# 可以利用js代码动态加载qml文件然后创建组件实例，

## 只需要创建组件一次就可以调用组件的createObject方法多次来创建组件实例

### 实例

|  |  |
| --- | --- |
| **//CustomColor.js**  function *getColor*(){  let *colors*=["blue","deeppink","steelblue","lime","orange","yellow"];  var *index* = *Math*.round(*Math*.random()\*10)%6; //Math.random()产生0-1之间的随机数，需要\*10变为1-10之间  //console.log(index)  return *colors*[*index*];  }  function *getRangedRnd*(max){  return *Math*.floor(*Math*.random() \* *max*)  }  function *getRndColor*(){  let *r* = *getRangedRnd*(256)  let *g* = *getRangedRnd*(256)  let *b* = *getRangedRnd*(256)  //console.log(r,g,b)  return {red: *r*/255,green: *g*/255,blue: *b*/255} //注意这个对象是所有参数的值必须在0-1之间，否则没有效果  }  //封装随机颜色生成器  function *randomColorFactory*(rgbFunction){  return function(){//返回一个函数  let *clr* =*getRndColor*()  return *rgbFunction*(*clr*.red,*clr*.green,*clr*.blue)//函数又有返回值  }  } | **//DateTimeLabel.qml**  import QtQuick 2.0  import QtQuick.Controls 2.15  import "./CustomColor.js" as *ColorGen*  Label{  width: 100  height:40  id:*lblDate*  font.bold: true  font.pixelSize: 30  //required property color rndClr  required property var rndClr  color:*rndClr*()  Timer{  repeat: true  triggeredOnStart: true  running: true  interval: 1000  onTriggered: {  *lblDate*.text = *Qt*.formatDateTime(new *Date*(),'yyyy-MM-dd hh:mm:ss')  }  }  } |
| **//main.qml**  import QtQuick 2.15  import QtQuick.Window 2.15  import QtQuick.Controls 2.15  import "./CustomColor.js" as *ColorGen*  Window {  width: 640  height: 480  visible: true  title: *qsTr*("Hello World")  Item {  id: *it*  property var randClr:*ColorGen*.randomColorFactory(*Qt*.rgba)  Column{  id:*col*  width: 200  height: 60  Component.onCompleted: {  //.创建js的Component组件,格式Qt.createComponent(组件文件名称，加载方式(同步或者异步)，父控件的id)  let *jsCom* = *Qt*.createComponent("./DateTimeLabel.qml",Component.PreferSynchronous,*it*)  *createOneDataTimeLabel*(*jsCom*)  //batchCreateDataTimeLabel(3,jsCom)  }  function *createOneDataTimeLabel*(com){ //封装创建一个组件的函数  if(*com* && *com*.status === Component.Ready){    *com*.createObject(this,{rndClr:*it*.randClr})  }  }  function *batchCreateDataTimeLabel*(num,com){ //批量创建自定义组件，在内部是调用创建一个的函数n次  for(var *i*=0;*i*<*num*;++*i*){  *createOneDataTimeLabel*(*com*)  }  }  }  }  } |  |

效果：

|  |
| --- |
|  |