| SL | Overloading | Overriding |
|---|---|---|
| 1 | Method overloading is a compile-time polymorphism. | Method overriding is a run-tin |
| 2 | It helps to increase the readability of the program. | It is used to grant the specific is already provided by its par |
| 3 | It occurs within the class. | It is performed in two classes |
| 4 | Method overloading may or may not require inheritance. | Method overriding always ne |
| 5 | In method overloading, methods must have the same name and different signatures. | In method overriding, methoc same signature. |
| 6 | In method overloading, the return type can or can not be the same, but we just have to change the parameter. | In method overriding, the retu variant. |

Overloading and Overriding in JavaScript

JavaScript does not support function overloading natively. If we will add functions with the same name a the last defined function.

It is true that JavaScript supports overriding not overloading. When you define multiple functions which defined will override all the previously defined ones and every time when you invoke a function, the last

Code Examples

Data Abstraction

JavaScript is a weakly typed language and does not have the classical built-in support for abstraction lil has the capability to support OOP, and thus can achieve OOP functionalities, which is known as Object abstraction can be achieved by inheritance and composition.

In JavaScript, prototypes and closures can help us accomplish data abstraction.

```
function Employee(name, age, baseSalary) {
    this.name = name;
    this.age = age;
    this.baseSalary = baseSalary

    let monthlyBonus = 1000;

    // This is not exposed to the user
    let calculateMonthlyBonus = function () {
        let finalSalary = baseSalary + monthlyBonus
        console.log("Final Salary is : ", finalSalary)
    }
```