Used to represent real world objects and in js it is more used to analyze more complex ones.

var arr=[2,2,3,4,5,6,1];

console.log(typeof arr); // knowing type of variable we have object type.

var art=67.1;

console.log(typeof art); // number

// more manipulations with objets

let car ={

     model : "A8",

    year : 2019,

    color : "Black",

    in\_Stock : true

};

var updatehere=car.color;

console.log(updatehere);

***Iterables :***

Iterables are also objects , which is used normally with for of loops.

Can be accessed with simple and efficient code.

Looping over a sequence of elements and wrote them one by one .

// iterables

for (const ton of arr) {

    console.log(ton);

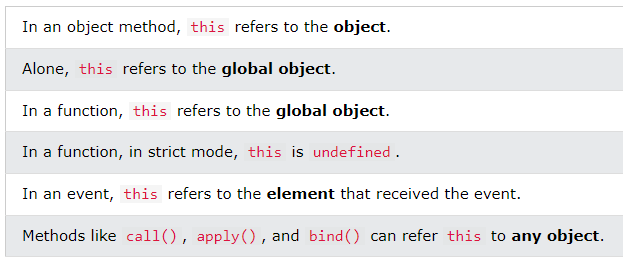
}

***This Keyword:***

In JavaScript, the this keyword refers to an **object**.

**Which** object depends on how this is being invoked (used or called).

The this keyword refers to different objects depending on how it is used:



// this is a keyword used with objects refers to the current used object.

// object binding

const person = {

  fname : "Ahmed",

  lastname : "Ali",

  id : 1402,

  my : function() {

    return this.fname + " " + this.lastname;

  }

};

Explicite function binding : getting a function into another function as an argument

const person1 = {

    fullName: function() {

      return this.firstName + " " + this.lastName;

    }

  }

  const person2 = {

    firstName:"John",

    lastName: "Doe",

  }

  // Return "John Doe":

  person1.fullName.call(person2);

example of binding another

// function binding another

  const per = {

    firstName:"John",

    lastName: "Doe",

    fullName: function () {

      return console.log(this.firstName + " " + this.lastName);

    }

  }

  const member = {

    firstName:"Hege",

    lastName: "Nilsen",

  }

  let fullName = per.fullName.bind(member);