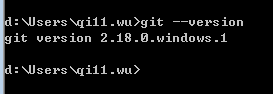
2018.7.11评审问题汇总

1. 在派件反馈中增加反馈原因、备注信息两栏
2. 在检索中增加市的选择
3. 将消息推送放在右上角、地图放在附近站点模块中
4. 将分类中的已完成、未完成改成已送达、未送达、已取消
5. 培训和认证不分顺序，去除标号，尽量在一个页面显示培训和认证两个模块
6. 在线客服放在右上角
7. 昨日订单中删掉订单详情、增加统计信息
8. 删掉金额下面的可提现金额
9. 删掉身份认证\_确认信息模块
10. 在骑手信息页面中黄色背景色中增加其他颜色的元素，显得没那么突兀
11. 联系Ta、派件反馈中黄色按钮太深
12. 我的钱包中余额明细中删掉单量的信息，选择其他元素填充

13.logo颜色太深，深蓝色底反差太大

14.接单率、活跃骑手是否需要展示，展示位置能否放在附近站点中

<http://jsbin.com/bupawagesi/1/edit>



1. git的使用
2. Vscode的使用
3. <https://www.jb51.net/article/139141.htm>
4. 观察和响应 Vue 实例上的数据变动：****侦听属性****
5. <template>
7. <div>
8. <p v-bind:title="msg">显示页面加载信息</p>
9. </div>
10. </template>
11. <script>
12. export default {
13. name: 'HelloWorld',
14. data () {
15. return {
16. msg: {'页面加载于'+new Date().toLocaleString()}}
17. }
18. }
19. }
20. </script>

var test=new Vue({

el:'#app',

data:{firstmsg:'hello',

lastmsg:'world'},

complete:{

fullmsg:{

get:function(){

return this.firstmsg+' '+this.lastmsg;

},

set:function(newvalue){

var msg=newvalue.split('')

this.firstmsg=msg[0]

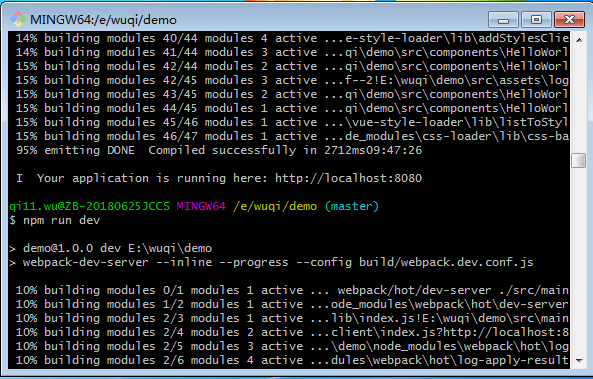
this.lastmsg=msg[msg.length-1]

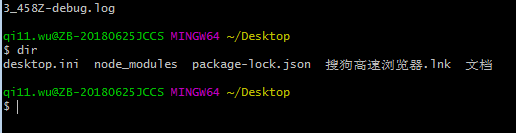
}

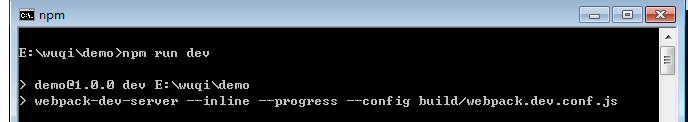
}

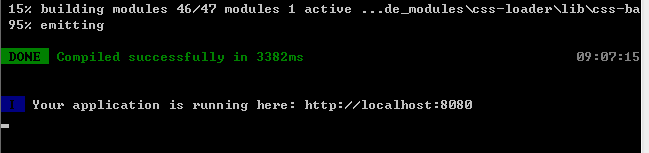
}

})









<template>

<div id="app">

<p>{{msg}}</p>

</div>

</template>

<script>

export default {

name: 'HelloWorld',

el:'#app',

data () {

return{

msg:'hello!'

}

}

}

</script>

<div id="components-demo">

<button-counter></button-counter>

<p>{{count}}</p>

</div>

<script>

export default{

name:'button-counter',

el:'#components-demo',

data:function()

{

return{

count:0

}

},

template:'<button v-on:click="count++">You click me {{count}} times.</button>'

}

</script>

<template>

<div>

name:<input v-model.trim="obj.name">

age:<input v-model="obj.age">

<button @click="add">submit</button>

</div>

</template>

<script>

import testObj from './test.vue'

export default {

data () {

return {

msg:'luojie',

msg2:'luoshengmen'

};

},

components: {testObj},

computed: {},

mounted () {

},

methods: {

clickEvent(){

alert(this.msg+'a')

}

}

}

</script>

<!-- Add "scoped" attribute to limit CSS to this component only -->

<style scoped>

h1, h2 {

font-weight: normal;

}

ul {

list-style-type: none;

padding: 0;

}

li {

display: inline-block;

margin: 0 10px;

}

a {

color: #42b983;

}

</style>

this.items.splice(firstIndex,1,{

id:firstIndex+1,

name:name,

age:age,

isShow:true

<template>

<div id="test">

name:<input v-model.trim="msgname">

age:<input v-model="msgage">

<button @click="add">submit</button>

<button @click="showAll(items)">显示全部</button><br>

<input v-model="searchMsg">

<button @click="search(searchMsg,items)">Search</button>

<table><tr>

<th>id</th><th>name</th><th>age</th><th>删除</th><th>修改</th>

<tr v-show="item.isShow&&item.flag"

v-for="(item) in items"

:key="item.id">

<td>{{item.id}}</td>

<td>{{item.name}}</td>

<td>{{item.age}}</td>

<td><button @click="del(item)">del</button></td>

<td><button @click="update(item)">update</button></td>

</tr>

</table>

</div>

</template>

<script>

export default {

data() {

return {

msgage: "",

msgname: "",

searchMsg:"",

items: [

{ id: 1, name: "nasng", age: 20, isShow: true ,flag:true},

{ id: 2, name: "abfhf", age: 30, isShow: true ,flag:true}

]

}

},

methods: {

add() {

this.items.push({

id: this.items.length + 1,

name: this.msgname,

age: this.msgage,

isShow: true

})

},

del(item) {

item.isShow = false;

},

showAll(items) {

for (var i = 0; i < this.items.length; i++) items[i].isShow = true;

},

update(item) {

var name = prompt("请输入姓名");

var age = prompt("请输入年龄");

item.name = name;

item.age = age;

},

search(searchMsg, items) {

var searchResult=[];

var msg=this.items;

for (var i = 0; i < msg.length; i++) {

var str = msg[i].name + "," + msg[i].age;

msg[i].flag = str.indexOf(searchMsg) > -1 ? true : false;

searchResult.push(msg[i]);

}

this.items = searchResult;

}

},

components: {},

computed: {},

mounted() {}

};

</script>

<!-- Add "scoped" attribute to limit CSS to this component only -->

<style scoped>

h1,

h2 {

font-weight: normal;

}

ul {

list-style-type: none;

padding: 0;

}

li {

display: inline-block;

margin: 0 10px;

}

a {

color: #42b983;

}

</style>

<template>

<div id="test">

<ul><li><button @click="showdetail('球类')">球类</button></li>

<li><button @click="showdetail('粮食')">粮食</button></li>

<li><button @click="showdetail('衣服')">衣服</button></li>

</ul>

<ul v-for="item in arr" :key="item.id">

<li><img :src="item.src"/></li><br>

<li>{{item.price}}</li><br>

<li>{{item.msg}}</li><br>

</ul>

</div>

</template>

<script>

export default {

data() {

return {

arr:[],

obj: {

"球类": [

{ src: "/static/football.jpg", price: 10, msg: "足球" },

{ src: "/static/basketball.jpg", price: 20, msg: "篮球" },

{ src: "/static/tennis.jpg", price: 30, msg: "乒乓球" }

],

"粮食": [

{ src: "/static/husked.jpg", price: 20, msg: "大米" },

{ src: "/static/corn.jpg", price: 5, msg: "玉米" },

{ src: "/static/mung.jpg", price: 10, msg: "绿豆" }

],

"衣服": [

{ src: "/static/T.jpg", price: 40, msg: "T恤" },

{ src: "/static/sports.jpg", price: 60, msg: "运动服" },

{ src: "/static/coat.jpg", price: 50, msg: "外套" }

]

}

};

},

methods: {

showdetail(cmsg) {

this.arr=this.obj[cmsg]

}

},

components: {},

computed: {},

mounted() {}

};

</script>

<!-- Add "scoped" attribute to limit CSS to this component only -->

<style scoped>

h1,

h2 {

font-weight: normal;

}

ul {

list-style-type: none;

padding: 0;

}

li {

display: inline-block;

margin: 0 10px;

}

a {

color: #42b983;

}

</style>

<template>

<div id="test">

<ul v-for="(key,value) in object" :key="key">

<li>{{key}}</li>

<li>{{value}}</li>

</ul>

<input v-model="c">

<button @click="addmsg('c')">add</button>

</div>

</template>

<script>

export default {

data() {

return {

c:"",

object:{

a:1,b:2,

}

}},

methods: {

addmsg(c)

{

// item={

// id:item.id,

// name:item.name,

// age:item.age,

// sex:this.msgsex

this.object = Object.assign({}, this.object, { c:this.c })

}

},

components: {},

computed: {},

mounted() {}

};

</script>

<!-- Add "scoped" attribute to limit CSS to this component only -->

<style scoped>

h1,

h2 {

font-weight: normal;

}

ul {

list-style-type: none;

padding: 0;

}

li {

display: inline-block;

margin: 0 10px;

}

a {

color: #42b983;

}

</style>

2018.7.31

<template>

<div>

<el-input placeholder="请输入你的名字" v-model="msgname"></el-input>

<el-input placeholder="请输入你的年龄" v-model="msgage"></el-input>

<el-button type="primary" @click="add()">添加信息</el-button>

<el-table :data="tabledata" border style="width:60%;text-align:left">

<el-table-column prop="id" label="序号" width="180">

</el-table-column>

<el-table-column prop="name" label="姓名" width="180">

</el-table-column>

<el-table-column prop="age" label="年龄" width="180">

</el-table-column>

<el-table-column label="delete" width="120">

<template slot-scope="scope">

<el-button @click.native.prevent="delete(scope.$index,tabledata)" size="small" type="primary">

删除

</el-button>

</template>

</el-table-column>

</el-table>

</div>

</template>

<script>

export default {

data() {

return {

msgname:'',

msgage:'',

tabledata:[{id:1,name:"max",age:"20"},

{id:2,name:"nancy",age:"30"},

{id:3,name:"cindy",age:"40"}]

};

},

methods: {

add(){

this.tabledata.push({id:this.tabledata.length+1,name:this.msgname,age:this.msgage})

},

delete(index,rows){

rows.splice(index,1);

}

}

};

</script>

<!-- Add "scoped" attribute to limit CSS to this component only -->

<style scoped>

h1,

h2 {

font-weight: normal;

}

ul {

list-style-type: none;

padding: 0;

}

li {

display: inline-block;

margin: 0 10px;

}

a {

color: #42b983;

}

.el-header {

background-color: #B3C0D1;

color: #333;

line-height: 60px;

}

.el-aside {

color: #333;

}

.el-select .el-input {

width: 130px;

}

.input-with-select .el-input-group\_\_prepend {

background-color: #fff;

}

</style>

2018.8.1

<template>

<div>

<el-input placeholder="请输入你的名字" v-model="msgname" size="large"></el-input>

<el-input placeholder="请输入你的年龄" v-model="msgage" size="large"></el-input>

<el-button type="primary" @click="add()">添加信息</el-button>

<el-input placeholder="请输入搜索关键字" v-model="searchmsg" size="large"></el-input>

<el-button type="primary" @click="search(searchmsg,tabledata)">搜索</el-button>

<el-table :data="tabledata" border style="width:60%;text-align:left">

<el-table-column prop="id" label="序号" width="180">

</el-table-column>

<el-table-column prop="name" label="姓名" width="180">

</el-table-column>

<el-table-column prop="age" label="年龄" width="180">

</el-table-column>

<el-table-column label="delete" width="120">

<template slot-scope="scope">

<el-button

@click="Delete(scope.$index, tabledata)"

type="primary"

size="small">

删除

</el-button>

</template>

</el-table-column>

<el-table-column label="edit" width="120">

<template slot-scope="scope">

<el-button

type="primary"

size="small"

@click="edit(scope.$index)">编辑</el-button>

</template>

</el-table-column>

</el-table>

</div>

</template>

<script>

export default {

data() {

return {

msgname: "",

msgage: "",

searchmsg: "",

tabledata: [

{ id: "1", name: "nancy", age: "20" },

{ id: "2", name: "cindy", age: "30" },

{ id: "3", name: "lucy", age: "40" }

]

};

},

methods: {

add() {

this.tabledata.push({

id: this.tabledata.length + 1,

name: this.msgname,

age: this.msgage

});

},

Delete(index, rows) {

rows.splice(index, 1);

},

edit(index) {

var newname = prompt("请输入更新后姓名");

var newage = prompt("请输入更新后年龄");

this.tabledata[index].name = newname;

this.tabledata[index].age = newage;

},

search(searchmsg, tabledata) {

var searchResult = [];

var items = this.tabledata;

for (var i = 0; i < items.length; i++) {

var msg = items[i].name + "," + items[i].age;

if (msg.indexOf(searchmsg) > -1) {

searchResult.push(items[i]);

}

}

this.tabledata = searchResult;

}

}

};

</script>

<!-- Add "scoped" attribute to limit CSS to this component only -->

<style scoped>

h1,

h2 {

font-weight: normal;

}

ul {

list-style-type: none;

padding: 0;

}

li {

display: inline-block;

margin: 0 10px;

}

a {

color: #42b983;

}

.el-header {

background-color: #b3c0d1;

color: #333;

line-height: 60px;

}

.el-aside {

color: #333;

}

.el-select .el-input {

width: 130px;

}

.input-with-select .el-input-group\_\_prepend {

background-color: #fff;

}

.el-input {

width: 200px;

float: left;

margin-bottom: 10px;

}

.el-button {

float: left;

margin-right: 10px;

margin-bottom: 10px;

}

</style>

2018.8.6

<template>

<div>

<el-menu :default-active="activeIndex" class="el-menu-demo" mode="horizontal" @select="handleSelect">

<el-menu-item Index="1">球类</el-menu-item>

<el-menu-item Index="2">粮食</el-menu-item>

<el-menu-item Index="3">衣服</el-menu-item>

</el-menu>

<div class="line"></div>

<div>

<el-table :data="arr">

<el-table-column props="src" label="图片" width=120></el-table-column>

<el-table-column props="name" label="名称" width=120></el-table-column>

<el-table-column props="price" label="价格" width=120></el-table-column>

</el-table>

</div>

</div>

</template>

<script>

export default {

data() {

return {

activeIndex:'1',

arr:[],

obj:

{

'1':[{src:'/static/basketball.jpg',name:'篮球',price:'50'},

{src:'/static/football.jpg',name:'足球',price:'60'},

{src:'/static/tennis.jpg',name:'乒乓球',price:'10'}

],

'2':[{src:'/static/corn.jpg',name:'玉米',price:'20'},

{src:'/static/husked.jpg',name:'大米',price:'30'},

{src:'/static/mung.jpg',name:'绿豆',price:'30'}],

'3':[{src:'/static/T.jpg',name:'t恤',price:'40'},

{src:'/static/coat.jpg',name:'大衣',price:'100'},

{src:'/static/sports.jpg',name:'运动服',price:'200'}]

}

}

},

methods: {

handleSelect(){

this.arr=this.obj[this.activeIndex]

}

}

}

</script>

<!-- Add "scoped" attribute to limit CSS to this component only -->

<style scoped>

h1,

h2 {

font-weight: normal;

}

ul {

list-style-type: none;

padding: 0;

}

li {

display: inline-block;

margin: 0 10px;

}

a {

color: #42b983;

}

.el-header {

background-color: #b3c0d1;

color: #333;

line-height: 60px;

}

.el-aside {

color: #333;

}

.el-select .el-input {

width: 130px;

}

.input-with-select .el-input-group\_\_prepend {

background-color: #fff;

}

.el-input {

width: 200px;

float: left;

margin-bottom: 10px;

}

.el-button {

float: left;

margin-right: 10px;

margin-bottom: 10px;

}

</style>

正则表达式

^a\nb\nc$