**openlayer加载本地svg**

**1.组织图层**

*/\*\**

*\*组*织*图层*

*\**@*param*{*\*geojson格式的数据} geojsonObject*

*\**@*param*{*\*图层类型} type*

*\**

*\**@*return*s*图层*

*\*/*

*exp*o*rt fun*c*tion organizeLayer(geojsonObject, type)* *{*

 *if (ty*p*e == "Point") {*

 *//* *数据*

 *va*r*vectorSource = new OlSourceVector({*

  *features: new OlGeoJSON().readFeatures*(*geojsonObject),*

  *attributions: "vectorSource-equipment"*

 *})*;

 *//* *图层样式*

 *va*r*styleFunction = function(feature) {*

  *return organizeStyleimg(*

  *feature.values\_.typeId,*

  *feature.values\_.deviceName,*

  *feature.values\_.status*

  *);*

 *};*

 *//* *图层*

 *va*r*vectorLayer = new OlLayerVector({*

  *source: vectorSource,*

  *name: "equipment-source",*

  *style: styleFunction*

 *})*;

 *} else* *if (type == "LineString") {*

 *//* *数据*

 *va*r*vectorSource = new OlSourceVector({*

  *features: new OlGeoJSON().readFeatures*(*geojsonObject),*

  *attributions: "vectorSource-line"*

 *})*;

 *va*r*styleFunction = function(feature) {*

  *return organizeStyleline(feature.value*s*\_.status);*

 *};*

 *//* *图层*

 *va*r*vectorLayer = new OlLayerVector({*

  *source: vectorSource,*

  *name: "line-source",*

  *style: styleFunction*

 *})*;

 *}*

 *return* *vectorLayer;*

*}*

**2.组织图层样式的方法**

*/\*\**

*\*组*织*地图上图标颜色*

*\**@*param {\*图片类*型*} typeId*

*\**@*param {\*文字描*述*} text*

*\**@*param {\*设备状*态*} status 0离线 1在线 2告警*

*\**@*returns 地图样*式

*\**

*\*/*

*exp*o*rt function* *organizeStyleimg(typeId,* *text, status) {*

 *var color =* *status == 0 ? "grey" : s*t*atus == 1 ? "green" :*"*red";*

 *if (typeId*=*= 10) {*

 *var svg* *= require("../../../stat*i*c/nodeImg/AJ.svg");*

 *} else if (*t*ypeId == 11) {*

 *var svg* *= require("../../../stat*i*c/nodeImg/service.svg"*)*;*

 *} else if (*t*ypeId == 12) {*

 *var svg* *= require("../../../stat*i*c/nodeImg/service.svg"*)*;*

 *} else if (*t*ypeId == 13) {*

 *var svg* *= require("../../../stat*i*c/nodeImg/service.svg"*)*;*

 *} else if (*t*ypeId == 14) {*

 *var svg* *= require("../../../stat*i*c/nodeImg/service.svg"*)*;*

 *} else if (*t*ypeId == 15) {*

 *var svg* *= require("../../../stat*i*c/nodeImg/service.svg"*)*;*

 *} else if (*t*ypeId == 16) {*

 *var svg* *= require("../../../stat*i*c/nodeImg/motor.svg");*

 *} else if (*t*ypeId == 17) {*

 *var svg* *= require("../../../stat*i*c/nodeImg/service.svg"*)*;*

 *} else if (*t*ypeId == 18) {*

 *var svg* *= require("../../../stat*i*c/nodeImg/service.svg"*)*;*

 *} else if (*t*ypeId == 19) {*

 *var svg* *= require("../../../stat*i*c/nodeImg/service.svg"*)*;*

 *} else if (*t*ypeId == 20) {*

 *var svg* *= require("../../../stat*i*c/nodeImg/service.svg"*)*;*

 *} else {*

 *var svg* *= require("../../../stat*i*c/nodeImg/motor.svg");*

 *}*

 *var geomSty*l*e = new OlStyleStyle({*

 *// 添加ci*r*cle*

 *// imag*e*: new Circle({*

 *//*r*adius: 10,*

 *//*f*ill: new Fill({*

 *//* *color: "rgba(255, 0,*0*, 0.1)"*

 *//*}*),*

 *//*s*troke: new Stroke({*

 *//* *color: color,*

 *//* *width: 1*

 *//*}*)*

 *// }),*

 *//添加spr*i*te图标*

 *image:*n*ew OlStyleIcon({*

 *opa*c*ity: 1.0,*

 *//*r*otateWithView: 28,*

 *rot*a*tion: 0.0,*

 *col*o*r: color,*

 *sca*l*e: 0.3,*

 *//*i*mg: svg,*

 *//*s*ize: 200,*

 *//*i*mgSize: [200, 200], // 这*个*大小必须为原始大小*

 *//*a*nchor: [0.2, 0.3],*

 *//*s*rc: "data:image/svg+xml;*b*ase64," + btoa(svg)*

 *src*:*svg*

 *}),*

 *text: n*e*w Text({*

 *tex*t*: text, // 添加文字描述*

 *fon*t*: "bold 18px serif", //*设*置字体大小*

 *fil*l*: new Fill({*

  *// 设置字体颜色*

  *color: "#1CAF9A"*

 *}),*

 *off*s*etX: 0,*

 *off*s*etY: 10, // 设置文字偏移量*

 *tex*t*Align: "center",*

 *tex*t*Baseline: "bottom",*

 *//*字*体阴影*

 *str*o*ke: new Stroke({*

  *color: "#149321",*

  *width: 1*

 *})*

 *})*

 *});*

 *return geom*S*tyle;*

*}*