





Overloading Ostream Operator ☆



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;</pre>

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;</pre>

Sample Input 0

john doe registered

Sample Output 0

first_name=john,last_name=doe registered

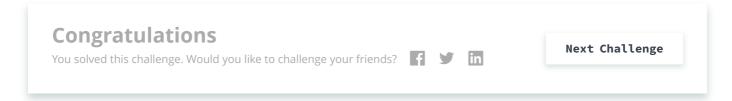


```
C++
      #include <iostream>
  1
  2
  3
      using namespace std;
  4
  5
      class Person {
  6
      public:
  7
          Person(const string& first_name, const string& last_name) : first_name_(first_name),
       last_name_(last_name) {}
  8
          const string& get_first_name() const {
  9
             return first_name_;
 10
          }
           const string& get last name() const {
 11
 12
             return last_name_;
 13
          }
 14
      private:
          string first_name_;
 15
 16
          string last_name_;
 17
      };
 18
 19
      ostream & operator << (ostream& out, Person &ob)</pre>
 20
 21
          out << "first name=" << ob.get first name() << ",last name="<<ob.get last name();</pre>
 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;</pre>
                                                                                      Line: 19 Col: 37
1 Upload Code as File
                   ■ Test against custom input
                                                                         Run Code
                                                                                        Submit Code
```

You have earned 20.00 points!

19/44 challenges solved.





Test case 0 ⊗	Compiler Message	
Test case 1 ⊗	Success	
Test case 2 ⊗	Input (stdin)	Download
Test case 3 ♂	<pre>john doe registered{-truncated-} Download to view the full testcase</pre>	
Test case 4 ⊗	Expected Output	Download
Test case 5 ⊗	<pre>first_name=john,last_name=doe registered{-truncated-} Download to view the full testcase</pre>	
Test case 6 ⊗		
Test case 7 ♂	•	

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature