

支付宝容器jsAPI文档手册PDF版本

本版本为实验版本，为线下独立查看使用，受制于生成PDF程序的限制，样式问题还没有很好的解决，例如分页切图、代码高亮。
推荐使用在线版本，方便实时查看jsAPI运行效果。
如需搜索，使用阅读PDF软件自带功能即可。
更多细节样式调整和目录索引探索建设中。
生成时间: 2017-06-08 05:40:30

Since 8.6

自定义键盘

自定义键盘使用：Android和ios方案不一样，Android支持直接通过原生的input输入框唤起，ios通过jsapi唤起

0.1. iOS

- 前端DOM结构

```
< input type="text" class="h5numInput" kb-type="0" data-paste="false" maxlength="8" data-placeholder="这里调用数字键盘" />
- type: 支持 type="password"
- class="h5numInput" 标识页面初始化加载就调用自定义键盘；
- kb-type: (9.9.1之后开始支持别名data-kb-type)需要调用的键盘类型，`0`为数字键盘，`1`为身份证键盘，`2`为不带小数点的键盘，不写该属性默认为数字键盘；键盘类型随时可扩展；
- data-paste: 若为 false 则不能进行粘贴，默认为 true；
- data-placeholder: 需要显示的提示信息，不能只写placeholder；
- maxlength:最大长度
- 居左、中、右对齐: 添加 css的 text-align, 支持 left, center, right, justify, 默认为 left
```

- ios数字键盘相关 API（只有ios有，android不支持）

numInput：用于手动调用自定义键盘

```
AlipayJSBridge.call('numInput');
```

inputFocus：用于当前端校验某input 有问题， focus 到 input 输入框

```
AlipayJSBridge.call('inputFocus', {
  index: index //用于 focus 的某 input, 从0开始 (带有class含h5numInput的输入框)
});
```

inputBackFill: 用于当前端去除无用值, 返回正确的值给 native (since 9.0)

```
AlipayJSBridge.call('inputBackFill', {
  index: index, //用于回填的某 input, 从0开始 (带有class含h5numInput的输入框)
  value: 111
});
```

numInputReset: 用于重新渲染某输入框的位置 (since 9.0)

```
AlipayJSBridge.call('numInputReset', {
  index: index //用于重新渲染 input位置, 从0开始 (带有class含h5numInput的输入框)
});
```

inputBlur: 手动blur (since 9.5)

```
AlipayJSBridge.call('inputBlur', function() {
  // do something
});
```

0.1. 回调事件

```
document.addEventListener('inputEvent', function(event) {
  console.log("index:" + event.data.index);
  console.log("number:" + event.data.value);
  console.log("eventType:" + event.data.eventType);

  if(event.data.eventType == "blur") {
    // do blur
  }
  else if(event.data.eventType == "focus") {
    // do focus
  }
  else if(event.data.eventType == "keyup") {
    // do keyup
  }
  else if(event.data.eventType == "keydown") {
    // do keydown
  }
  else if(event.data.eventType == "change") {
    // do change
  }
}, false);
```

1. 使用方法

2. 代码演示

2.1. 基本功能

```
<style>
* {
  padding: 0;
  margin: 0;
}
input {
  display: block;
  padding: 8px;
  box-sizing: border-box;
  -webkit-box-sizing: border-box;
  margin: 10px auto;
}
#J_inputLog {
  line-height: 1.2;
  font-size: 12px;
}
</style>
```

```
<br>
<input
  type="text"
  class="h5numInput"
  data-kb-type="0"
  kb-type="0"
  data-paste="false"
  maxlength="6"
  data-placeholder="iOS kb-type=0"
  id="J_input1"
/>
```

```
<br>
<input
  type="text"
  class="h5numInput"
  data-kb-type="1"
  kb-type="1"
  data-paste="false"
  maxlength="6"
  data-placeholder="iOS kb-type=1"
  id="J_input2"
/>
```

```
<br>
<input
  type="text"
  class="h5numInput"
  data-kb-type="2"
  kb-type="2"
  data-paste="false"
  maxlength="6"
  data-placeholder="iOS kb-type=2"
  id="J_input3"
/>
```

```
<br>
<input
  type="number"
  placeholder="system/Android"
  id="J_input4"
/>
```

```
<h5>键盘事件日志</h5>
<div id="J_inputLog"></div>
```

```
<script>
function ready(callback) {
  // 如果jsbridge已经注入则直接调用
```

```

    if (window.AlipayJSBridge) {
        callback && callback();
    } else {
        // 如果没有注入则监听注入的事件
        document.addEventListener('AlipayJSBridgeReady', callback, false);
    }
}
ready(function() {
    function doLog(logStr) {
        document.getElementById('J_inputLog').innerHTML += logStr + '<br>';
    }

    document.addEventListener('inputEvent', function(event) {
        doLog("index:" + event.data.index);
        doLog("number:" + event.data.value);
        doLog("eventType:" + event.data.eventType);

        console.log(event);

        if(event.data.eventType == "blur") {
            // do blur
        }
        else if(event.data.eventType == "focus") {
            // do focus
        }
        else if(event.data.eventType == "keyup") {
            // do keyup
        }
        else if(event.data.eventType == "keydown") {
            // do keydown
        }
        else if(event.data.eventType == "change") {
            // do change
        }
    }, false);
});
</script>

```

2.2. Android

安卓自定义键盘直接识别type='number'， 'tel'或者其他以正则表达式约束的input输入框， 'tel'支持input的其他扩展项，如maxLength等。如果当前页面存在其他输入框，可在数字类型输入框的onfocus调用'hideKeyboard'接口，以隐藏显示的系统键盘之后再唤起自定义键盘。 < input type="number" data-placeholder="这里调用数字键盘" />

hideAllKeyboard：隐藏所有键盘（包括自定义键盘）（since 10.0.15）

```
AlipayJSBridge.call('hideAllKeyboard');
```

2.3. Android数字键盘demo

```
<h1>Android自定义数字键盘</h1>
<input type="number" data-placeholder="这里调用数字键盘"/>
```