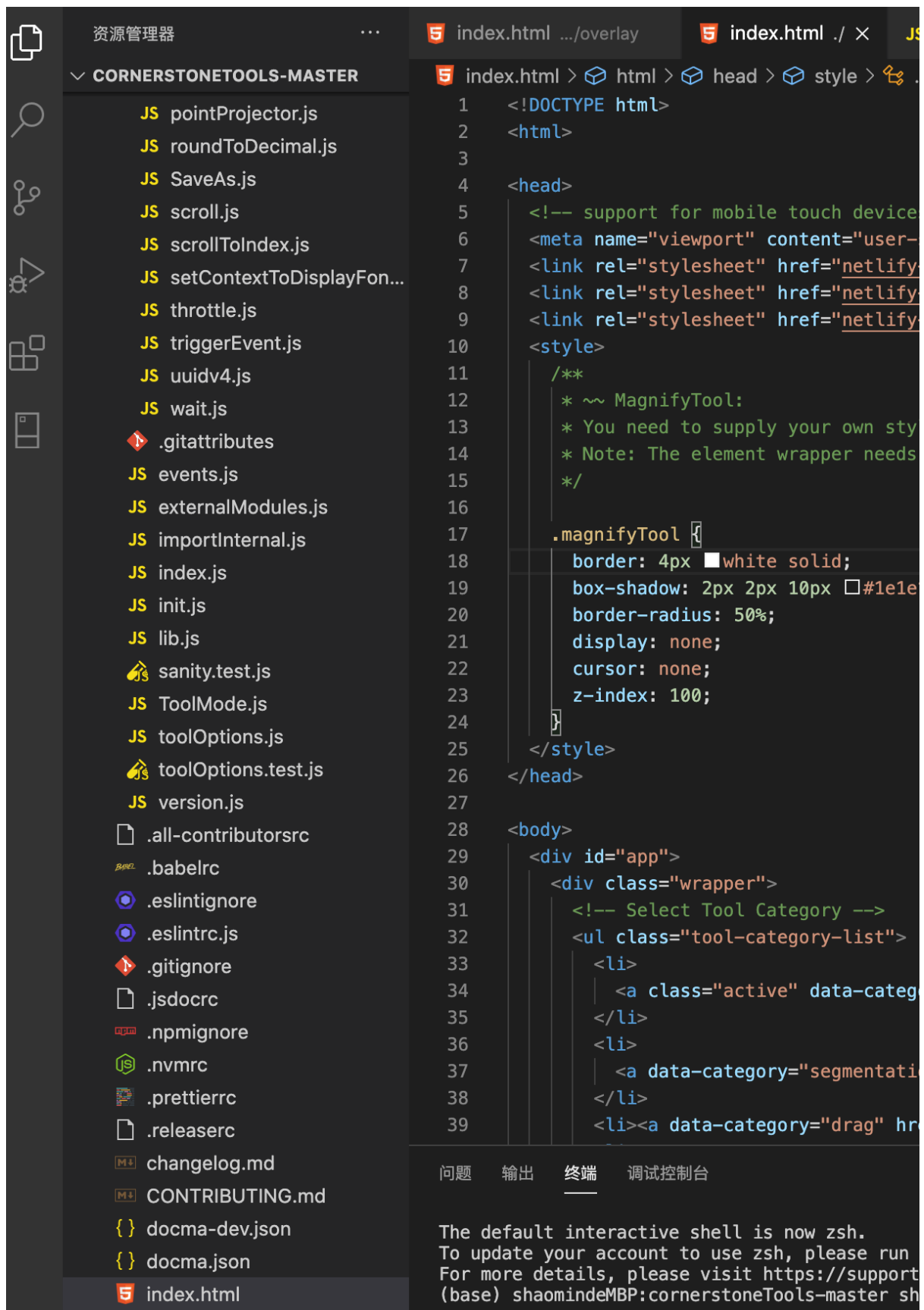


源码疑惑点

主文件为这个index.html

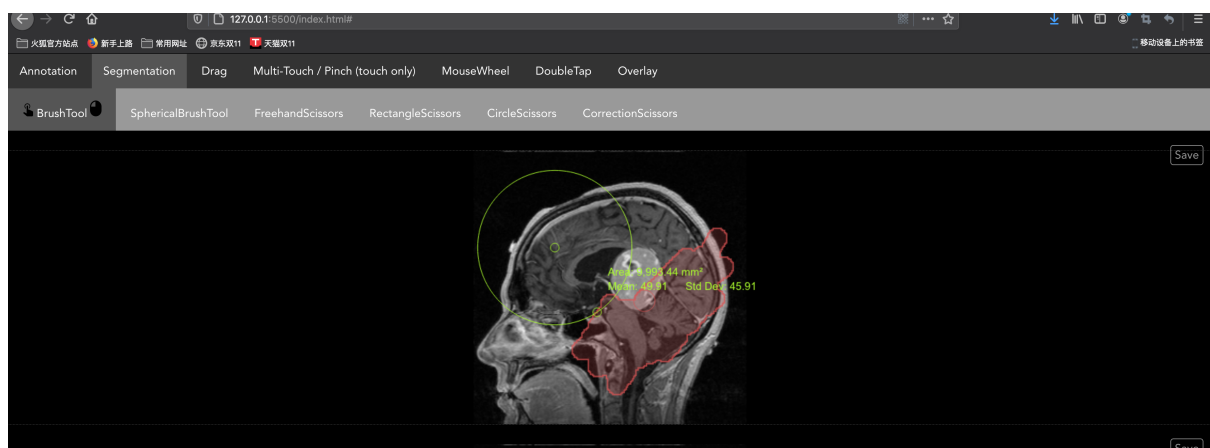


我的疑惑点很简单，在主文件index.html中可以看到所用到的工具名称，但是我不清楚他们是如何被引用的，然后数据之间是怎么传递的，看源码的目的主要是找到数据生成、传递、存储、加载的方式。

roi工具主要有两类，一个原理是用svg矢量工具，



一个是canvas画板



我可以根据名称查找到当前的工具类文件，但是不清楚他们之间是如何调用的。若是后端工程项目，可以从主函数往下一步步查找到底层函数，我不清楚这里要怎么一步步查找调用的底层工具类，还有相关的类是怎么通信的。

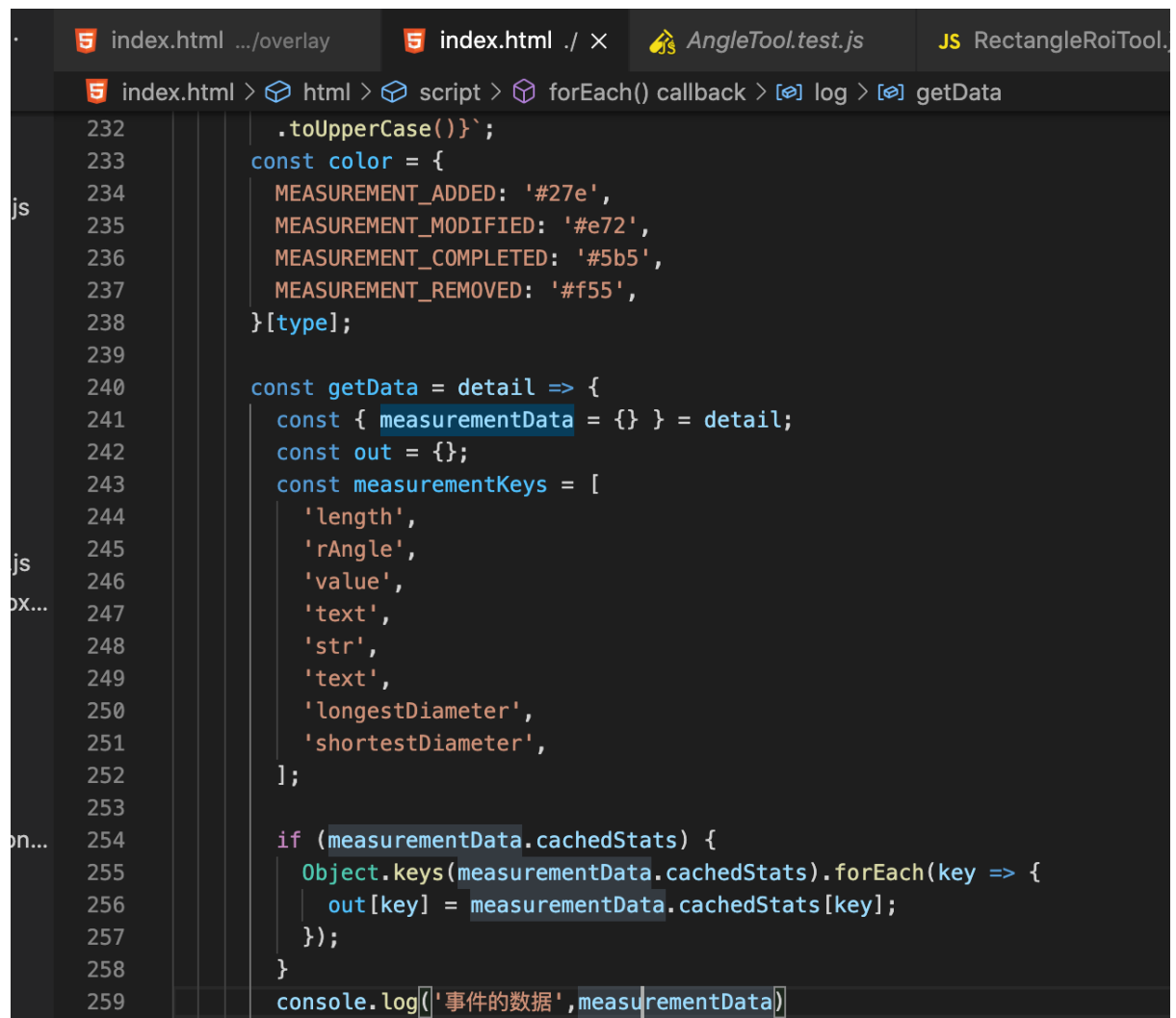
拿刷子工具举例

```
index.html .../overlay index.html ./ JS BrushTool.js X JS RectangleRotTool.js
> tools > segmentation > JS BrushTool.js > ...
7  */
8  export default class BrushTool extends BaseBrushTool {
9    constructor(props = {}) {
10      const defaultProps = {
11        name: 'Brush',
12        supportedInteractionTypes: ['Mouse', 'Touch'],
13        configuration: {},
14        mixins: ['renderBrushMixin'],
15      };
16    };
17    super(props, defaultProps);
18
19    this.touchDragCallback = this._paint.bind(this);
20  }
21
22  /**
23   * Paints the data to the labelmap.
24   *
25   * @protected
26   * @param {Object} evt The data object associated with the event.
27   * @returns {void}
28   */
29  _paint(evt) {
30    const { configuration } = segmentationModule;
31    const eventData = evt.detail;
32    const element = eventData.element;
33    const { rows, columns } = eventData.image;
34    const { x, y } = eventData.currentPoints.image;
35    // console.log('当前的坐标x=', x, '----', 'y', evt.detail);
36    // console.log('evt', evt);
37
38    if (x < 0 || x > columns || y < 0 || y > rows) {
39      return;
40    }
41
42    const radius = configuration.radius;
43    const pointerArray = getCircle(radius, rows, columns, x, y);
44
45    const { labelmap2D, labelmap3D, shouldErase } = this.paintEventData;
46  }
```

这里找到了鼠标事件的接口，可以获取当前鼠标的xy坐标，但是不知道最后的坐标集合要在哪里查找，在这里绘制调用了_paint函数，并绑定在回调函数里，不清楚是在哪里定义的touchDragCallback

```
this.touchDragCallback = this._paint.bind(this);
```

annotation系列的测量数据通过监听器的接口暴露出来了，只有测量结果没有图形路径。如下所示



The screenshot shows a web browser with three tabs: 'index.html .../overlay', 'index.html ./ ×', and 'AngleTool.test.js'. The developer console is open, showing a breadcrumb trail: 'index.html > html > script > forEach() callback > log > getData'. The code being executed is as follows:

```
232     .toUpperCase()}`;  
233     const color = {  
234         MEASUREMENT_ADDED: '#27e',  
235         MEASUREMENT_MODIFIED: '#e72',  
236         MEASUREMENT_COMPLETED: '#5b5',  
237         MEASUREMENT_REMOVED: '#f55',  
238     }[type];  
239  
240     const getData = detail => {  
241         const { measurementData = {} } = detail;  
242         const out = {};  
243         const measurementKeys = [  
244             'length',  
245             'rAngle',  
246             'value',  
247             'text',  
248             'str',  
249             'text',  
250             'longestDiameter',  
251             'shortestDiameter',  
252         ];  
253  
254         if (measurementData.cachedStats) {  
255             Object.keys(measurementData.cachedStats).forEach(key => {  
256                 out[key] = measurementData.cachedStats[key];  
257             });  
258         }  
259         console.log(['事件的数据', measurementData])
```