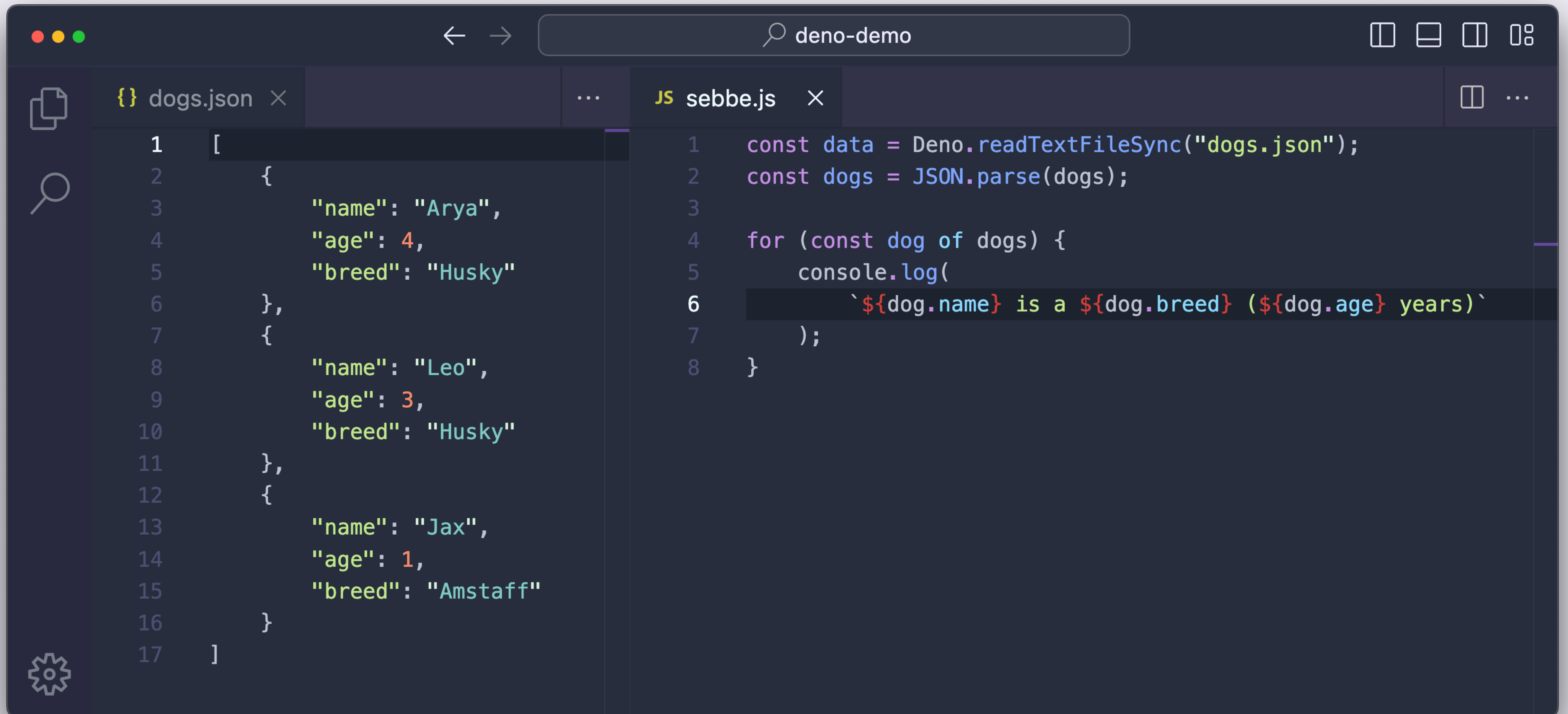


Läsa JSON



The image shows a code editor window with two tabs: `dogs.json` and `sebbe.js`. The `dogs.json` tab contains a JSON array of three dog objects. The `sebbe.js` tab contains JavaScript code that reads the `dogs.json` file, parses it, and logs each dog's information to the console using a template string.

```
1  [
2    {
3      "name": "Arya",
4      "age": 4,
5      "breed": "Husky"
6    },
7    {
8      "name": "Leo",
9      "age": 3,
10     "breed": "Husky"
11   },
12   {
13     "name": "Jax",
14     "age": 1,
15     "breed": "Amstaff"
16   }
17 ]
```

```
1  const data = Deno.readFileSync("dogs.json");
2  const dogs = JSON.parse(dogs);
3
4  for (const dog of dogs) {
5    console.log(
6      `${dog.name} is a ${dog.breed} (${dog.age} years)`
7    );
8  }
```

Skriva JSON

Obs! Vi kan inte bara "lägga till" när vi skriver till en JSON-fil

names.json

```
["Sebbe", "Erik", "Johannes"]
```

```
const filename = "names.json";  
const newName = "Sara";  
const data = JSON.stringify(newName);  
  
Deno.writeTextFileSync(filename, data, { append: true });
```

names.json

```
["Sebbe", "Erik", "Johannes"]"Sara"
```

Detta är **inte korrekt JSON**

Skriva JSON

Vi behöver först läsa in vår data, göra något med den, sedan spara det

names.json

```
["Sebbe", "Erik", "Johannes"]
```

```
const filename = "names.json";  
const data = Deno.readTextFileSync(filename);  
const names = JSON.parse(data);  
names.push("Sara");  
  
Deno.writeTextFileSync(filename, JSON.stringify(names));
```

names.json

```
["Sebbe", "Erik", "Johannes", "Sara"]
```

Detta **är korrekt** JSON