 All Methods of Objects

Adds the greet method to the given person object:

let person = {

  firstName: 'John',

  lastName: 'Doe'

};

function greet() {

  console.log('Hello, World!');

}

person.greet = greet;

person.greet();

2.First, use a function expression to define a function and assign it to the greet property of the person object.

\*Then, call the method greet() method.

Besides using a function expression, you can define a function and assign it to an object like this:

let person = {

  firstName: 'John',

  lastName: 'Doe'

};

function greet() {

  console.log('Hello, World!');

}

person.greet = greet;

person.greet();

for...in  allows you to access each property and value of an object without knowing the specific name of the property

Write a Program To print all properties from person using for in loop?

var person = {

  firstName: 'John',

  lastName: 'Doe',

  ssn: '299-24-2351'

};

for(var prop in person) {           // All properties from person output= firsName: John, lastName:Doe , ssn:"299-24"//

  console.log(prop + ':' + person[prop]);

}

let person = {

  firstName: 'John',

  lastName: 'Doe',

  greet() {

      console.log('Hello, World!');

  }

};

person.greet();

Write a program to print all key of given objects using for loop?

let person = {

  gender : "male"

}

let person1 = Object.create(person);

person1.name = "Jon";

person1.age = 45;

person1.nationality = "Australian";

for (let key in person1) {

// Output : name, age, nationality

// and gender

  console.log(key);

}

In this example, we first create an object, and further we will be adding some properties to it.

let object = {

    name: "ABC",

    email: "xyz@abc.com"

  };

  console.log(object);

  // Adding some properties into the

  // above created object...

  object.age = 10;

  object.contact = 123467890;

  object.place = "Earth";

  object["country"]= "India";

  console.log(object);

A screenshot of a computer

Description automatically generated

In this example, we will first create an object and then remove some properties.

let object = {

    name: "ABC",

    email: "xyz@abc.com",

    age: 10,

    contact: 1234567890,

    place: "Earth",

    country: "India",

  };

  console.log(object);

  // Removing some properties

  // from the above created object

  delete object.age; // age property deletion

  console.log(object);

  // country property deletion

  delete object["country"];

  console.log(object);

  delete object.place; // place property deletion

  console.log(object);

A screenshot of a computer

Description automatically generated

In this example, we will try to use the same array of objects which we have created previously, and then we will use the [for-of loop](https://www.geeksforgeeks.org/javascript-syntaxerror-a-declaration-in-the-head-of-a-for-of-loop-cant-have-an-initializer/) in order to update the existing object’s property’s value.

  let employees\_data = [

    {

      employee\_id: 1,

      employee\_name: "Aman",

    },

    {

      employee\_id: 2,

      employee\_name: "Bhargava",

    },

    {

      employee\_id: 3,

      employee\_name: "Chaitanya",

    },

  ];

  for (let object of employees\_data) {

    if (object.employee\_id === 2) {

      object.employee\_name = "Anthony";

    }

  }

  console.log("Updated Data: ");

  console.log(employees\_data);

A screenshot of a computer

Description automatically generated

Write a Program using all object key, object value, object entities methods using given object?

//object.key(ob)

//object.values(obj)

//object.entries(obj)

//create an iterable object from the object//

let person ={

  firstName: "Pooja",

  lastName: "Savant",

  Education: "IT",

  hobby: "learning new skills",

  location: "Pune"

}

//for(let prop of arr)

// foreach() filter( ) map() reduce()

let keys = Object.keys(person);

console.log('Keys', keys);                 // Print keys of array

let vals= Object.values(person);

console.log('Vals', vals)                  // To print values of array

let entries = Object.entries(person);

console.log('Entries', entries)               // to print keys and values

console.log( entries[2][1])                 // gives the answer in form of row and column

**Add Methods to a Constructor Function Using Prototype**

You can also add new methods to a constructor function using prototype. For example,

// constructor function

function Person () {

    this.name = 'John',

    this.age = 23

}

// creating objects

const person1 = new Person();

const person2 = new Person();

// adding property to constructor function

Person.prototype.gender = 'male';

// prototype value of Person

console.log(Person.prototype);

// inheriting the property from prototype

console.log(person1.gender);

console.log(person2.gender);

## Changing Prototype

If a prototype value is changed, then all the new objects will have the changed property value. All the previously created objects will have the previous value

// constructor function

function Person() {

  this.name = 'John'

}

// add a property

Person.prototype.age = 20;

// creating an object

const person1 = new Person();

console.log(person1.age); // 20

// changing the property value of prototype

Person.prototype = { age: 50 }

// creating new object

const person3 = new Person();

console.log(person3.age); // 50

console.log(person1.age); // 20