



Debugging **Tips & Tricks**

Benchmark loops using `console.time()` and `console.timeEnd()`

```
console.time('Timer1');

var items = [];

for(var i = 0; i < 100000; i++){
    items.push({index: i});
}

console.timeEnd('Timer1');
```

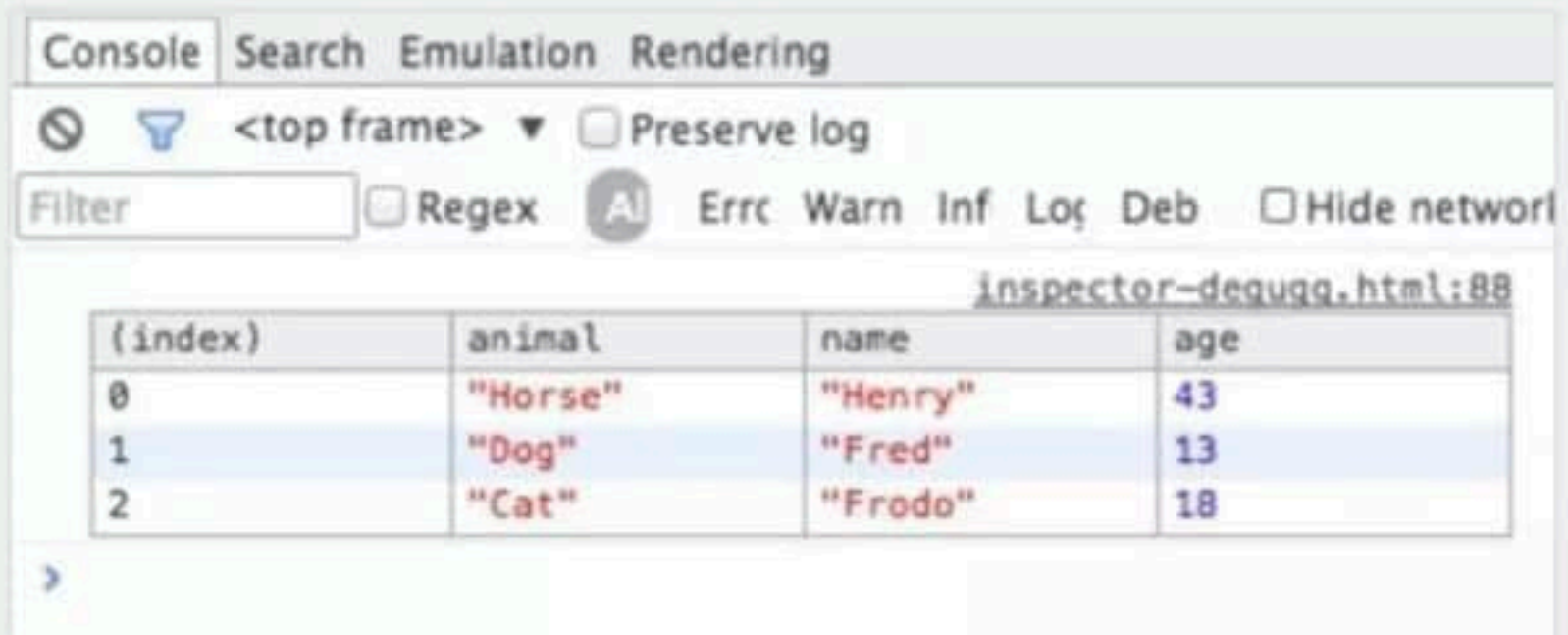


The screenshot shows the Chrome DevTools Console with the 'Console' tab selected. The code from the previous block is pasted into the console. The execution result shows the timing for 'Timer1' as 15.928ms. Below the timing result, the value 'undefined' is shown, which is the return value of the `console.timeEnd()` function.

```
> console.time('Timer1');
var items = [];
for(var i = 0; i < 100000; i++){
    items.push({index: i});
}
console.timeEnd('Timer1');
Timer1: 15.928ms
< undefined
```

Display objects as a table

```
var animals = [  
  { animal: 'Horse', name: 'Henry', age: 43 },  
  { animal: 'Dog', name: 'Fred', age: 13 },  
  { animal: 'Cat', name: 'Frodo', age: 18 }  
];  
  
console.table(animals);
```



The screenshot shows the Chrome DevTools Console with the 'Console' tab selected. The log shows a single entry from `inspector-debugg.html:88` displaying a table of animal data. The table has four columns: (index), animal, name, and age. The rows are indexed 0, 1, and 2, corresponding to the objects in the `animals` array. The 'animal' and 'name' columns are highlighted in red, and the 'age' column is highlighted in blue.

(index)	animal	name	age
0	"Horse"	"Henry"	43
1	"Dog"	"Fred"	13
2	"Cat"	"Frodo"	18

Log variables in the console in new way



```
console.log('name', name)
```

Since ES2015, we can use object shorthand notation whenever we want to log out things like this.



```
console.log({name})
```

A better way to log multiple items



```
const sunil = { name: "Sunil", member: true, id: 134323 };  
const ilaria = { name: "Ilaria", member: false, id: 489348};  
console.log(sunil);  
console.log(ilaria);
```



```
const sunil = { name: "Sunil", member: true, id: 134323 };  
const ilaria = { name: "Ilaria", member: false, id: 489348};  
console.log({sunil, ilaria});
```

VM98:1

```
▼ {sunil: {...}, ilaria: {...}} ⓘ  
  ► ilaria: {name: "Ilaria", member: false, id: 489348}  
  ► sunil: {name: "Sunil", member: true, id: 134323}  
  ► __proto__: Object
```


Group grouped logs

```
console.group('User Details');  
console.log('name: Sunil Sandhu');  
console.log('position: Software Developer');  
console.groupEnd();  
  
console.group('Account');  
console.log('Member Type: Premium Member');  
console.log('Member Since: 2018');  
console.log('Expiry Date: 20/12/2022');  
console.groupEnd();
```

▼ User Details	<u>VM419:1</u>
name: Sunil Sandhu	<u>VM419:2</u>
position: Software Developer	<u>VM419:3</u>
▼ Account	<u>VM419:5</u>
Member Type: Premium Member	<u>VM419:6</u>
Member Since: 2018	<u>VM419:7</u>
Expiry Date: 20/12/2022	<u>VM419:8</u>