To compare two JSON objects that have the same properties but may not be in the same order, you can follow these steps:

Convert both JSON objects to strings.

Sort the strings to ensure that the properties are in the same order.

Compare the sorted strings.

Here's how you can do it in JavaScript:

let obj1 = { name: "person 1", age: 5 };

let obj2 = { age: 5, name: "Person 1" };

// Convert the JSON objects to strings

let str1 = JSON.stringify(obj1);

let str2 = JSON.stringify(obj2);

// Sort the strings

let sortedStr1 = str1.split('').sort().join('');

let sortedStr2 = str2.split('').sort().join('');

// Compare the sorted strings

if (sortedStr1 === sortedStr2) {

console.log("The JSON objects are equivalent.");

} else {

console.log("The JSON objects are not equivalent.");

}

In this code, we first convert the JSON objects to strings using JSON.stringify(). Then, we split, sort, and join the characters in the strings to ensure that the properties are in the same order. Finally, we compare the sorted strings to determine if the JSON objects are equivalent, regardless of the property order.