**Prototype en JavaScript**

En JavaScript, un prototipo es un objeto que sirve como plantilla para crear nuevos objetos. Cuando se crea un nuevo objeto, este hereda las propiedades y métodos del prototipo asociado. Esto significa que los objetos pueden compartir propiedades y comportamientos comunes.

**¿Cómo funciona?**

* **Prototipo por defecto:** Cada objeto en JavaScript tiene un prototipo asociado. Si no se especifica explícitamente, el prototipo por defecto es Object.prototype.
* **Encadenamiento de prototipos:** Los prototipos pueden tener sus propios prototipos, creando una cadena de prototipos. Cuando se busca una propiedad o método en un objeto, JavaScript busca primero en el objeto mismo. Si no la encuentra, busca en su prototipo y así sucesivamente hasta llegar a Object.prototype.
* **Creación de prototipos personalizados:** Puedes crear tus propios prototipos utilizando la propiedad \_\_proto\_\_ o la función Object.create().

**Ejemplo:**

JavaScript

function Person(name, age) {

this.name = name;

this.age = age;

}

Person.prototype.greet = function() {

console.log("Hola, mi nombre es " + this.name);

}; const person1 = new Person("Juan", 30); const person2 = new Person("María", 25); person1.greet(); // Output: Hola, mi nombre es Juan person2.greet(); // Output: Hola, mi nombre es María

Use code [with caution.](file:///C:/faq#coding)

En este ejemplo, Person.prototype es el prototipo para los objetos creados a partir de la función Person. Ambos objetos, person1 y person2, heredan el método greet del prototipo.

**Ventajas de los prototipos:**

* **Flexibilidad:** Los prototipos permiten modificar el comportamiento de los objetos en tiempo de ejecución.
* **Reutilización de código:** Los prototipos permiten compartir código común entre diferentes objetos.
* **Heredabilidad:** Los prototipos proporcionan un mecanismo de herencia basado en objetos.

**Conclusión**

Los prototipos son una característica fundamental de JavaScript que permiten crear objetos con propiedades y métodos heredados de un prototipo común. Comprender los prototipos es esencial para escribir código JavaScript eficiente y modular.