

OperAPI.vue

<template>

<!--显示内容-->

<div>

<div>数值为:{{ num }}</div>

<button @click="addNum">自增1</button>

</div>

</template>

<style>

/\*显示样式\*/

</style>

<script setup>

//脚本

import {ref} from 'vue'

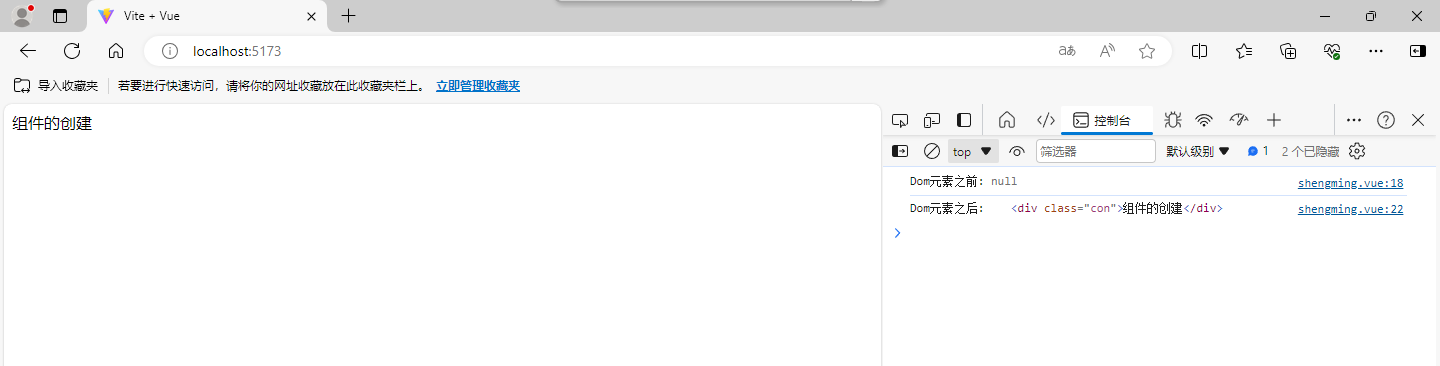
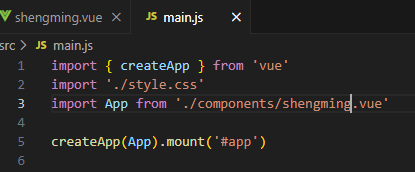
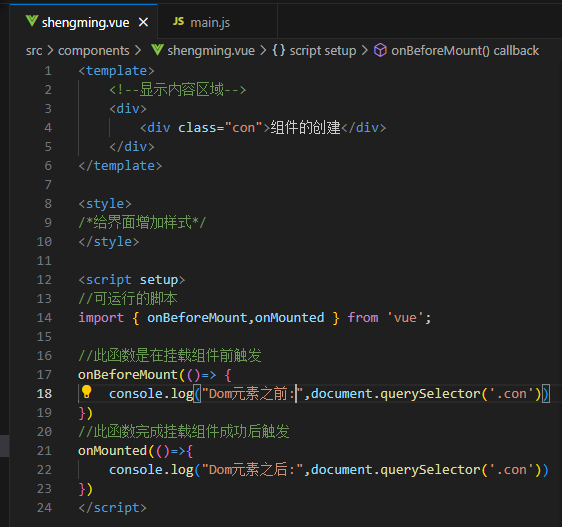
let num=ref(1)

const addNum=()=>{

num.value++

}

</script>



Shengming.vue

<template>

    <!--显示内容区域-->

    <div>

        <div class="con">组件的创建</div>

    </div>

</template>

<style>

/\*给界面增加样式\*/

</style>

<script setup>

//可运行的脚本

import { onBeforeMount,onMounted } from 'vue';

//此函数是在挂载组件前触发

onBeforeMount(()=> {

    console.log("Dom元素之前:",document.querySelector('.con'))

})

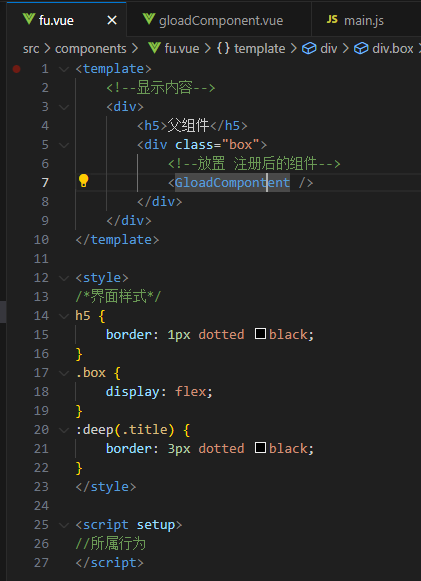
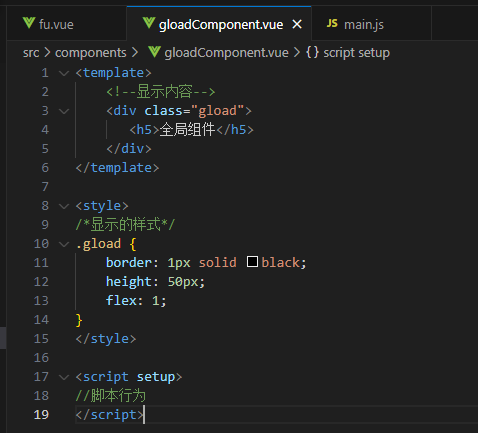
//此函数完成挂载组件成功后触发

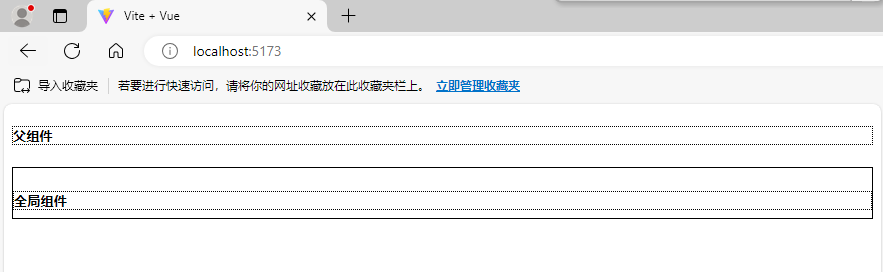
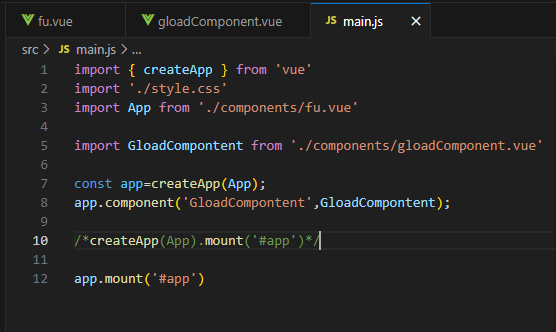
onMounted(()=>{

    console.log("Dom元素之后:",document.querySelector('.con'))

})

</script>





gloadComponent.vue

<template>

    <!--显示内容-->

    <div class="gload">

       <h5>全局组件</h5>

    </div>

</template>

<style>

/\*显示的样式\*/

.gload {

    border: 1px solid black;

    height: 50px;

    flex: 1;

}

</style>

<script setup>

//脚本行为

</script>

Fu.vue

<template>

    <!--显示内容-->

    <div>

        <h5>父组件</h5>

        <div class="box">

            <!--放置 注册后的组件-->

            <GloadCompontent />

        </div>

    </div>

</template>

<style>

/\*界面样式\*/

h5 {

    border: 1px dotted black;

}

.box {

    display: flex;

}

:deep(.title) {

    border: 3px dotted black;

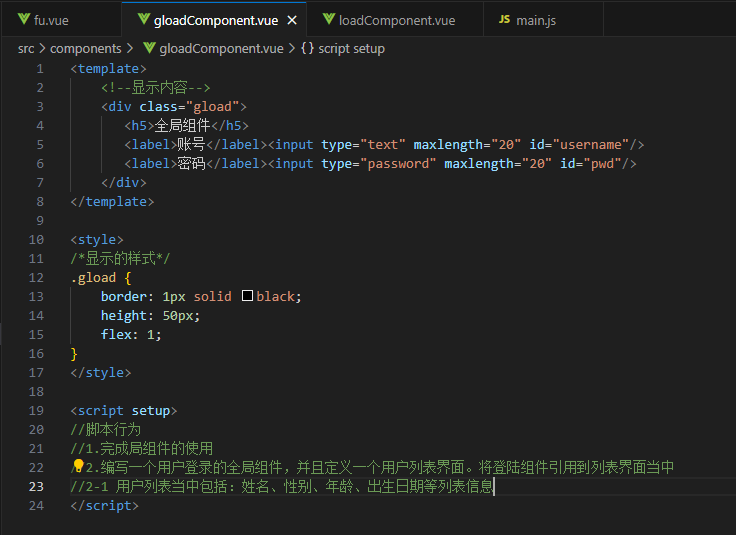
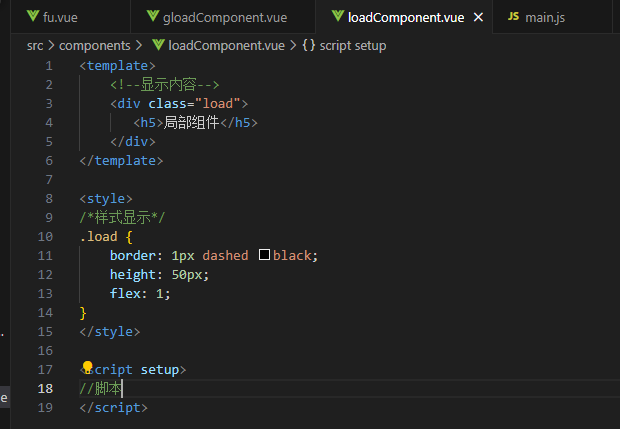
}

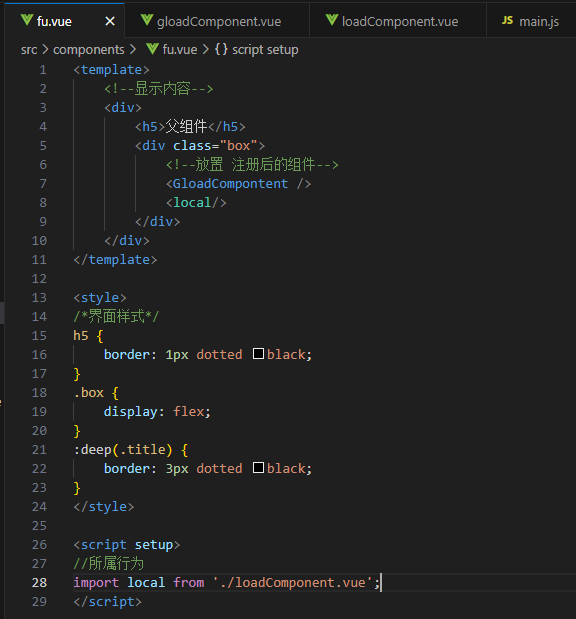
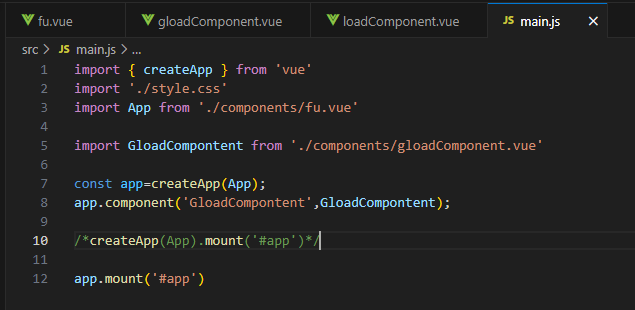
</style>

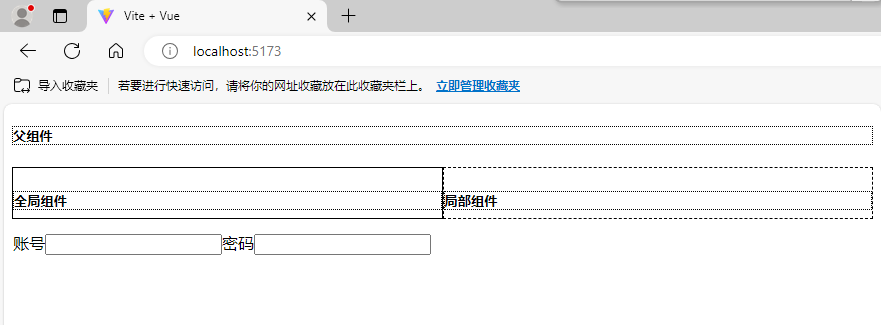
<script setup>

//所属行为

</script>







Fu.vue

<template>

    <!--显示内容-->

    <div>

        <h5>父组件</h5>

        <div class="box">

            <!--放置 注册后的组件-->

            <GloadCompontent />

            <local/>

        </div>

    </div>

</template>

<style>

/\*界面样式\*/

h5 {

    border: 1px dotted black;

}

.box {

    display: flex;

}

:deep(.title) {

    border: 3px dotted black;

}

</style>

<script setup>

//所属行为

import local from './loadComponent.vue';

</script>

gloadComponent.vue

<template>

    <!--显示内容-->

    <div class="gload">

       <h5>全局组件</h5>

       <label>账号</label><input type="text" maxlength="20" id="username"/>

       <label>密码</label><input type="password" maxlength="20" id="pwd"/>

    </div>

</template>

<style>

/\*显示的样式\*/

.gload {

    border: 1px solid black;

    height: 50px;

    flex: 1;

}

</style>

<script setup>

//脚本行为

//1.完成局组件的使用

//2.编写一个用户登录的全局组件，并且定义一个用户列表界面。将登陆组件引用到列表界面当中

//2-1 用户列表当中包括：姓名、性别、年龄、出生日期等列表信息

</script>

loadComponent.vue

<template>

    <!--显示内容-->

    <div class="load">

       <h5>局部组件</h5>

    </div>

</template>

<style>

/\*样式显示\*/

.load {

    border: 1px dashed black;

    height: 50px;

    flex: 1;

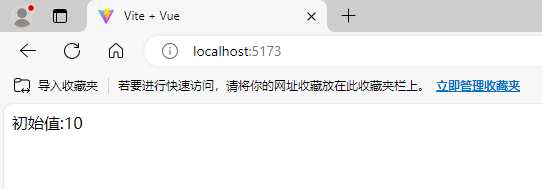
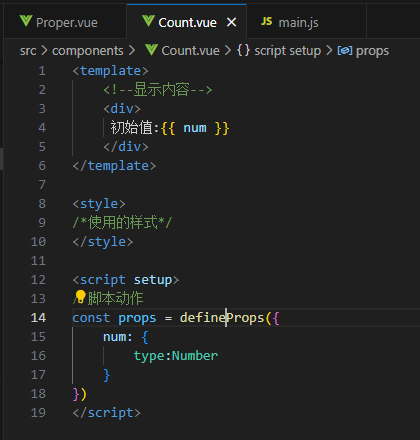
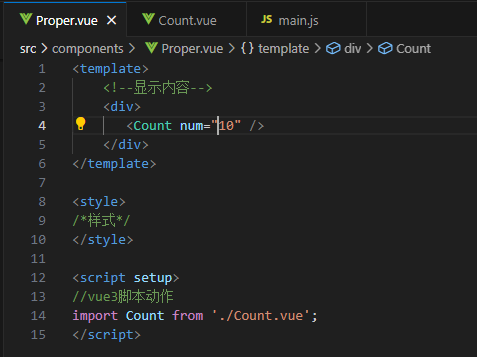
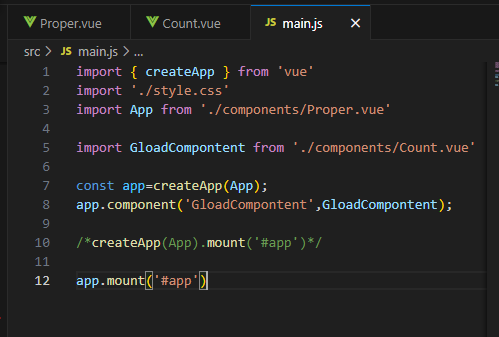
}

</style>

<script setup>

//脚本

</script>



Proper.vue

<template>

    <!--显示内容-->

    <div>

       <Count num="10" />

    </div>

</template>

<style>

/\*样式\*/

</style>

<script setup>

//vue3脚本动作

import Count from './Count.vue';

</script>

Count.vue

<template>

    <!--显示内容-->

    <div>

     初始值:{{ num }}

    </div>

</template>

<style>

/\*使用的样式\*/

</style>

<script setup>

//脚本动作

const props = defineProps({

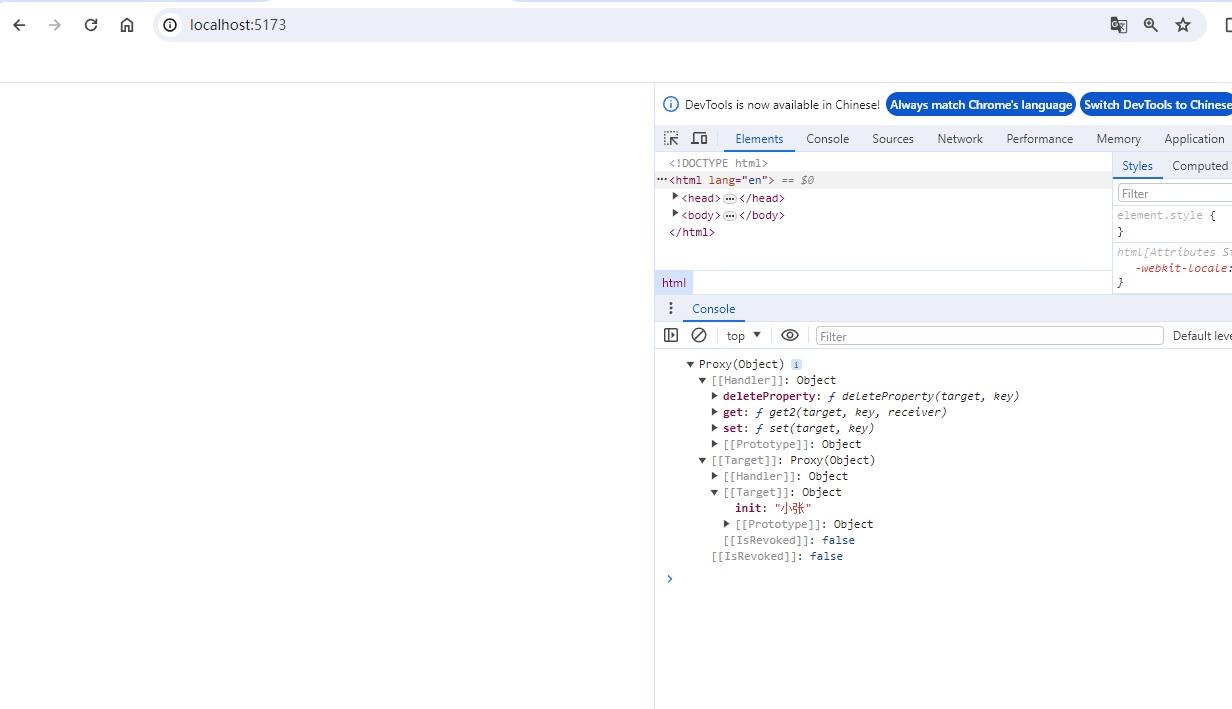
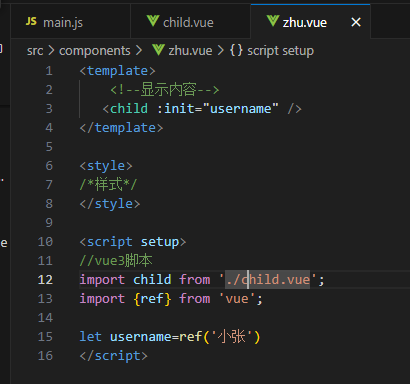
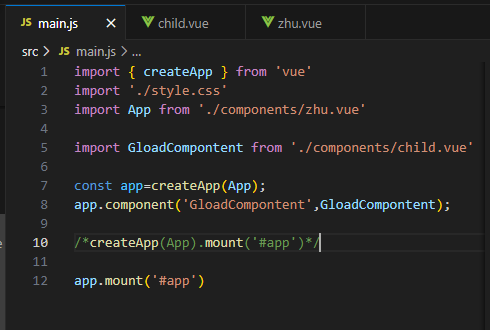
    num: {

        type:Number

    }

})

</script>



child.vue

<template>

    <!--显示内容-->

</template>

<style>

/\*样式\*/

</style>

<script setup>

//脚本

const props = defineProps(['init'])

console.log(props)

</script>

zhu.vue

<template>

    <!--显示内容-->

   <child :init="username" />

</template>

<style>

/\*样式\*/

</style>

<script setup>

//vue3脚本

import child from './child.vue';

import {ref} from 'vue';

let username=ref('小张')

</script>