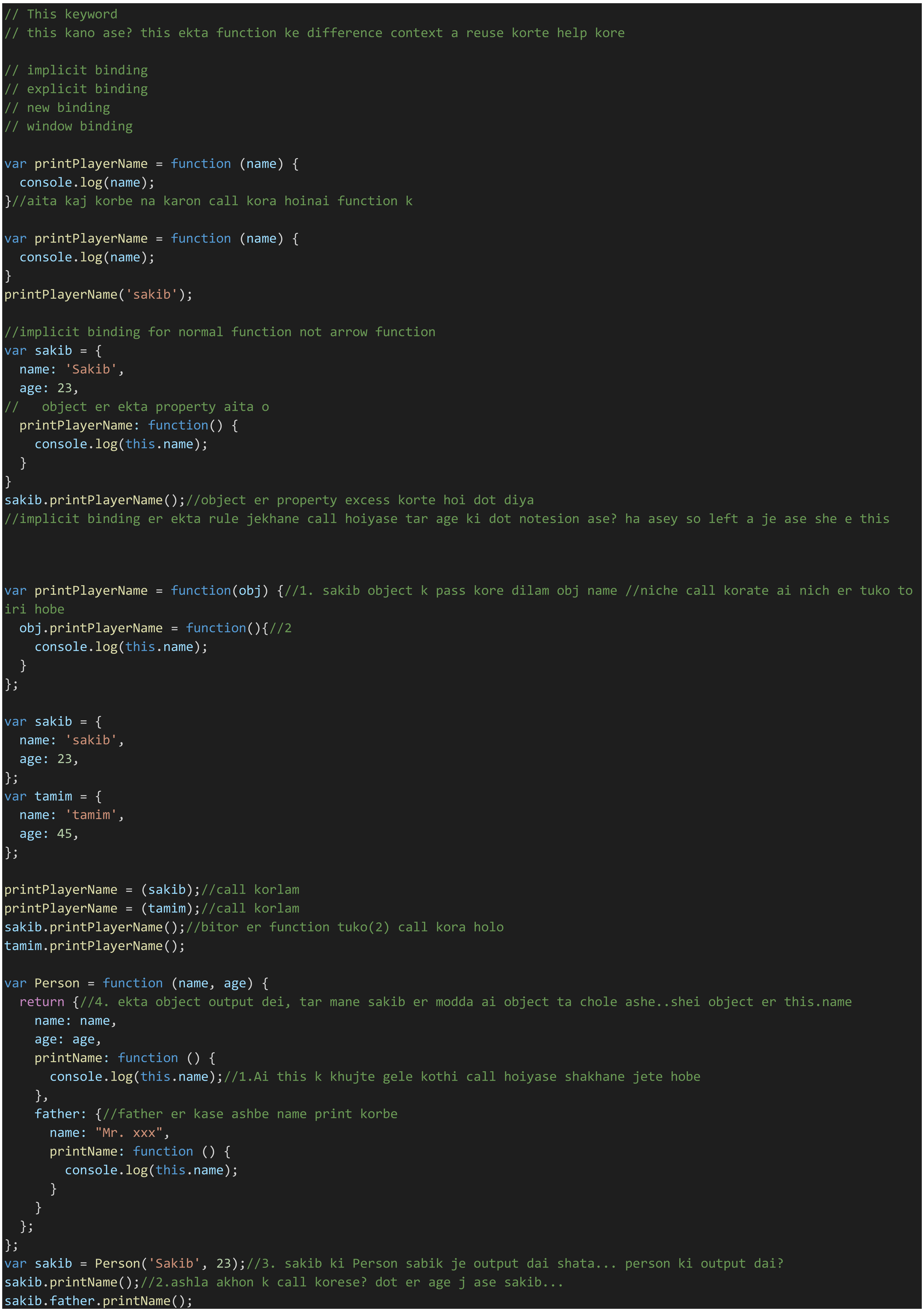
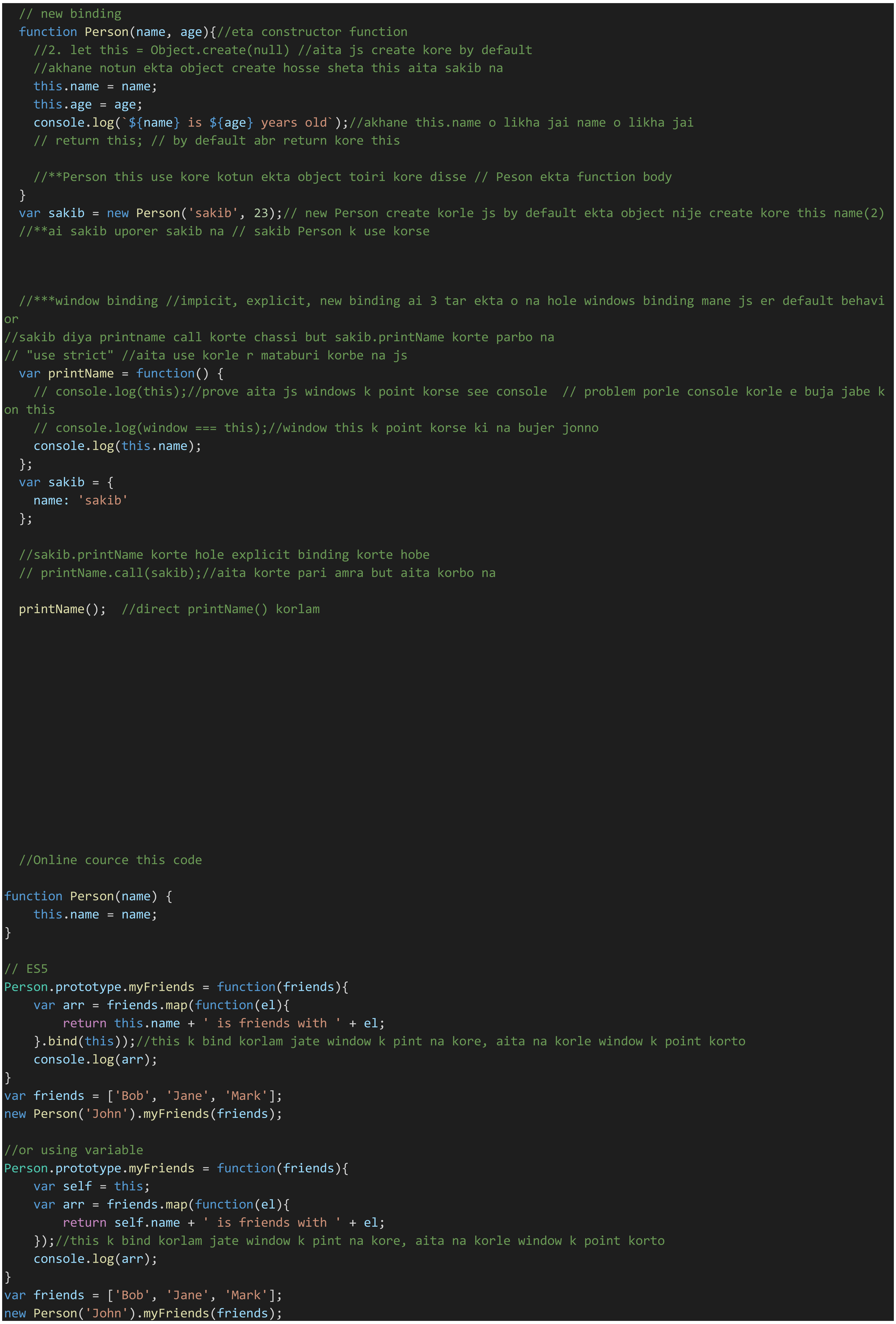
This Keyword



```
// This keyword
// this kano ase? this ekta function ke difference context a reuse korte help kore

// implicit binding
// explicit binding
// new binding
// window binding

var printPlayerName = function (name) {
  console.log(name);
}//aita kaj korbe na karon call kora hoinai function k

var printPlayerName = function (name) {
  console.log(name);
}
printPlayerName('sakib');

//implicit binding for normal function not arrow function
var sakib = {
  name: 'Sakib',
  age: 23,
//   object er ekta property aita o
  printPlayerName: function() {
    console.log(this.name);
  }
}
sakib.printPlayerName();//object er property excess korte hoi dot diya
//implicit binding er ekta rule jekhane call hoiyase tar age ki dot notesion ase? ha asey so left a je ase she e this



var printPlayerName = function(obj) {//1. sakib object k pass kore dilam obj name //niche call korate ai nich er tuko to
iri hobe
  obj.printPlayerName = function(){//2
    console.log(this.name);
  }
};

var sakib = {
  name: 'sakib',
  age: 23,
};
var tamim = {
  name: 'tamim',
  age: 45,
};

printPlayerName = (sakib);//call korlam
printPlayerName = (tamim);//call korlam
sakib.printPlayerName();//bitor er function tuko(2) call kora holo
tamim.printPlayerName();

var Person = function (name, age) {
  return {//4. ekta object output dei, tar mane sakib er modda ai object ta chole ashe..shei object er this.name
    name: name,
    age: age,
    printName: function () {
      console.log(this.name);//1.Ai this k khujte gele kothi call hoiyase shakhane jete hobe
    },
    father: {//father er kase ashbe name print korbe
      name: "Mr. xxx",
      printName: function () {
        console.log(this.name);
      }
    }
  };
};
var sakib = Person('Sakib', 23);//3. sakib ki Person sabik je output dai shata... person ki output dai?
sakib.printName();//2.ashla akhon k call korese? dot er age j ase sakib...
sakib.father.printName();
```

```
//Explicit binding
var sakib = {
  name: 'Sakib',
  age: 23,
  printName: function(){
    console.log(this.name);
  }
}
sakib.printName();

// printName functionke jodi baire niya ashi?
var printName = function(v1, v2, v3){
  console.log(`${this.name} is ${v1}, ${v2}, ${v3}`);
};//ata ekhon global scope shole ashese

var sakib = {
  name: 'Sakib',
  age: 23,
}

var v1 = 'Handsome';
var v2 = 'All- rounder';
var v3 = 'Best player';
printName.call(sakib, v1, v2, v3);//dot call 2 ta parameter pass kore, 2nd ta te onekgula o pass kora jabe // call ekta ekta kore parameter nai
//call() eta function k call kore // dot call 1st je parameter ta recive kore sheta ekta object, 2nd object unlimited jo to iccha
//ami bole dilam sakib k this hisebe use koro




// use apply with array
var printName = function(v1, v2, v3){
   console.log(`${this.name} is ${v1}, ${v2}, ${v3}`);
};

var sakib = {
  name: 'Sakib',
  age: 23,
}

var v1 = 'Handsome';
var v2 = 'All- rounder';
var v3 = 'Best player';

var v = [v1, v2, v3];// ekta array toiri korlam jate shob gula k eke bare pathate pari
printName.apply(sakib,v);


//dot bind dot call er moto
var printName = function(v1, v2, v3){
console.log(`${this.name} is ${v1}, ${v2}, ${v3}`);
 };
  var sakib = {
    name: 'Sakib',
    age: 23,
  }

  var v1 = 'Handsome';
  var v2 = 'All- rounder';
  var v3 = 'Best player';
  var newFunc = printName.bind(sakib, v1, v2, v3);//dot bind just function take direct call kore dai na ekta function er instance return kore mane oi function take e return kore dai
  //so printName k newFunc er modda store holo call hoinai
  newFunc();// newFunc k call korlam
```



```
  // new binding
  function Person(name, age){//eta constructor function
    //2. let this = Object.create(null) //aita js create kore by default
    //akhane notun ekta object create hosse sheta this aita sakib na
    this.name = name;
    this.age = age;
    console.log(`${name} is ${age} years old`);//akhane this.name o likha jai name o likha jai
    // return this; // by default abr return kore this

    //**Person this use kore kotun ekta object toiri kore disse // Peson ekta function body
  }
  var sakib = new Person('sakib', 23);// new Person create korle js by default ekta object nije create kore this name(2)
  //**ai sakib uporer sakib na // sakib Person k use korse



  //***window binding //impicit, explicit, new binding ai 3 tar ekta o na hole windows binding mane js er default behavi
or
//sakib diya printname call korte chassi but sakib.printName korte parbo na
// "use strict" //aita use korle r mataburi korbe na js
  var printName = function() {
    // console.log(this);//prove aita js windows k point korse see console  // problem porle console korle e buja jabe k
on this
    // console.log(window === this);//window this k point korse ki na bujer jonno
    console.log(this.name);
  };
  var sakib = {
    name: 'sakib'
  };

  //sakib.printName korte hole explicit binding korte hobe
  // printName.call(sakib);//aita korte pari amra but aita korbo na

  printName();  //direct printName() korlam















  //Online cource this code

function Person(name) {
    this.name = name;
}

// ES5
Person.prototype.myFriends = function(friends){
    var arr = friends.map(function(el){
        return this.name + ' is friends with ' + el;
    }.bind(this));//this k bind korlam jate window k pint na kore, aita na korle window k point korto
    console.log(arr);
}
var friends = ['Bob', 'Jane', 'Mark'];
new Person('John').myFriends(friends);

//or using variable
Person.prototype.myFriends = function(friends){
    var self = this;
    var arr = friends.map(function(el){
        return self.name + ' is friends with ' + el;
    });//this k bind korlam jate window k pint na kore, aita na korle window k point korto
    console.log(arr);
}
var friends = ['Bob', 'Jane', 'Mark'];
new Person('John').myFriends(friends);
```

```
//ES6
Person.prototype.myFriends = function(friends){
    var arr = friends.map(el => `${this.name} is friend with ${el}`);
    console.log(arr);
}

var friends = ['Bob', 'Jane', 'Mark'];
new Person('Mike').myFriends(friends);
```