## Problem 4 – Students, Courses, Grades, Visits

You are given a **list of students score** given as text table with the following columns: **student** name, **course**, **grade**, number of **visits**. A student can have several grades and visits for the same course. Write a JavaScript function to **aggregate the results** and print then as **JSON string** as shown in the examples below.

### Input

The input is passed to the first JavaScript function found in your code as **array of strings** holding the table lines. The input data will always be valid and in the format described. There is no need to check it explicitly.

### Output

Print at the console a **JSON string** that holds the **courses** (in alphabetical order), the **average grade** and average visits for each course and a **list of students** for each course (in alphabetical order). **Duplicates** should be removed (all strings are **case-sensitive**). Please follow exactly the **JSON format** from the example below.

The average numbers should be **rounded to 2 digits** after the decimal point and printed **without trailing zeroes**:

* 5 🡪 5; 5.50 🡪 5.5; 5.491 🡪 5.49; 5.495 🡪 5.5; 5.000001 🡪 5; 5.500 🡪 5.5

### Constraints

* The numbers of **input lines** is between 1 and 10 000.
* The names of **students** and **courses** consists of Latin letters and spaces. Their **length** is between 1 and 50 characters. Leading and trailing **whitespace** should be removed.
* The values of **grades** and **visits** will be numbers in the range [0…50].
* Allowed working time: 0.2 seconds. Allowed memory: 16 MB.

**function** *solve*(input) {  
 **"use strict"**;  
  
 **let** results = {};  
 **for** (**let** row **of** input) {  
  
 **let** [name, course, grade, visits] = row.**split**(**'|'**);  
 name = name.trim();  
 course = course.trim();  
 **if**(!results.hasOwnProperty(course)){  
 results[course] = {**grades**: [], **visits**: [], **students**: []};  
 }  
  
 results[course].**grades**.push(grade);  
 results[course].**visits**.push(visits);  
 **if**(results[course].**students**.indexOf(name) == -1){  
 results[course].**students**.push(name);  
 }  
 }  
  
 **let** output = {};  
 **let** courses = Object.keys(results).sort();  
 **for** (**let** c **of** courses) {  
  
 **let** courseName = c;  
 **let** courseInfo = {  
 **avgGrade**: *average*(results[courseName].**grades**),  
 **avgVisits**: *average*(results[courseName].**visits**),  
 **students**: results[courseName].**students**.sort()  
 };  
 output[courseName] = courseInfo;  
 }  
 **console**.log(***JSON***.stringify(output))  
  
 **function** *average*(arr) {  
 **let** sum = 0;  
 **for** (**let** i **of** arr) {  
 sum += Number(i);  
 }  
 **let** avg = sum/arr.length;  
 avg = Number(avg.toFixed(2));  
 **return** avg;  
 }  
}

### Examples

|  |
| --- |
| **Input** |
| Peter Nikolov | PHP | 5.50 | 8  Maria Ivanova | Java | 5.83 | 7  Ivan Petrov | PHP | 3.00 | 2  Ivan Petrov | C# | 3.00 | 2  Peter Nikolov | C# | 5.50 | 8  Maria Ivanova | C# | 5.83 | 7  Ivan Petrov | C# | 4.12 | 5  Ivan Petrov | PHP | 3.10 | 2  Peter Nikolov | Java | 6.00 | 9 |
| **Output** |
| {"C#":{"avgGrade":4.61,"avgVisits":5.5,"students":["Ivan Petrov","Maria Ivanova","Peter Nikolov"]},"Java":{"avgGrade":5.92,"avgVisits":8,"students":["Maria Ivanova","Peter Nikolov"]},"PHP":{"avgGrade":3.87,"avgVisits":4,"students":["Ivan Petrov","Peter Nikolov"]}} |