

Google Maps API 教程

在本教程中我们将学习如何使用谷歌地图API V3创建交互式地图。

[现在开始学习 Google Maps API ！](#)

什么是 API？

API = 应用程序编程接口（Application programming interface）。

API（Application Programming Interface，应用编程接口） 其实就是操作系统留给应用程序的一个调用接口，应用程序通过调用操作系统的 API 而使操作系统去执行应用程序的命令（动作）。

Google 地图API Key

开始学习本教程前，你需要拥有一个免费的 Google 地图 API key。

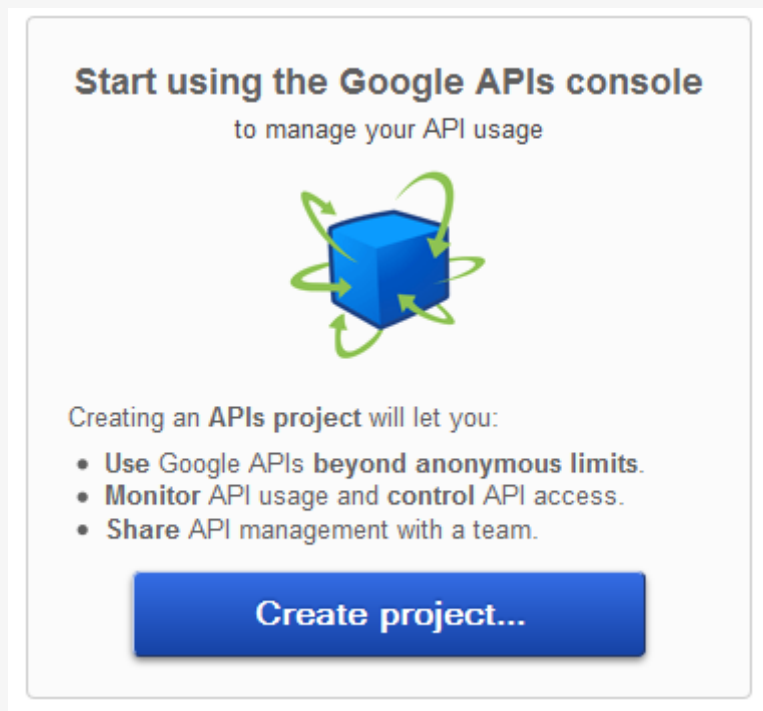
开始学习？

开始学习本教程前，你需要在Google上申请一个指定的API key。

通过以下步骤我们可以免费获取 API key 。

访问 <https://code.google.com/apis/console/>, 使用你的Google账号登陆。

登陆后会出现如下界面：



点击 "Create Project" 按钮。

在服务列表中找到 **Google Maps API v3**, 然后点击 "off" (关闭) 让其开启该服务器

在下一个步骤中，选择 "I Agree..." 然后点击 "Accept" 按钮。现在你在服务列表中应该就可以看到 Google Maps API v3 已经变为 "on" (开启) 状态。

接着在左侧菜单中点击 "API Access" ，在右侧栏中将看到以下提示 "Create an OAuth 2.0 client id..."。

点击 "Create an OAuth 2.0 client id..."，将弹出一个表单，表单需要你填入你的项目名称，项目图片或者logo，然后点击 "Next" 按钮。

然后，我们需要选择应用类型 ("Web application" : 网站应用), 然后填写你的站点地址，之后点击 "Create Client Id" 按钮即可。

最后我们就可以得到我们需要的 API key,如下图所示：

Simple API Access

Use API keys to identify your project when you do not need to access user data. [Learn more](#)

Key for browser apps (with referers)

API key: AIzaSyDY0kkJiTPVd2U7aTOAwhc9ySH6oHxOIYM

Referers: Any referer allowed

Activated on: Mar 20, 2012 5:46 AM

Activated by: support@w3schools.com – you

Create new Server key...

Create new Browser key...

💡注意：保存你的API key！（在填写的指定 URL 中开发所有的 Google 地图应用你需要使用该API key）。

← Google API 教程

Google 地图基础 →

✍ 点我分享笔记

Google Maps 基础

创建一个简单的 Google 地图

现在让我们创建一个简单的 Google 地图。

以下是显示了英国伦敦的 Google 地图：

实例

```
<!DOCTYPE html>
<html>
<head>
<script src="https://maps.googleapis.com/maps/api/js?key=AIzaSyDJW4jsPlNKgv6jFm3B5Edp5ywgdqLWdm
c&callback=initMap" async defer></script>
<script>
function initialize()
{
var mapProp = {
center:new google.maps.LatLng(51.508742,-0.120850),
zoom:5,
mapTypeId:google.maps.MapTypeId.ROADMAP
};
var map=new google.maps.Map(document.getElementById("googleMap"), mapProp);
}
google.maps.event.addDomListener(window, 'load', initialize);
</script>
</head>
<body>
<div id="googleMap" style="width:500px;height:380px;"></div>
</body>
</html>
```

[尝试一下 »](#)

实例解析

我们以以上实例来解析 Google 地图的创建过程。

应用为什么要声明 HTML5?

```
<!DOCTYPE html>
```

大多数浏览器使用 "标准模式" 的 HTML5 文档渲染页面，这就意味着你的应用是兼容各大浏览器的。

另外，如果没有DOCTYPE标签，浏览器则使用混杂模式 (quirks mode)进行渲染页面内容。

提示：应该注意的是一些"混杂模式"中的CSS并不能使用于标准模式中。在具体的应用中，所有基于百分比的大小都必须从父块元素继承。如果在父模块中没有指定大小，默认值为 0 x 0 像素。如果你想使用百分比，可以在<style> 标签中声明，如下所

示：

```
<style type="text/css">
html {height:100%}
body {height:100%;margin:0;padding:0}
#googleMap {height:100%}
</style>
```

这个样式声明表明地图模块的 (GoogleMap) 应 HTML高度为100%。

添加 Google 地图 API Key

在以下实例中第一个<script> 标签中必须包含 Google 地图 API：

```
<script src="http://maps.googleapis.com/maps/api/js?
key=YOUR_API_KEY&sensor=TRUE_OR_FALSE"></script>
```

将google生成的 API key 放置于 **key** 参数中(key=YOUR_API_KEY)。

The **sensor** 参数是必须的，该参数用于指明应用程序是否使用一个传感器 (类似 GPS 导航) 来定位用户的位置。参数值可以设置为 true 或者 false。

HTTPS

如果你的应用是安全的HTTP(HTTPS:HTTP Secure)应用,你可以使用 HTTPS 来加载 Google 地图 API：

```
<script src="https://maps.googleapis.com/maps/api/js?
key=YOUR_API_KEY&sensor=TRUE_OR_FALSE"></script>
```

异步加载

同样我们也可以在页面完全载入后再加载 Google 地图 API。

以下实例使用了 window.onload 来实现页面完全载入后加载 Google 地图。loadScript() 函数创建了加载 Google 地图 API <script> 标签。此外在标签的末尾添加了 callback=initialize 参数，initialize()作为回调函数会在API完全载入后执行：

实例

```
function loadScript()
{
var script = document.createElement("script");
script.type = "text/javascript";
script.src = "https://maps.googleapis.com/maps/api/js?key=AIzaSyBzE9xAEsye6Kde-3hT-6B90nfwUkcS8Yw&sensor=false&callback=initialize";
document.body.appendChild(script);
}
window.onload = loadScript;
```

尝试一下 »

定义地图属性

在初始化地图前，我们需要先创建一个 Map 属性对象来定义一些地图的属性：

```
var mapProp = {
  center:new google.maps.LatLng(51.508742,-0.120850),
  zoom:7,
  mapTypeId: google.maps.MapTypeId.ROADMAP
};
```

center (中心点)

center 属性指定了地图的中心，该中心通过坐标（纬度，经度）在地图上创建一个中心点。

Zoom (缩放级数)

zoom 属性指定了地图的 缩放级数。zoom: 0 显示了整个地球地图的完全缩放。

MapTypeId (地图的初始类型)

mapTypeId 属性指定了地图的初始类型。

mapTypeId包括如下四种类型：

- google.maps.MapTypeId.HYBRID：显示卫星图像的主要街道透明层
- google.maps.MapTypeId.ROADMAP：显示普通的街道地图
- google.maps.MapTypeId.SATELLITE：显示卫星图像
- google.maps.MapTypeId.TERRAIN：显示带有自然特征（如地形和植被）的地图

在哪里显示 Google 地图

通常 Google 地图使用于 <div> 元素中：

```
<div id="googleMap" style="width:500px;height:380px;"></div>
```

注意：地图将以div中设置的大小来显示地图的大小，所以我们可以 在 <div> 元素中设置地图的大小。

创建一个 Map 对象

```
var map=new google.maps.Map(document.getElementById("googleMap"),mapProp);
```

以上代码使用参数(mapProp)在<div> 元素 (id为googleMap) 创建了一个新的地图。

提示：如果想在页面中创建多个地图，你只需要添加新的地图对象即可。

以下实例定义了四个地图实例 (四个地图使用了不同的地图类型):

实例

```
var map = new google.maps.Map(document.getElementById("googleMap"),mapProp);
var map2 = new google.maps.Map(document.getElementById("googleMap2"),mapProp2);
var map3 = new google.maps.Map(document.getElementById("googleMap3"),mapProp3);
var map4 = new google.maps.Map(document.getElementById("googleMap4"),mapProp4);
```

尝试一下 »

加载地图

窗口载入后通过执行 initialize() 函数来初始化 Map 对象，这样可以确保在页面完全载入后再加载 Google 地图：

```
google.maps.event.addDomListener(window, 'load', initialize);
```

 点我分享笔记

Google 地图

在Google地图中添加一个标记

Google 地图 - 叠加层

叠加层是地图上绑定到经度/纬度坐标的对象，会随您拖动或缩放地图而移动。

Google 地图 API 有如下几种叠加层：

- 地图上的点使用标记来显示，通常显示自定义图标。标记是 `GMarker` 类型的对象，并且可以利用 `GIcon` 类型的对象来自定义图标。
- 地图上的线使用折线（表示点的集合）来显示。线是类型为 `GPolyline` 的对象。
- 地图上的区域显示为多边形（如果是任意形状的区域）或底面叠加层（如果是矩形区域）。多边形类似于闭合的折线，因此可以是任何形状。地面叠加层通常用于地图上与图块有直接或间接关联的区域。
- 地图本身使用图块叠加层显示。如果您有自己的系列图块，可以使用 `GTileLayerOverlay` 类来改变地图上已有的图块，甚至可以使用 `GMapType` 来创建您自己的地图类型。
- 信息窗口也是一种特殊的叠加层。但是请注意，信息窗口会自动添加到地图中，并且地图只能添加一个类型为 `GInfoWindow` 的对象。

Google 地图 - 添加标记

标记标识地图上的点。默认情况下，它们使用 `G_DEFAULT_ICON`（您也可以指定自定义图标）。`GMarker` 构造函数将 `GLatLng` 和 `GMarkerOptions`（可选）对象作为参数。

标记设计为可交互。例如，默认情况下它们接收 "click" 事件，常用于在事件侦听器中打开信息窗口。

通过 setMap() 方法在地图上添加标记:

实例

```
var marker=new google.maps.Marker({
  position:myCenter,
});
marker.setMap(map);
```

尝试一下 »

Google 地图 - 可拖动的标记

以下实例将介绍如何使用 animation 属性来拖动标记:

实例

```
marker=new google.maps.Marker({
  position:myCenter,
  animation:google.maps.Animation.BOUNCE
});
marker.setMap(map);
```

尝试一下 »

Google 地图 - 图标

标记可以用自定义的新图标来显示，以替代默认图标:

实例

```
var marker=new google.maps.Marker({
  position:myCenter,
  icon:'pinkball.png'
});
marker.setMap(map);
```

尝试一下 »

Google 地图 - 折线

GPolyline 对象可在地图上创建线性叠加层。GPolyline 包括一系列点，并创建一系列有序连接这些点的线段。

折线支持以下属性：

- path - 指定了多个直线的纬度/经度坐标
- strokeColor - 指定直线的十六进制颜色值(格式: "#FFFFFF")
- strokeOpacity - 指定直线的透明度(该值为 0.0 到 1.0)

- strokeWeight - 定义线的宽度,以像素为单位。
- editable - 定义用户是否可编辑直线(true/false)

实例

```
var myTrip = [stavanger, amsterdam, london];
var flightPath = new google.maps.Polyline({
  path:myTrip,
  strokeColor:"#0000FF",
  strokeOpacity:0.8,
  strokeWeight:2
});
```

[尝试一下 »](#)

Google 地图 - 多边形

GPolygon 对象类似于 GPolyline 对象，因为它们都包括一系列有序的点。但是，多边形不像折线一样有两个端点，而是设计为定义形成闭环的区域。

和折线一样，您可以自定义多边形边（线）的颜色、粗细和透明度，以及封闭的填充区域的颜色和透明度。颜色应是十六进制数字 HTML 样式。

多边形支持以下属性：

- path - 指定多个直线纬度的坐标 (第一个和最后一个坐标是相等的)
- strokeColor - 指定直线的十六进制颜色值(格式: "#FFFFFF")
- strokeOpacity -指定直线的透明度(该值为 0.0 到 1.0)
- strokeWeight - 定义线的宽度,以像素为单位。
- fillColor - 指定闭合区域的十六进制颜色值 (格式: "#FFFFFF")
- fillOpacity - 指定填充颜色的透明度 (该值为 0.0 到 1.0)
- editable - 定义用户是否可编辑直线(true/false)

实例

```
var myTrip = [stavanger, amsterdam, london, stavanger];
var flightPath = new google.maps.Polygon({
  path:myTrip,
  strokeColor:"#0000FF",
  strokeOpacity:0.8,
  strokeWeight:2,
  fillColor:"#0000FF",
  fillOpacity:0.4
});
```

[尝试一下 »](#)

Google 地图 - 圆

圆支持以下属性：

- center - 指定圆的中心点参数 google.maps.LatLng
- radius - 指定圆的半径，以米为单位
- strokeColor - 指定弧线的十六进制颜色值(格式: "#FFFFFF")
- strokeOpacity - 指定弧线的透明度(该值为 0.0 到 1.0)
- strokeWeight - 定义线的宽度,以像素为单位。
- fillColor - 指定圆的十六进制颜色值填充值 (格式: "#FFFFFF")
- fillOpacity - 指定填充颜色的透明度 (该值为 0.0 到 1.0)
- 定义用户是否可编辑直线(true/false)

实例

```
var myCity = new google.maps.Circle({
  center: amsterdam,
  radius: 20000,
  strokeColor: "#0000FF",
  strokeOpacity: 0.8,
  strokeWeight: 2,
  fillColor: "#0000FF",
  fillOpacity: 0.4
});
```

尝试一下 »

Google 地图 - 信息窗口

在一个标记上显示一个文本信息窗口：

实例

```
var infowindow = new google.maps.InfoWindow({
  content: "Hello World!"
});
infowindow.open(map, marker);
```

尝试一下 »

Google 地图 - 叠加层参考手册

[Google Maps API 参考手册.](#)

[← Google 地图基础](#)

[Google 地图事件 →](#)

[点我分享笔记](#)

Google 地图事件

点击标记缩放地图 - 绑定在google地图上的事件。

点击标记缩放地图

我们仍然使用上一篇文章使用的英国伦敦的地图。

点用户点击标记时实现缩放地图的功能(点击标记时绑定地图缩放事件)。

代码如下：

实例

```
// Zoom to 9 when clicking on marker
google.maps.event.addListener(marker, 'click', function() {
  map.setZoom(9);
  map.setCenter(marker.getPosition());
});
```

[尝试一下 »](#)

使用 `addListener()` 事件处理程序来注册事件的监听。该方法使用一个对象，一个事件来监听，当指定的事件发生时 函数将被调用。

重置标记

我们通过给地图添加事件处理程序来改变 'center' 属性，以下代码使用 `center_changed` 事件在3秒后标记移会中心点：

实例

```
google.maps.event.addListener(map, 'center_changed', function() {  
  window.setTimeout(function() {  
    map.panTo(marker.getPosition());  
  }, 3000);  
});
```

[尝试一下 »](#)

点击标记时打开信息窗口。

点击标记在信息窗口显示一些文本信息：

实例

```
var infowindow = new google.maps.InfoWindow({  
  content: "Hello World!"  
});  
google.maps.event.addListener(marker, 'click', function() {  
  infowindow.open(map, marker);  
});
```

[尝试一下 »](#)

设置标记及打开每个标记的信息窗口

当用户点击地图时执行一个窗口

用户点击地图某个位置时使用 placeMarker() 函数在指定位置上放置一个标记，并弹出信息窗口：

实例

```
google.maps.event.addListener(map, 'click', function(event) {  
  placeMarker(event.latLng);  
});  
function placeMarker(location) {  
  var marker = new google.maps.Marker({  
    position: location,  
    map: map,  
  });  
  var infowindow = new google.maps.InfoWindow({  
    content: 'Latitude: ' + location.lat() +  
    '<br>Longitude: ' + location.lng()  
  });  
  infowindow.open(map, marker);  
}
```

[尝试一下 »](#)

Google 地图 - 事件参考手册

[Google Maps API 参考手册](#)。

[← Google 地图叠加层](#)

[Google 地图控件集 →](#)

[✎ 点我分享笔记](#)

Google 地图控件集

一个Google 地图 - 默认控件集设置：

Google 地图 - 默认控件集设置：

当使用一个标准的google地图，它的默认设置如下：

- .Zoom-显示一个滑动条来控制map的Zoom级别
- .PPan-地图上显示的是一个平底碗样的控件，点击4个角平移地图
- .MapType-允许用户在map types(roadmap 和 satallite)之间切换
- .StreetView-显示为一个街景小人图标，可拖拽到地图上某个点来打开街景

Google 地图 - 更多控件集

除了以上默认控件集,Google还有：

- .Scale-显示地图比例尺
- .Rotate-显示一个小的圆周图标，可以转动地图
- .verview Map-从一个广域的视角俯视地图

创建地图时你可以通过设置选项指定哪些控件集显示或者通过调用 `setOptions()` 来改变地图的设置选项。

Google 地图 - 关闭默认控件集

你可能希望能够关闭默认的控件集。

为了关闭默认控件集,设置地图的disableDefaultUI的属性为true:

实例

```
disableDefaultUI:true
```

[尝试一下 »](#)

Google 地图 - 开所有控件集

一些控件集默认显示在地图上,而其它的不会,除非你设置它们。

设置控件为true使其可见-设置控件为false则隐藏它。

以下实例开启所有的控件:

实例

```
panControl:true,  
zoomControl:true,  
mapTypeControl:true,  
scaleControl:true,  
streetViewControl:true,  
overviewMapControl:true,  
rotateControl:true
```

[尝试一下 »](#)

Google 地图 - 修改控件集

某些地图控件是可配置的。通过制定控件选项域改变控件集。

举个例子来说,修改Zoom 控件的选项在zoomControlOptions指定。zoomControlOptions包含如下3种选项:

- .google.maps.ZoomControlStyle.SMALL-显示最小化zoom 控件
- .google.maps.ZoomControlStyle.LARGE-显示标准zoom滑动控件
- .google.maps.ZoomControlStyle.DEFAULT-基于设备和地图大小,选择最合适的控件

实例

```
zoomControl:true,  
zoomControlOptions: {  
  style:google.maps.ZoomControlStyle.SMALL  
}
```

[尝试一下 »](#)

注意： 如果需要修改一个控件，首先开启控件(设置其为true)。

另一个控件为 MapType 控件。

MapType 控件可显示为以下 style 选项之一：

- google.maps.MapTypeControlStyle.HORIZONTAL_BAR，用于在水平栏中将一组控件显示为如 Google Maps 中所示按钮。
- google.maps.MapTypeControlStyle.DROPDOWN_MENU，用于显示单个按钮控件，以便您从下拉菜单中选择地图类型。
- google.maps.MapTypeControlStyle.DEFAULT，用于显示"默认"的行为，该行为取决于屏幕尺寸，并且可能会在 API 以后的版本中有所变化。

实例

```
mapTypeControl:true,  
mapTypeControlOptions: {  
  style:google.maps.MapTypeControlStyle.DROPDOWN_MENU  
}
```

尝试一下 »

同样你可以使用ControlPosition属性指定控件的位置:

实例

```
mapTypeControl:true,  
mapTypeControlOptions: {  
  style:google.maps.MapTypeControlStyle.DROPDOWN_MENU,  
  position:google.maps.ControlPosition.TOP_CENTER  
}
```

尝试一下 »

Google 地图 - 自定义控件集

创建一个返回伦敦自定义控件，用于点击事件：(如果地图被拖拽):

实例

```
controlDiv.style.padding = '5px';  
var controlUI = document.createElement('div');  
controlUI.style.backgroundColor = 'yellow';  
controlUI.style.border='1px solid';  
controlUI.style.cursor = 'pointer';  
controlUI.style.textAlign = 'center';  
controlUI.title = 'Set map to London';  
controlDiv.appendChild(controlUI);  
var controlText = document.createElement('div');  
controlText.style.fontFamily='Arial,sans-serif';  
controlText.style.fontSize='12px';  
controlText.style.paddingLeft = '4px';
```

```
controlText.style.paddingRight = '4px';
controlText.innerHTML = '<b>Home<b>'
controlUI.appendChild(controlText);
```

尝试一下 »

Google 地图 - 控件集参考手册

[Google Maps API 参考手册](#).

← Google 地图事件

Google 地图类型 →

 点我分享笔记

Google 地图类型

HYBRID类型的google地图:

Google 地图- 基本地图类型

Google Maps API 中提供了以下地图类型：

- `MapTypeId.ROADMAP`，用于显示默认的道路地图视图
- `MapTypeId.SATELLITE`，用于显示 Google 地球卫星图片
- `MapTypeId.HYBRID`，用于同时显示普通视图和卫星视图
- `MapTypeId.TERRAIN`，用于根据地形信息显示实际地图。

要通过 Map 修改正在使用的地图类型，您可以为其设置 `mapTypeId` 属性:

```
var mapProp = {  
  center:new google.maps.LatLng(51.508742,-0.120850),  
  zoom:7,  
  mapTypeId: google.maps.MapTypeId.HYBRID  
};
```

或者动态修改 `mapTypeId`：

```
map.setMapTypeId(google.maps.MapTypeId.HYBRID);
```

Google 地图- 45° 图像

Google Maps API 针对特定位置支持特殊的 45° 图像。

此类高分辨率图像可提供朝向各基本方向（东南西北）的透视视图。对于支持的地图类型，这些图片还可提供更高的缩放级别。

现有的 `google.maps.MapTypeId.SATELLITE` 和 `google.maps.MapTypeId.HYBRID` 地图类型支持高缩放级别的 45° 透视图像（如果有的话）。如果您放大的位置拥有此类图像，那么这些地图类型将会自动通过以下方式更改其视图：

- 地图上现有的任何平移控件都将会变更为在现有的导航控件周围添加一个罗盘转轮。您可以通过该罗盘来更改任意 45° 图像的方向，方法是：拖动该罗盘转轮，然后将方向对准包含图像的最近支持方向。
- 一个旋转控件将会间隙显示在现有的平移和缩放控件之间，它可用于将图像围绕支持方向旋转。旋转控件仅支持顺时针方向旋转。
- 以当前位置为中心的 45° 透视图像将会替代卫星图像或混合图像。默认情况下，此类视图会朝向北方。如果您缩小地图，则地图会重新显示默认的卫星图像或混合图像。
- MapType 控件将启用子菜单切换控件，用于显示 45° 图像。

注意：缩小显示 45° 图像的地图类型将会还原所有更改，并重新构建原始地图类型。

以下示例显示了意大利威尼斯公爵宫 45° 视图：

实例

```
var mapProp = {  
  center:myCenter,  
  zoom:18,  
  mapTypeId:google.maps.MapTypeId.HYBRID  
};
```

尝试一下 »

提示：Google 正在不断地为更多城市添加 45° 图像。有关最新信息，请参阅 [Google 地图上的 45° 图像列表](#)。

Google 地图 - 启用和停用 45° 图像 - setTilt(0)

您可以通过在 Map 对象上调用 `setTilt(0)` 来停用 45° 图像。要启用适用于支持的地图类型的 45° 透视图像，请调用 `setTilt(45)`。

实例

```
map.setTilt(0);
```

尝试一下 »

Google 地图 - 参考手册

[Google 地图 API 参考手册](#)。

← Google 地图控件集

Google 地图参考手册 →

 点我分享笔记

Google 地图 API 参考手册

Google 地图API 参考手册

地图

构造函数/对象	描述
Map()	在指定的 HTML 容器中创建新的地图，该容器通常是一个DIV元素。

叠加层

构造函数/对象	描述
Marker	创建一个标记。
MarkerOptions	标记的选项。由DirectionsRenderer渲染的所有标记都将使用这些选项。
MarkerImage	A structure representing a Marker icon or shadow image
MarkerShape	Defines the marker shape to use in determination of a marker's clickable region (type and coord)
Animation	Specifies animations that can be played on a marker (bounce or drop)
InfoWindow	Creates an info window
InfoWindowOptions	Options for rendering the info window
Polyline	Creates a polyline (contains path and stroke styles)
PolylineOptions	Options for rendering the polyline
Polygon	Creates a polygon (contains path and stroke+fill styles)
PolygonOptions	Options for rendering the polygon
Rectangle	Creates a rectangle (contains bounds and stroke+fill styles)
RectangleOptions	Options for rendering the rectangle
Circle	Creates a circle (contains center+radius and stroke+fill styles)
CircleOptions	Options for rendering the circle

GroundOverlay	
GroundOverlayOptions	
OverlayView	
MapPanes	
MapCanvasProjection	

事件

构造函数/对象	描述
MapsEventListener	It has no methods and no constructor. Its instances are returned from addListener(), addDomListener() and are eventually passed back to removeListener()
event	Adds/Removes/Trigger event listeners
MouseEvent	Returned from various mouse events on the map and overlays

控件集

构造函数/对象	描述
MapTypeControlOptions	Holds options for modifying a control (position and style)
MapTypeControlStyle	Specifies what kind of map control to display (Drop-down menu or buttons)
OverviewMapControlOptions	Options for rendering of the overview map control (opened or collapsed)
PanControlOptions	Options for rendering of the pan control (position)
RotateControlOptions	Options for rendering of the rotate control (position)
ScaleControlOptions	Options for rendering of the scale control (position and style)
ScaleControlStyle	Specifies what kind of scale control to display
StreetViewControlOptions	Options for rendering of the street view pegman control (position)
ZoomControlOptions	Options for rendering of the zoom control (position and style)
ZoomControlStyle	Specifies what kind of zoom control to display (large or small)
ControlPosition	Specifies the placement of controls on the map

 点我分享笔记

地图 API Map() 构造器

实例

创建一个 Google 地图:

```
var map=new google.maps.Map(document.getElementById("googleMap"),mapOpt);
```

尝试一下 »

定义和用法

Map() 构造器创建了一个新的地图并插入到指定的HTML元素中 (<div> 元素)。

语法

```
new google.maps.Map (HTMLElement, MapOptions)
```

参数值

参数	描述
HTMLElement	规定要把地图放置在那个 HTML 元素中。
MapOptions	带有地图初始化变量/选项的 MapOptions 对象。

Map() 的方法

方法	返回值	描述
fitBounds(LatLngBounds)	None	设置要包含给定边界的视口。
getBounds()	LatLng,LatLng	返回当前视口的西南纬度/经度和东北纬度/经度。
getCenter()	LatLng	返回地图的中心的纬度/经度。
getDiv()	Node	返回包含地图的 DOM 对象。
getHeading()	number	返回航拍图像的罗盘航向 (支持 SATELLITE 和 HYBRID 地图类型) 。
getMapTypeId()	HYBRID ROADMAP SATELLITE TERRAIN	返回当前地图类型。
getProjection()	Projection	返回当前 Projection (投影) 。

getStreetView()	<i>StreetViewPanorama</i>	返回绑定到地图的默认的 StreetViewPanorama。
getTilt()	<i>number</i>	返回航拍图像的入射角度数（支持 SATELLITE 和 HYBRID 地图类型）。
<u>getZoom()</u>	<i>number</i>	返回地图的当前缩放级别。
panBy(<i>xnumber,ynumber</i>)	None	通过以像素计的给定距离改变地图的中心。
panTo(<i>LatLng</i>)	None	改变地图的中心为给定的 LatLng。
panToBounds(<i>LatLngBounds</i>)	None	将地图平移所需的最小距离以包含给定的 LatLngBounds。
setCenter(<i>LatLng</i>)	None	
setHeading(<i>number</i>)	None	设置航拍图像的罗盘方向（以度为单位进行测量），基本方向为北方。
<u>setMapTypeId(<i>MapTypeId</i>)</u>	None	改变要显示的地图类型。
setOptions(<i>MapOptions</i>)	None	
setStreetView(<i>StreetViewPanorama</i>)	None	绑定一个 StreetViewPanorama 到地图上。
setTilt(<i>number</i>)	None	设置航拍图像的入射角度数（支持 SATELLITE 和 HYBRID 地图类型）。
setZoom(<i>number</i>)	None	

Map() 的属性

属性	类型	描述
controls	<i>Array.<MVCArray.<Node>></i>	要附加到地图上的额外控件。
mapTypes	<i>MapTypeRegistry</i>	按字符串 ID 划分的 MapType 实例的注册表。
overlayMapTypes	<i>MVCArray.<MapType></i>	要叠加的额外地图类型。

Map() 的事件

事件	参数	描述
bounds_changed	None	当可视区域范围更改时会触发此事件。
center_changed	None	当地图 center（中心）属性更改时会触发此事件。

click	<i>MouseEvent</i>	当用户点击地图（但不是点击标记或信息窗口）时会触发此事件。
dblclick	<i>MouseEvent</i>	当用户双击地图时会触发此事件。请注意，触发此事件前还会触发点击事件。
drag	None	当用户拖动地图时会反复触发此事件。
dragend	None	当用户停止拖动地图时会触发此事件。
dragstart	None	当用户开始拖动地图时会触发此事件。
heading_changed	None	当地图 heading（方向）属性更改时会触发此事件。
idle	None	当地图在平移或缩放之后变为闲置状态时会触发此事件。
maptypeid_changed	None	当 mapTypeId 属性更改时会触发此事件。
mousemove	<i>MouseEvent</i>	只要用户的鼠标在地图容器上移动，就会触发此事件。
mouseout	<i>MouseEvent</i>	当用户的鼠标从地图容器上退出时会触发此事件。
mouseover	<i>MouseEvent</i>	当用户的鼠标进入地图容器时会触发此事件。
projection_changed	None	当投影更改时会触发此事件。
resize	None	当地图（div）更改尺寸时会触发此事件。
rightclick	<i>MouseEvent</i>	当用户右击地图时会触发此事件。
tilesloaded	None	当可见图块载入完成后会触发此事件。
tilt_changed	None	当地图 tilt（倾斜）属性更改时会触发此事件。
zoom_changed	None	当地图 zoom（缩放）属性更改时会触发此事件。

