1:计算器

视图:</view>

<view >

<form bindsubmit="sum">

<input name="num1" />

<text>+</text>

<input name="num2" />

<button form-type="submit">=</button>

<text>{{result}}</text>

</form>

</view>

js代码:Page({

data:{}//设置页面默认参数

bindKeyInput: function(e) {

this.setData({

inputValue: e.detail.value

})

},

sum:function(e){

var num1=e.detail.value.num1;

var num2=e.detail.value.num2;

var res=parseInt(num1)+parseInt(num2);

this.setData({result:res});

}

})

2.var app=getApp();//获取应用实例,全局

var page=this;//在页面的function中this就是当前页面对象

app.getUserInfo(function(userInfo){

page.setData({userInfo:userInfo})

})

3.获取用户输入内容时候使用 formsubmit,普通点击事件使用bindtap

4.APP.js

1>onLaunch当app启动时候触发的事件

2>wx.getUserInfo 微信内置方法

module.exports={formatTime:formatTime}把js中的函数通过module.exports导出给其他人使用

5.var util=require(“../../utils/util.js)//导入js文件

6.wxml必须严格遵守xml规范,必须有闭合标签,使用{{Messages}}绑定数据,是单项绑定,

7.progress 进度条 percent 进度值 show-info=”{{true}}” 当属性值为 true和false时候必须这样写

8.条件渲染: 隐藏时候元素不存在,如果使用hidden=”{{true}}”隐藏时候元素是存在的,如果频繁切换hidden比if效率高

<text wx:if={{n>4}}>你好1!</text>

<text wx:elif={{n>10}}>你好2!</text>

<text wx:else>你好3!</text>

9.<block>元素</block>可以把元素包起来,不影响布局

10.列表渲染 默认变量名是item 也可以通过 wx:for-item=”a” 改变item变量名wx:for-index=”b”改变索引变量名 {{index}}索引 wx:key=”{{item.id}}”给元素设定主键,优化程序,防止卡顿

<image wx:key=”{{item.id}}” wx:for=”{{meinvs}}” src=”/images/{{item.imgSrc}}”></image>

<text wx:for=”{{meinvs}}”>{{item.name}}</text>

11.下拉刷新 onPullDownRefresh 上拉加载 onReachBottom 弹窗显示个消息 wx.showToast({title:”hello“})

persons :[ {id:1,name:”zhangsan”,imgSrc:”1.jpg”},{id:2,name:”lisi”,imgSrc:”2.jpg”},{id:3,name:”wangwu”,imgSrc:”3.jpg”},{id:4,name:”zhaoliu”,imgSrc:”4.jpg”}]

menus:[]

onPullDownRefresh:function(){

var persons=this.data.persons;

var menus=this.data.menus;

if(menus.length<persons.length)//如果当前集合

{

var p=person[menus.length];

menus.push(p);

}else

{

wx.showToast({title:”已经到达底部”});

}

this.setData(menus:menus);//不写的话页面不会刷新

}

12:容器组件

1. view 横向或者纵向布局的控件 修改样式 style=” flex-direction: row; display:flex;” 横向 style =”flex-direction:column;display:flex;” 纵向 必须同时设定 display:flex; 几乎所有控件都有bindtap事件;
2. (\*)scroll-view 带滚动条的区域
3. swiper:滑块容器 轮播图

<swiper style=”hieght:600” autoplay=”{{true}}” interval=”1000” indicator-dots=”{{false}}”>

<swiper-item wx:for=”{{person}}”>

<image src=”/images/{{item.imgSrc}}”/>

</swiper-item>

</swiper>

属性:

indicator-dots 是否显示指示点:默认false

autoplay:是否自动切换:默认false

current:当前页面的序号

interval:自动切换时间毫秒,默认5000

duration:滑动动画时间毫秒,默认1000

bindchange:页面改变事件 event.detail.current 可以获取当前的序号

13:icon 内置的图标:<icon type=”success” size=”80” color=”blue”> icon库参考文档

14:text 文本;支持嵌套和转译符,和换行 “\n”

15:<switch/> <input/> <checkbox/> <slider/><radio/> <picker/>放到form中,则可以在bindsubmit事件中获得表单的内容(需要有name),点击form中的<button form-type=”submit”>会提交表单触发bindsubmit;如果点击<button form-type=”reset”>表单会清空,并且触发bindreset; checkbox/radio必须放到 checkbox-group/radio-group里面,并且给check-group/radio-group附加属性name

16:button 按钮

size:尺寸,有效的值 default,mini;

type:样式.有效值 primary ,default,warn

plain:是否背景色透明;默认是 false ,如果属性是bool类型的,那么plain和plain={{true}}是等价的.

disabled:是否禁用;

loading:是否显示”加载中”的图标,标示”正在执行中”

form-type:可选值 submit、reset；

17:checkbox多选/radio单选 value不是显示的值是获的选中的值 也要放到checkbox-group中

<label for=”checkid”>第一个</label> label配合checkbox需要for指向checkbox的id

check-box更适合表单提交的时候,而switch适合于立即响应

checkbox-group可以绑定bindchange事件,可以监听checkbox改变事件 e.detail.value获取的是数组类型,WXML中的组件如果是以intertext方式(而不是属性的方式)设置,那么就不仅仅可以放文本,还可以放image等组件.

18:switch 开关

checked :是否选中 bool值

type:类型,可选值 switch,checkbox

bindchange:选中发生改变的事件

19:slider 进度滑条

min:最小值,默认0;max最大值,默认100;

step:步长,可被整除

disabled:禁用

value:当前值

bindchange:进度改变时候触发的事件

show-value:是否在右侧显示当前值

20:input 输入框

value:初始内容

type:输入的内容 有效值:text,Number,idcard,digit,time,date.模拟器不支持,在真机上影响键盘布局

password:bool,是否密码框.

placeholder:唯恐事后的提示文字;用input时候最好都设定placeholder

disabled:禁用

maxlength:最大输入长度

auto-focus是否自动聚焦;

bindfocus,bindblur;获得/市区焦点的事件

{{value}}是单向绑定,

setDat()如果设置的值和现有的值一样,那么就不会更新界面,可以先改成其他值,再改回.

21;textarea 多行文本 auto-height:bool值,自动适应高度

22:picker 选择器,可以选择日期\时间以及列表数据

1. mode属性有三个可选值:selector 列表;time 时间;date 日期
2. 通用属性\事件;value当前选中的值:mode为selector时为数字,标示选择了range中的第几个,从0开始.date\time时为日期;bindchange:选项发生改变的事件,event.detail.value获取value的值
3. 特有属性:mode=selector时,range为一个数组,标示可选数据;mode=date时,start\end为起止时间;fields(有效值 year,month,day)表示选择的粒度;mode=time时,start\end为起止时间;

注意,picker并不是可见的控件,需要把可见控件作为picker的子元素,可以用按钮\view\文本等几乎任意控件.

23:navigator 超链接 url=”pages/test1/test1” 跳转指定页面 redirect:bool类型,默认是false, true为重定向指定页面,无法后退.可以像网址一样带querystring:/Test1/Test?name=zhangsan&age=14,取值时候,在onload事件里面 onload:function(params){ let name=params.name,let age=params.age }

24:image 图片组件 mode:image组件默认宽度高度300px\高度225px.图片的缩放模式,可选值:scaleToFill不保持横纵比 ,aspectFit图片保持横纵比,宽的方向有空白,窄的方向填充满,aspectFil宽的方向裁减,窄的完整显示等 bindError/bindLoad绑定错误和加载事件.

25:audio 音乐组件 id:标志 给wx.createAudioContext()创建用

src:音频文件,播放网络地址 loop:bool 是否循环播放,默认false; controls:bool,是否显示播放控制条,默认true;poster:封面图片地址;name:显示到歌名;author:作者名;事件:binderror(错误,event.detail.MediaError.code获得错误码,和文档中可选值比较)bindplay(开始播放,继续播放)bindpause(暂停)bindtimeupdate(event.detail.currentTime获取当前播放进度,event.detail.duration获取音乐时长.都是小数类型,以秒为单位)bindended(播放结束)

使用AudioContext对audio组件进行播放/暂停/定位等控制;调用wx.createAudioContext(‘myAudio’)根据audio 的id创建AudioContext对象;AudioContext的方法:play播放;pause暂停;seek(10)跳转到第10秒;

26:video 不支持所有视频格式:只支持开放格式

video标签默认宽度300px高度225px; id:标示;src视频地址;controls,是否系那是播放控制界面;danmu-list:弹幕对象数组,对象包含text\color\timer三个属性;danmu-btn:bool是否显示弹幕切换按钮,默认false,只能在初始时赋值;autoplay:bool,是否自动播放,默认false

事件:bindplay\bindpause\bindended\binderror(可以通过event.detail.errMsg获取文本类型的错误报错信息)

27:微信API API接口一般都接收一个object作为参数,api都是异步的,同步的以sync结尾.

object中可以指定success,FAIL,complete来接收接口调用结果.成功失败都会执行complete

28:小程序的三种类型文件:资源文件\本地临时文件\本地持久文件,调用wx.saveFile把淋湿而文件持久化保存,以后也可以使用.微信小程序无法直接读写用户手机上的文件,也无法读写其他APP的文件,只能读写给你自己划定的一个区域的文件.

29:网络通讯:手机端和服务器之间采用https(模拟器支持http,正式环境只支持https,因为更安全)websocket(高效实时通讯)两种通讯方式.正式环境中只能向小程序服务器端管理.wx.request发出https请求类似ajax,同时只能有5个请求;url/data/header(请求的报文头\对象\不能设置Referer)/method(POST\GET\PUT\DELETE等,支持restful)/success(返回值的data为返回内容)/fail/complete