‘THIS’ in JavaScript

**1.Global Context** : In the global execution context (outside of any function), this refers to the global object whether in strict mode or not.

console.log(this === window); // true

a = 37;

console.log(window.a); // 37

**2.Function Context :**Inside a function, the value of this depends on how the function is called.

(i)function f1() {

return this;

}

// In a browser:

f1() === window; // true

(ii)function f2() {

'use strict'; // see strict mode

return this;

}

f2() === undefined; // true

(iii) let car = {

Brand:'Honda'

getBrand: function () {

return this.brand;

}

Let bike = {

brand:’Harley Davidson

}

}

let brand = car.getBrand.bind(bike);

console.log(brand())

Sol : Harley Davidson

i.e when call the brand() method, the this keyword is bound to the car object.

**3.Arrow functions:** this retains the value of the enclosing lexical context's this. In global code, it will be set to the global object:

var globalObject = this;

var foo = (() => this);

console.log(foo() === globalObject); // true