# JS常用操作

1. 数据集合：  
   a)、创建一个集合：（可用于数组去重）  
    **let** arr = **new** Set(['张三', '露露', '丝丝', '露露']);  
    console.log(arr);  
     
   b)、一个属性：  
    console.log(arr.size);  
     
   c)、四个方法：  
    (1、add: （向集合中添加一个或多个值）  
    arr.add('刘德华').add('张韶涵');  
    console.log(arr);  
     
      
    (2、delete: (删除集合中某个值)  
    arr.delete('刘德华');  
    console.log(arr);  
     
      
    (3、has: (判断集合中是否存在某个值)  
    arr.has('张三');  
    arr.has('张韶涵');  
    console.log(arr);  
     
    (4、clear: (清空集合)  
    arr.clear();  
    console.log(arr);  
     
    (5、keys / values:  
    console.log(arr.keys());  
    console.log(arr.values());
2. 解构赋值：  
   数值型不具备解构赋值  
   a)、数组的解构赋值：  
    let [name, age, sex] = ['张韶涵', '30', '女'];  
    console.log(name);  
    console.log(age);  
    console.log(sex);  
     
   b)、对象的解构赋值：  
    let {name, age, sex} = {name: '张韶涵', age: '30', sex: '女'};  
    console.log(name);  
    console.log(age);  
    console.log(sex);  
     
   c)、复杂的解构赋值：  
    let {name, age, sex, pets, details} = {name: '张韶涵',   
    age: '30', sex: '女', pets: ['小筑', '佩奇', '小懒'],   
    details: {price: 200, number: 20}};  
    console.log(name);  
    console.log(age);  
    console.log(sex);  
    console.log(pets);  
    console.log(details);  
   复杂的数组解构赋值方法同上。  
     
   d)、有键值对的解构赋值：  
    let {name: text} = {name: '小筑'};  
    console.log(name); //获取到空值  
    console.log(text); //能拿到值  
     
   e)、字符串的解构赋值：  
    let [a, b, c, d, e] = '我来自四川';  
    console.log(a);  
    console.log(b);  
    console.log(c);  
    console.log(d);  
    console.log(e);
3. 数据集合 - map  
   a)、创建 map：  
    let obj1 = {nick: '二娃'}, obj2 = {nick: '七娃'};  
    const map = new Map([  
    ['name', '小筑'],  
    ['age', 20],  
    ['sex', '女'],  
    [obj1, '网易云课堂学习'],  
    [obj2, '巩固学习的是ES6基础'],  
    [['小七', '小狼'], '正在学习数据集合'],  
    ['name', '红汞'],  
    ['age', 888]  
    ]);  
    //注意，这里有去重功效但name和age会被替换掉  
    console.log(map);  
    console.log(map.size);  
     
   b)、set 和 get  
    map.set('pets', ['红毛', '黄毛']);  
    console.log(map.get(obj1));  
     
   c)、delete  
    map.delete(obj2);  
    console.log(map);  
     
   d)、has  
    console.log(map.has(obj1));  
    console.log(map.has(obj2));  
     
   e)、clear  
    map.clear();  
    console.log(map);  
     
   f)、keys() / values() / entries()  
    console.log(map.keys()); //获取所有的键  
    console.log(map.values()); //获取所有的值  
    console.log(map.entries()); //获取所有的键值  
     
   g)、遍历  
    map.forEach(function(value, index) {  
    console.log(value + '：' + index);  
    });  
      
      
     
   h)、注意事项  
    map.set({}, '哈哈哈');  
    map.set({}, '呵呵呵呵');  
    //以上两个都会被添加进去  
    console.log(map);  
    console.log({} === {}); //结果为false
4. Class的基本运用  
   a)、构造函数：（ES6之前的构造方法）  
   /\*   
    function Person(name, age) {  
    this.name = name;  
    this.age = age;  
    }  
    Person.prototype = {  
    constructor: Person,  
    print() {  
    console.log('我叫' + this.name + '，今年'  
    + this.age + '岁。');  
    }  
    }  
    let person = new Person('菲菲', 22);  
    console.log(person);  
   \*/  
     
   b)、通过class面向对：（ES6的构造方法）  
    class Person {  
    constructor(name, age) {  
    this.name = name;  
    this.age = age;  
    }  
    print() {  
    console.log('我叫' + this.name + '，今年'  
    + this.age + '岁。');  
    }  
    }  
    let person = new Person('菲菲', 22);  
    console.log(person);  
    person.print();