

# Overloading ostream Operator

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
```