## 数组中是否存在某元素

## 数组中最大值

```
Array.prototype.max = function() {
var max = this[0];
var len = this.length;
for (var i = 1; i < len; i++){
if (this[i] > max) {
max = this[i];
}
}
return max;
}
Array.max = function( array ){
return Math.max.apply( Math, array );
};
```

## 数组中最小值

```
Array.prototype.min = function() {
var min = this[0];
var len = this.length;
for (var i = 1; i < len; i++){
if (this[i] < min){
min = this[i];
}
}
return min;
}
Array.min = function( array ){
return Math.min.apply( Math, array );
};
```

## 数组去重

```
var arr=[2,8,5,0,5,2,6,7,2];
function unique1(arr){
  var hash=[];
  for (var i = 0; i < arr.length; i++) {
     if(hash.indexOf(arr[i])==-1){
      hash.push(arr[i]);
     }
  }
```

```
  return hash;
}
```

## 数组对象对比

```
var result = arr1.reduce(function (v, k) {
        var filters = v.filter(function (data) {
            return data.mapProvince == k.mapProvince &&
data.mapCity == k.mapCity && data.mapDistrict == k.mapDistrict
        });
        console.log(filters)
        if (filters.length === 0) {
            v.push({
                mapProvince: k.mapProvince,
                mapCity: k.mapCity,
                mapDistrict: k.mapDistrict,
                shopName: [k.shopName]
            })
        } else {
            filters[0].shopName.push(k.shopName)
        };
        return v
    }, [])
    console.log(result)
```