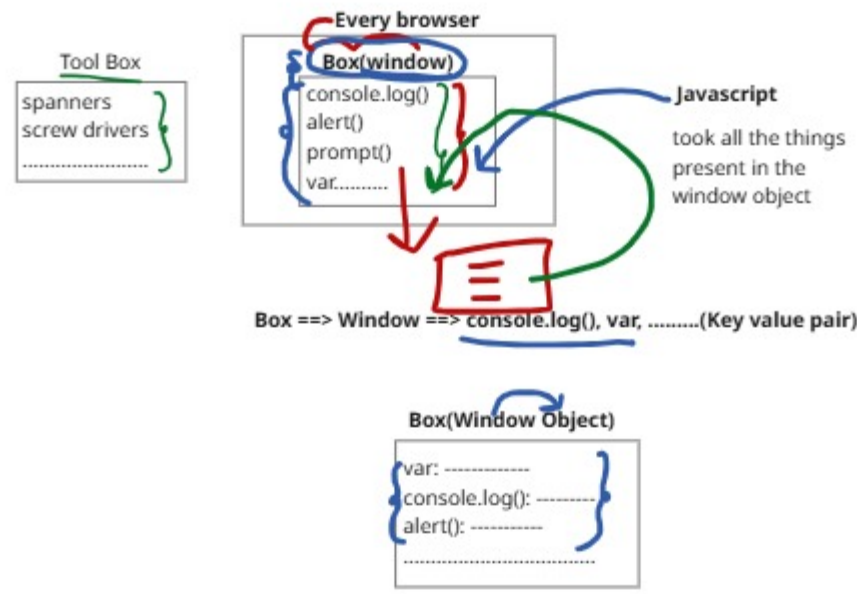


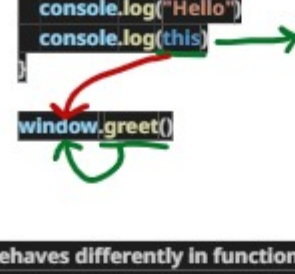
Array in Javascript ==> []
Object in Javascript
Object and Map ==> 99% same

Map ==> key value pairs ==> new Map() ==> { }
Object ==> Map and Object will have keys, in Map a key can be of any type, but on object a key should always be a string



Box ==> Window ==> console.log(), var,(Key value pair)

Box(Window Object)



this behaves differently in function expression and arrow function
In function expression this points to the object on which the function is called, but in arrow function this always POINTS to window object.

1 What will be the output?
const student = {
 name: "Raju",
 age: 20,
 show: () => {
 console.log(student.name);
 }
};

student.show();
a) Raju
b) undefined
c) Error
d) Nothing

Answer: a) Raju
✓ Because arrow function does not create its own this, but here you are using student.name.

2 What will be the output?
const student = {
 name: "Raju",
 show: () {
 console.log(this.name);
 }
};

student.show();
a) Raju
b) undefined
c) Error
d) null

Answer: a) Raju
✓ Normal function → this refers to student.

3 What will be the output?
const student = {
 city: "Mumbai",
 getCity: () => {
 console.log(this.city);
 }
};

student.getCity();
a) Mumbai
b) undefined
c) Error
d) null

Answer: b) undefined
✓ Arrow function → this = window/global, not student object.
✓ No city on window → undefined.

4 What gets printed?
const obj = {
 x: 10,
 y: 20,
 add: () {
 console.log(this.x + this.y);
 }
};

obj.add();
a) 1020
b) 10
c) 30
d) Error

Answer: c) 30

5 What is the output?
const data = {
 name: "Raju",
 print: () => {
 console.log(this.name);
 }
};

data.print();
a) Raju
b) undefined
c) Error
d) null

Answer: b) undefined
✓ this inside arrow function does NOT refer to object.

6 What is printed?
const user = {
 name: "Amit",
 getName: () {
 return this.name;
 }
};

console.log(user.getName());
a) Amit
b) undefined
c) getName
d) Error

Answer: a) Amit

7 What is the output?
const info = {
 a: 5,
 b: 3,
 calc: () => {
 console.log(info.a * info.b);
 }
};

info.calc();
a) 15
b) Error
c) undefined
d) NaN

Answer: a) 15
✓ Using object name directly inside arrow function is valid.

8 What will this code log?
const student = {
 name: "Raju",
 print: () {
 console.log(this);
 }
};

student.print();
a) Window object
b) undefined
c) The student object
d) null

Answer: c) The student object

9 What will be printed?
const student = {
 name: "Raju",
 printName: function () {
 console.log(this.name.toUpperCase());
 }
};

student.printName();
a) RAJU
b) rajju
c) undefined
d) Error

Answer: a) RAJU

10 What is the final output?
const person = {
 name: "Raju",
 age: 20,
 details: () {
 console.log(`\${this.name} is \${this.age} years old`);
 }
};

person.details();
a) Raju age 20
b) Raju is 20 years old
c) undefined
d) Error

Answer: b) Raju is 20 years old