# 添加图标显示

map.clearOverlays();   -----清除已有标记

1. **var** point = **new** BMap.Point(px, py);  -------经纬度
2. **var** marker = **new** BMap.Marker(point);  ------根据经纬度新建图标对象
3. map.addOverlay(marker);  --------地图添加图标对象
4. // 生成标记信息(table)
5. **var** content = "<table>";
6. content = content + "<tr><td> 设备编号：" + data\_a[i].id + "</td></tr>";
7. content = content + "<tr><td> 安装地点：" + data\_a[i].addr + "</td></tr>";
8. content = content + "<tr><td> 记录时间：" + data\_a[i].time + "</td></tr>";
9. content += "</table>";
10. // 捕获标记点击事件，并且显示信息
11. // 函数闭包，总是执行
12. (**function** () {
13. **var** infoWindow = **new** BMap.InfoWindow(content);
14. marker.addEventListener("click", **function** () {
15. **this**.openInfoWindow(infoWindow);
16. });
17. })()

# 添加右键菜单

1. // 添加右键菜单   ContextMenu
2. **var** contextMenu = **new** BMap.ContextMenu();
3. **var** txtMenuItem = [
4. {
5. text: "放大",
6. callback: **function** () { map.zoomIn() }
7. },
8. {
9. text: "缩小",
10. callback: **function** () { map.zoomOut() }
11. },
12. {
13. text: '查看北京',
14. callback: **function** () { map.centerAndZoom("北京") }
15. },
16. {
17. text: '放置到最大',
18. callback: **function** () { map.zoomTo(18) }
19. },
20. {
21. text: '获取改点坐标',
22. callback: **function**(p){
23. **var** px = p.lng;
24. **var** py = p.lat;
25. alert("该点坐标：\n经度：" + px + "; \n纬度：" + py);
26. }
27. },
28. {
29. text: '添加该店标注',
30. callback: **function** (p) {
31. **var** marker = **new** BMap.Marker(p), px = map.pointToPixel(p);
32. map.addOverlay(marker);
33. marker.enableDragging(**true**);
34. }
35. }
36. ];
37. // 遍历菜单items，添加进菜单
38. **for** (**var** i = 0; i < txtMenuItem.length; i++) {
39. contextMenu.addItem(**new** BMap.MenuItem(txtMenuItem[i].text, txtMenuItem[i].callback, 100));
40. **if** (i == 1 || i == 3) {
41. contextMenu.addSeparator();
42. }
43. }
44. map.addContextMenu(contextMenu);    // 添加菜单到map

# 给浏览器缓存一张图片

-------------建立图像对象：图像对象名称=new Image([宽度],[高度])

**var** img = **new** Image();-----建立图片对象

**var** imageUrl = '/mapv/img/building.png';

img.src = imageUrl;

img.onload事件会在 window.onload 之后触发。

尽量不要在 Image 对象里放置过多的图片

# 添加控件

1.NavigationControl：缩放地图的控件，默认在左上角；

2.OverviewMapControl：地图的缩略图的控件，默认在右下方；

3.ScaleControl：地图显示比例的控件，默认在左下方；

4.MapTypeControl：地图类型控件，默认在右上方；

var map = new BMap.Map("container");//在container容器中创建一个地图,参数container为div的id属性;

var point = new BMap.Point(500,500);//定位

map.centerAndZoom(point,15); //将point移到浏览器中心，并且地图大小调整为15;

map.addControl(new BMap.NavigationControl());

map.addControl(new BMap.MapTypeControl());

map.addControl(new BMap.ScaleControl());

map.addControl(new BMap.OverviewMapControl());

# 标注覆盖物

var marker = new BMap.Marker(point);

(2)添加覆盖物：map.addOverlay(marker);

(3)激活标注的拖拽功能：marker.enableDragging();

(4)为标注添加事件：marker.addEventListener("名称",function(){

    //点击标注后的事件

});

(5)删除覆盖物：map.removeOverlay(marker);

(6)销毁标注：marker.dispose();

# 创建对象

|  |
| --- |
| **var** map = **new** BMap.Map("container"); // 创建地图实例  **var** point = **new** BMap.Point(121.51, 31.24); // 创建点坐标 上海为中心    map.centerAndZoom(point, 15);  map.enableScrollWheelZoom(); |

## 设置控件偏移量

|  |
| --- |
| 始化控件时，可提供一个可选参数，参数类型为一个JavaScript对象。其中anchor和offsetX、offsetY  允许的值为：  BMAP\_ANCHOR\_TOP\_LEFT  BMAP\_ANCHOR\_TOP\_RIGHT  BMAP\_ANCHOR\_BOTTOM\_LEFT  BMAP\_ANCHOR\_BOTTOM\_RIGHT  var opts = {anchor: BMAP\_ANCHOR\_TOP\_RIGHT, offset: new BMap.Size(10, 10)}  map.addControl(new BMap.NavigationControl(opts)); |

## 控件

地图API中提供的控件有：

· Control：控件的抽象基类，所有控件均继承此类的方法、属性。通过此类您可实现自定义控件。

· NavigationControl：地图平移缩放控件，默认位于地图左上方，它包含控制地图的平移和缩放的功能。

· OverviewMapControl：缩略地图控件，默认位于地图右下方，是一个可折叠的缩略地图。

· ScaleControl：比例尺控件，默认位于地图左下方，显示地图的比例关系。

· CopyrightControl：版权控件，默认位于地图左下方

## 提供数据信息的接口

 LocalSearch：本地搜索，提供某一特定地区的位置搜索服务，比如在北京市搜索“公园”。

· TransitRoute：公交导航，提供某一特定地区的公交出行方案的搜索服务。

· DrivingRoute：驾车导航，提供驾车出行方案的搜索服务。

· WalkingRoute：步行导航，提供步行出行方案的搜索服务。

· Geocoder：地址解析，提供将地址信息转换为坐标点信息的服务。

· LocalCity：本地城市，提供自动判断您所在城市的服务。

· TrafficControl：实时路况控件，提供实时和历史路况信息服务。

## 地址解析

new BMap.Geocoder();

## 图层

var tilelayer = new BMap.TileLayer();

可以实现用户自定义图层功能

map.addTileLayer(tilelayer);       //将图层添加到地图上

## 添加覆盖物

map.addOverlay(goodPointCollection);