

## JavaScript: Using the console Object ... by Fred Brack

```
// console.log methods; cut-and-paste to test; use Ctrl-Shift-I or -K to see the console
console.clear();
console.time("Whole Program"); // start a named timer
console.info("This is an informational message: we just started a timer");
console.debug("This is a debug message");
console.warn("This is a warning message");
console.error("This is an error message");
console.assert(false,"This message appears only when the assertion is false");
console.group("console.log/table demo:");
const person1 = {firstName:"Fred",lastName:"Brack"};
const person2 = {firstName:"Kathy",lastName:"Brack"};
console.info("The two 'person' properties will be displayed individually");
console.log(person1); // this will just identify person contents, not property name
console.log(person2);
console.info("But you could put header text on the console.log line");
console.log("Person 1: ",person1);
console.info("Or you could use the dir method to display person1");
console.dir(person1);
console.info("Displaying the properties within an object gives more detail");
console.log({person1,person2}); // this will name the 'person' property first
console.info("And here is how the object looks when presented in table format");
console.table([person1,person2]); // this will name the 'person' variable first
console.groupEnd();
console.info("You can also apply CSS formatting to text using %c");
console.info("This is %cIMPORTANT!","color:yellow; background-color:red");
console.info("The following console options are not demonstrated:");
console.info("  count, countReset, dirxml, groupCollapsed, timeLog, trace; %d, $f, %i, %o, %s");
console.info("Note, however, that trace is useful within a function for determining how you got there");
console.info("and count is useful to determine how many times you entered a function");
console.timeEnd("Whole Program");
//END - Fred Brack, fbrack@bracksc.com, last updated Sep 13, 2020
```

Console was cleared.

⚠ Timer "Whole Program" already exists.

ℹ This is an informational message: we just started a timer

⚠ This is a warning message

❌ ▶ This is an error message

❌ Assertion failed: This message appears only when the assertion is false

▼ console.log/table demo:

ℹ The two 'person' properties will be displayed individually

- ▶ Object { firstName: "Fred", lastName: "Brack" }
- ▶ Object { firstName: "Kathy", lastName: "Brack" }

ℹ But you could put header text on the console.log line

Person 1: ▶ Object { firstName: "Fred", lastName: "Brack" }

ℹ Or you could use the dir method to display person1

▼ {...}

- firstName: "Fred"
- lastName: "Brack"
- ▶ <prototype>: Object { ... }

ℹ Displaying the properties within an object gives more detail

▶ Object { person1: {...}, person2: {...} }

ℹ And here is how the object looks when presented in table format

console.table()

(index)	firstName	lastName
0	Fred	Brack
1	Kathy	Brack

ℹ You can also apply CSS formatting to text using %c

ℹ This is **IMPORTANT!**

ℹ The following console options are not demonstrated:

ℹ count, countReset, dirxml, groupCollapsed, timeLog, trace; %d, \$f, %i, %o, %s

ℹ Note, however, that trace is useful within a function for determining how you got there

ℹ and count is useful to determine how many times you entered a function

Whole Program: 658135ms - timer ended

» |