1. box-sizing: border-box; 规定两个并排的带边框的框

|  |  |
| --- | --- |
| border-box | 为元素设定的宽度和高度决定了元素的边框盒。  就是说，为元素指定的任何内边距和边框都将在已设定的宽度和高度内进行绘制。 |

1. border-right: 1rpx solid #d9d9d9; /\*右边框样式：厚度，实线，颜色\*/
2. flex-flow属性是flex-direction属性和flex-wrap属性的简写形式，默认值为row nowrap。
3. 在.wxml文件的通过data-\* 传值，后台通过e.currentTarget.dataset.\*取值。

在循环出来的界面中，不同的界面可触发不同点击事件

<view class='btnimg' wx:for="{{imgTaps}}" wx:key="" bindtap="aaa" data-id='{{item.id}}'>

</view>

aaa: function (e) {

console.log(e.currentTarget.dataset.id);

}

1. Swiper 滑块视图容器。可用于轮播图片
2. 引用工具函数

头部引入：const WxParse = require('../../utils/ wxparse.js');

Js里使用：WxParse.lunbo

../../utils/ wxparse.js中添加：

function lunbo () {

return

}

module.exports = {

lunbo: lunbo,

}

7.//获取当前路径下的URL，可当工具函数使用

/\*获取当前页url\*/

function getCurrentPageUrl() {

var pages = getCurrentPages() //获取加载的页面

var currentPage = pages[pages.length - 1] //获取当前页面的对象

var url = currentPage.route //当前页面url

return url

}

/\*获取当前页带参数的url\*/

function getCurrentPageUrlWithArgs() {

var pages = getCurrentPages() //获取加载的页面

var currentPage = pages[pages.length - 1] //获取当前页面的对象

var url = currentPage.route //当前页面url

var options = currentPage.options //如果要获取url中所带的参数可以查看options

//拼接url的参数

var urlWithArgs = url + '?'

for (var key in options) {

var value = options[key]

urlWithArgs += key + '=' + value + '&'

}

urlWithArgs = urlWithArgs.substring(0, urlWithArgs.length - 1)

return urlWithArgs

}

8.标准请求

onLoad: function (options) {

var that = this;

wx.request({

url: CONFIG.API\_URL.keshi,

data: {

id:options.name

},

success: function (response) {

that.setData({

title: ti

})

//// WxParse.wxParse('detail', 'html', data.content, that, 25);

}

});

},

9.富文本编码转换

现在头部引入：const WxParse = require('../../utils/wxParse/wxParse.js');

再使用：WxParse.wxParse('detail', 'html', data.content, that, 25);

10. 带数据发送请求：

wx.request({

url: CONFIG.API\_URL.keshi, //先通过配置文件config.js添加keshi

data: {

id:options.name

},

success: function (response) {

console.log(response);

}

});

再在后台：namespace addons\shuiguo\shop\controller下的Api.php中

添加keshi()方法。

方法中获取传过来的参数用：$id=input('param.id');

11. 在.html文件中套入tp5判断语法：

{if condition="$category.pid eq '0'"} 哈哈 //$category.pid === 0

{else /} 呵呵

{/if}

11.造成发送两次请求的原因：const BASE = 'https://yiy.ggyxkj.com';

http没有改成https

12.

tp5查询数据库单条语句：

$data=Db::table('表名')->where('字段','eq',$id传过的值)->order('id asc')->find();

查询完毕返回json类型信息：return json\_encode($data);

查询多条数据：

$data=Db::table('addon\_article\_category')->where('pid','eq',$id)->select();

连表查询，重组字段：

$data=Db::table('addon\_article\_category')->alias('a')->join('file f','f.id=a.file\_id','LEFT')

->field('a.id,a.name,f.savepath,f.savename')->where('pid','eq',$id)->select();

模块查询：

public function lunbo(){

$Slide =AddonShopSlids::with(['file'=>function($query){$query->field('id,savepath,savename');}])->order('rank desc,id desc')->field('id,file\_id,goods\_id')->select()->toArray();

return json\_encode($Slide);

}

添加滚动，隐藏x滚动条

style="overflow:scroll;overflow-x:hidden;"

循环输出富文本

获取数据：lunbotext=[{…}, {…}, {…}]

0:{id: 96, category\_id: 158, title: "微信团队分享：“蘑菇街女装特卖”小程序实", author: "", sub: "", …}

1:{id: 95, category\_id: 158, title: "微信团队分享：小程序产品能力", author: "", sub: "", …}

2:{id: 94, category\_id: 158, title: "商业转化", author: "餐饮小程序方法论", sub: "", …}

Index.Wxml:

<template is="wxParse" data= "{{wxParseData:replyTemArray[index]}}" />

Index.js:

for (let i = 0; i < lunbotext.length; i++) {

WxParse.wxParse('detail' + i, 'html', lunbotext[i].content, that, 25);

if (i === lunbotext.length - 1) {

WxParse.wxParseTemArray("replyTemArray", 'detail', lunbotext.length, that)

}

}

图片左右滑动效果图：



//index.wxml

<scroll-view scroll-x="true">

<view class="uploadWrap" scroll-x="true">

<view class="upload\_Item">

<image class="upload\_Item\_img" src="/pages/image/1.png"></image>

</view>

<view class="upload\_Item">

<image class="upload\_Item\_img" src="/pages/image/1.png"></image>

</view>

<view class="upload\_Item">

<image class="upload\_Item\_img" src="/pages/image/1.png"></image>

</view>

<view class="upload\_Item">

<image class="upload\_Item\_img" src="/pages/image/1.png"></image>

</view>

<view class="upload\_Item">

<image class="upload\_Item\_img" src="/pages/image/1.png"></image>

</view>

<view class="upload\_Item">

<image class="upload\_Item\_img" src="/pages/image/1.png"></image>

</view>

<view class="upload\_Item">

<image class="upload\_Item\_img" src="/pages/image/1.png"></image>

</view>

</view>

</scroll-view>

//index.wxss

.uploadWrap {

height: 300rpx;

width: 100%;

display: flex;

display: -webkit-box;

flex-direction: column;

}

.upload\_Item {

width: 300rpx;

height: 300rpx;

flex: 1;

}

.upload\_Item\_img {

width: 300rpx;

height: 300rpx;

}

13.两个方法传值

第一个方法中赋值：

that.setData({

logourl: tempFilePaths

})

第二个方法取值：

Logourl=that.data.logourl;

14.上传单张图片

/\*\*上传证件idnex.js\*/

uptrafficimg: function () {

var that = this;

//选择相册

wx.chooseImage({

count: 1, // 默认9

sizeType: ['original', 'compressed'], // 可以指定是原图还是压缩图，默认二者都有

sourceType: ['album', 'camera'], // 可以指定来源是相册还是相机，默认二者都有

success: function (res) {

var tempFilePaths = res.tempFilePaths;

that.setData({

logourl: tempFilePaths

})

wx.uploadFile({

url: CONFIG.API\_URL.upImage,

filePath: tempFilePaths[0],

name: 'file',

formData: {

'user': 'test'

},

success: function (res) {

var data = res.data;

console.log(data);

//do something

}

})

}

})

},

/\*\* 服务器//上传图片

function upImage(){

$file=request()->file('file');

if($file){

$info = $file->move(ROOT\_PATH . 'public' . DS . 'uploads');

if($info){

// 成功上传后 获取上传信息

// 输出 jpg

// echo $info->getExtension();

// 输出 20160820/42a79759f284b767dfcb2a0197904287.jpg

echo $info->getSaveName();

// 输出 42a79759f284b767dfcb2a0197904287.jpg

echo $info->getFilename();

}else{

// 上传失败获取错误信息

echo $file->getError();

}

}

}

}

15.验证手机和邮箱：

if (!(/^1[34578]\d{9}$/.test(e.detail.value.phone))) {

wx.showModal({

title: '温馨提示',

content: '电话号码格式有误！',

})

}

if (!(/^[a-zA-Z0-9\_-]+@[a-zA-Z0-9\_-]+(\.[a-zA-Z0-9\_-]+)+$/.test(e.detail.value.email))){

wx.showModal({

title: '温馨提示',

content: '邮箱格式有误！',

})

}

16.超出行数隐藏。。。

.detailsdiv1 {

line-height: 60rpx;

font-size: 28rpx;

padding-left: 20rpx;

padding-right: 10rpx;

padding-bottom: 10rpx;

display: -webkit-box;

word-break: break-all;

text-overflow: ellipsis;

overflow: hidden;

-webkit-box-orient: vertical;

-webkit-line-clamp: 4;

}

<view class=" {{show==1?'show':'detailsdiv1'}}">

////////一行隐藏

overflow:hidden;

white-space:nowrap;

text-overflow:ellipsis;

/\*\*

\* 页面上拉触底事件的处理函数

\*/

onReachBottom: function () {

var that = this;

console.log('上啦')

// if (this.data.loading) return;

// this.setData({ loading: true });

// that.updateRefreshIcon.call(this);

that.setData({

page: (that.data.page + 1)

});

console.log(that.data.page)

that.latlog(that.data.page);

// setTimeout(() => {

// wx.showToast({

// title: '加载中',

// icon: 'loading',

// duration: 2000

// })

// }, 3000)

},

/\*\*

\* 旋转刷新图标

\*/

// updateRefreshIcon: function () {

// var deg = 0;

// console.log('旋转开始了.....')

// var animation = wx.createAnimation({

// duration: 3000

// });

// var timer = setInterval(() => {

// if (!this.data.loading)

// clearInterval(timer);

// animation.rotateZ(deg).step();//在Z轴旋转一个deg角度

// deg += 360;

// this.setData({

// refreshAnimation: animation.export()

// })

// }, 3000);

// },

//入驻的汽车信息/本地服务加载

function getSettledlist(){

$s = input('param.');

$count = 3;

$yema = $count\*($s['page']-1);

$data=Settledlist::order('id desc')->limit($yema,$count)->select();

foreach ($data as $k => $v) {

$li=explode(",",$v['file\_id']);

$file = array();

foreach ($li as $ks => $vs) {

$file[] = Db::name('file')->find($vs);

}

$data[$k]['file']=$file;

}

return json\_encode($data);

}

//同一个表，先把相同属性a 的 属性b值相加 ，再排序

//排行榜

function rank(){

$data=Db::table('jh\_payments')

->field('\*,sum(paymoney) as sum\_pay')

->group('uid')

->order('sum\_pay desc')

->select();

return json\_encode($data);}

///渐变

background: -webkit-linear-gradient(left, #7DE369 , #c9dd22); /\* Safari 5.1 - 6.0 \*/

background: -o-linear-gradient(right, #7DE369, #c9dd22); /\* Opera 11.1 - 12.0 \*/

background: -moz-linear-gradient(right, #7DE369, #c9dd22); /\* Firefox 3.6 - 15 \*/

background: linear-gradient(to right, #7DE369 , #c9dd22); /\* 标准的语法（必须放在最后） \*/

微信小程序跳转传参

一般都是传字符串到下一页，如果要想传对象怎么办呢？

我的解决办法是先将对象转换为json字符串然后到下个页面将json字符串，再转化为对象。如下：

let str=JSON.stringify(e.currentTarget.dataset.item);

wx.navigateTo({

url: '../toMybaby/babyDetail/babyDetail?jsonStr='+str,

})

//另一个页面加载

onLoad:function(options){

// 生命周期函数--监听页面加载

let item=JSON.parse(options.jsonStr);

this.setData({ward:item});

},

PHP 将json的stdClass Object转成数组array 方法二

1. //$user是一个stdClass  每次调用都是 $user->uid  其实也是挺方便的
2. //把$user转为array数组
3. $user  =json\_encode($user); //把她转换为json字符串
4. $user = json\_decode($user,true); //再把json字符串格式化为数组
5. //或者直接是:
6. $user = json\_decode(json\_encode($user),true);

//判断标签

{switch name="$vo.odstatus"}

{case value="0"}待支付{/case}

{case value="1"}已完成{/case}

{case value="2"}已取消{/case}

{default /}default

{/switch}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*获取天气

//小程序获取

//请求的参数，

var datas = {

city:'哈尔滨市'

};

//请求的路径

var postUrl = "https://www.sojson.com/open/api/weather/json.shtml";//请求的路径

app.postDatas('GET', postUrl, datas, function (res) {

console.log(res)

})

///////服务器端获取

$data =file\_get\_contents('http://www.weather.com.cn/data/sk/101300801.html');

$obj=json\_decode($data);

dump($obj);

die;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*去重

->group("building")

//记录操作日志

$request= \think\Request::instance();

$param = ['name'=>'用户登录','way'=>$request->module().'/'.$request->controller().'/'.$request->action(),'username'=>session('username'),'create\_time'=>date('Y-m-d H:i:s',time()),'ip'=>$\_SERVER['REMOTE\_ADDR']];

//保存日志

saveLog($param);

/\*\*\*\*\*\*\*\*带参数分页

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*订单管理\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//订单列表

public function order()

{

$where=array();

$where['odstatus']=1;//默认先显示已支付

if($this->request->get()){

$post = $this->request->get();

if (isset($post['order\_id'])&&isset($post['odstatus'])&&isset($post['test10'])&&isset($post['ps\_status'])) {

if($post['order\_id'] != ''){

$where['order\_id']=array('like','%'.$post['order\_id'].'%');

}

if($post['odstatus'] != ''){

$where['odstatus']=(int)$post['odstatus'];

}

}

}

$order\_list = OrderModel::where($where)

->order('status desc,id desc')

->paginate(12,false,['query'=>request()->param()]); //带参数分页

$this->assign('orders', $order\_list);

return $this->fetch();

}

强制换行

      1、word-break: break-all;       只对英文起作用，以字母作为换行依据。

      2、word-wrap: break-word;   只对英文起作用，以单词作为换行依据。

      3、white-space: pre-wrap;     只对中文起作用，强制换行。

禁止换行

      white-space:nowrap; overflow:hidden; text-overflow:ellipsis;

/\*\*\*\*\*\*\*\*模板调用方法

<!-- 调用公共方法settled\_name，$vo.settled\_id作为参数。 参考手册：模板->使用函数https://www.kancloud.cn/manual/thinkphp5/125005 -->

<td>{$vo.settled\_id | settled\_name}</td>

TP5的连表查询加计算总数

$list=Db::name("storage\_name")

->alias('a')

->join('storage b','a.name=b.storage\_name')

->field(['a.id,a.name,sum(b.number) as count'])

->group('a.name')

->paginate(15);