first_name=john,last_name=doe registered
```



C++



```
1  #include <iostream>
2
3  using namespace std;
4
5  class Person {
6  public:
7      Person(const string& first_name, const string& last_name) : first_name_(first_name),
        last_name_(last_name) {}
8      const string& get_first_name() const {
9          return first_name_;
10     }
11     const string& get_last_name() const {
12         return last_name_;
13     }
14 private:
15     string first_name_;
16     string last_name_;
17 };
18
19 ostream & operator << (ostream& out, Person &ob)
20 {
21     out << "first_name=" << ob.get_first_name() << ",last_name=" << ob.get_last_name();
22     return out;
23 }
24
25
26 int main() {
27     string first_name, last_name, event;
28     cin >> first_name >> last_name >> event;
29     auto p = Person(first_name, last_name);
30     cout << p << " " << event << endl;
```

Line: 19 Col: 37

☒ Upload Code as File ☐ Test against custom input

Run Code

Submit Code

You have earned 20.00 points!

19/44 challenges solved.

43%



## Congratulations

You solved this challenge. Would you like to challenge your friends?



Next Challenge

Test case 0 ✓

Test case 1 ✓

Test case 2 ✓

Test case 3 ✓

Test case 4 ✓

Test case 5 ✓

Test case 6 ✓

Test case 7 ✓

Compiler Message

**Success**

Input (stdin)

[Download](#)**john doe registered{-truncated-}**[Download to view the full testcase](#)

Expected Output

[Download](#)**first\_name=john,last\_name=doe registered{-truncated-}**[Download to view the full testcase](#)