1. build/webpack.base.conf.js:（红色为修改或者添加的地方）

'use strict'

const path = require('path')

const utils = require('./utils')

const config = require('../config')

const vueLoaderConfig = require('./vue-loader.conf')

function resolve(dir) {

return path.join(\_\_dirname, '..', dir)

}

var webpack = require("webpack")

module.exports = {

context: path.resolve(\_\_dirname, '../'),

entry: {

app: './src/main.js'

},

output: {

path: config.build.assetsRoot,

filename: '[name].js',

publicPath: process.env.NODE\_ENV === 'production' ?

config.build.assetsPublicPath : config.dev.assetsPublicPath

},

resolve: {

extensions: ['.js', '.vue', '.json'],

alias: {

'vue$': 'vue/dist/vue.esm.js',

'@': resolve('src'),

}

},

module: {

rules: [{

test: /\.vue$/,

loader: 'vue-loader',

options: vueLoaderConfig

},

{

test: /\.js$/,

loader: 'babel-loader',

include: [resolve('src'), resolve('test'), resolve('node\_modules/webpack-dev-server/client')]

},

{

test: /\.(png|jpe?g|gif|svg)(\?.\*)?$/,

loader: 'url-loader',

options: {

limit: 10000,

name: utils.assetsPath('img/[name].[hash:7].[ext]')

}

},

{

test: /\.(mp4|webm|ogg|mp3|wav|flac|aac)(\?.\*)?$/,

loader: 'url-loader',

options: {

limit: 10000,

name: utils.assetsPath('media/[name].[hash:7].[ext]')

}

},

{

test: /\.(woff2?|eot|ttf|otf)(\?.\*)?$/,

loader: 'url-loader',

options: {

limit: 10000,

name: utils.assetsPath('fonts/[name].[hash:7].[ext]')

}

}

]

},

node: {

// prevent webpack from injecting useless setImmediate polyfill because Vue

// source contains it (although only uses it if it's native).

setImmediate: false,

// prevent webpack from injecting mocks to Node native modules

// that does not make sense for the client

dgram: 'empty',

fs: 'empty',

net: 'empty',

tls: 'empty',

child\_process: 'empty'

},

plugins: [

new webpack.optimize.CommonsChunkPlugin('common.js'),

new webpack.ProvidePlugin({

jQuery: 'jquery',

$: 'jquery'

})

]

}

2.测试部分：

* 1. Test.vue:(红色为测试部分)

<template>

<div class="test">

<h2>{{title}}</h2>

<h3 v-if="showName">姓名：{{user.name}}</h3>

<h3 v-else="showName">地址：{{user.address}}</h3>

<h3>年龄：{{user.age}}</h3>

<ul>

    <li v-for="item in items">{{item.title}}</li>

    </ul>

<button v-on:click="greet()">Say Greeting</button>

<input type="text" v-on:keyup="pressKeybord" id="input"/>

</div>

</template>

<script>

//引入jquery

import $ from 'jquery'

export default {

name: "test",

data(){

return{

title:'Hello Word Vue.js!',

user:{

name:'zhuxu',

age:12,

address:'江苏南京',

},

showName:true,

items:[

{title:'item1'},

{title:'item2'},

{title:'item3'}

]

}

},

methods:{

greet:function(){

alert("Hello Word")

},

pressKeybord:function(){

var content=$("#input").val();

}

}

};

</script>

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

<!--scope只会影响到当前组件的样式-->

<style scoped>

</style>