

2

1.moment.js (js date) 日期格式化中文api

moment.js (js date) 日期格式化处理插件强大, moment.js中文api, 当前日期格式化、当前日期向前或者向后推的日期格式化、指定日期格式化调用演示

当前日期格式化

```
moment().format('MMMM Do YYYY, h:mm:ss a'); -> 四月 6日 2015, 3:55:57 下午
moment().format('dddd'); -> 星期一
moment().format("MMM Do YY"); -> 4月 6日 15
moment().format('YYYY [escaped] YYYY'); -> 2015 escaped 2015
moment().format(); -> 2015-04-06T15:55:57+08:00
-----
moment().format('YYYY-MM-DD'); -> 2015-04-06
moment().format('YYYY-MM-DD h:mm:ss a'); -> 2015-04-06 03:55:57 下午
```

指定日期格式化

```
moment("20111031", "YYYYMMDD").fromNow(); -> 3年前
moment("20120620", "YYYYMMDD").fromNow(); -> 3年前
moment().startOf('day').fromNow(); -> 16小时前
moment().endOf('day').fromNow(); -> 8小时内
moment().startOf('hour').fromNow(); -> 1小时前
```

当前日期向前或者向后推的日期格式化

```
moment().subtract(10, 'days').calendar(); -> 2015年3月27日
moment().subtract(6, 'days').calendar(); -> 上周二下午3点55
moment().subtract(3, 'days').calendar(); -> 上周五下午3点55
moment().subtract(1, 'days').calendar(); -> 昨天下午3点55
moment().calendar(); -> 今天下午3点55
moment().add(1, 'days').calendar(); -> 明天下午3点55
moment().add(3, 'days').calendar(); -> 本周四下午3点55
moment().add(10, 'days').calendar(); -> 2015年4月16日
```

也可以使用下面方式日期格式化

```
moment().format('L'); -> 2015-04-06
moment().format('l'); -> 2015-04-06
moment().format('LL'); -> 2015年4月6日
moment().format('ll'); -> 2015年4月6日
moment().format('LLL'); -> 2015年4月6日下午3点55
moment().format('lll'); -> 2015年4月6日下午3点55
moment().format('LLLL'); -> 2015年4月6日星期一下午3点55
moment().format('llll'); -> 2015年4月6日星期一下午3点55
```

2.上传图片

前端js代码

```

$(init);

function init() {
    var socket = io();

    socket.on('uploadProgress' , function(percent){
        console.log(percent);
        $(".pg-bar").progressbar( "option", "value", parseInt(percent));
        $(".pg-info").text( percent + '%');
    });
    // $("#defaultForm").validate({
    //     wrapper:"span",
    //     onfocusout:false,
    //     submitHandler:function(form) {
    //         doAddNews(); //验证成功则调用添加新闻函数
    //     }
    // })

    // $(".pg-bar").progressbar({value: 0});
    //
    // $(".pg-bar").progressbar( "option", "max", 100 );
    $("body").on('click', '#addNewsBtn', doAddNews);
    $("body").on('click', '#UploadBtn', doUpload);
    $("body").on('change', '#uploadFile', preUpload);
}

function preUpload() {
    $("#UploadBtn").removeClass('disabled');
}

function doUpload() {

    $(".pg-wrapper").show();

    var file = $("#uploadFile")[0].files[0];
    var form = new FormData();
    form.append("file", file);

    $.ajax({
        url: "/admin/uploadImg",
        type: "POST",
        data: form,
        async: true,
        processData: false,
        contentType: false,
        success: function(result) {
            startReq = false;
            if (result.code == 0) {

                var picUrl = $.format("[Alt text]({0})",result.data)
                $("#newsContent").insertAtCaret(picUrl);
                $(".pg-wrapper").hide();
                // console.log(result.data);
            }
        }
    });
}

```

后端Nodejs代码

```

var formidable = require('formidable');

//上传图片
router.post('/uploadImg', function(req, res, next) {
  console.log("开始上传");
  var io = global.io;

  var form = new formidable.IncomingForm();
  var path = "";
  var fields = [];

  form.encoding = 'utf-8'; //上传文件编码格式
  form.uploadDir = "public/uploadFile"; //上传文件保存路径（必须在public下新建）
  form.keepExtensions = true; //保持上传文件后缀
  form.maxFieldsSize = 30000 * 1024 * 1024; //上传文件格式最大值

  var uploadprogress = 0;
  console.log("start:upload----"+uploadprogress);

  form.parse(req);

  form.on('field', function(field, value) {
    console.log(field + ":" + value); //上传的参数数据
  })
  .on('file', function(field, file) {
    path = '\\' + file.path; //上传的文件数据
  })
  .on('progress', function(bytesReceived, bytesExpected) {

    uploadprogress = (bytesReceived / bytesExpected * 100).toFixed(0); //计算进度
    console.log("upload----"+ uploadprogress);
    io.sockets.in('sessionId').emit('uploadProgress', uploadprogress);
  })
  .on('end', function() {
    //上传完发送成功的json数据
    console.log('-> upload done\n');
    entries.code = 0;
    entries.data = path;
    res.writeHead(200, {
      'content-type': 'text/json'
    });
    res.end(JSON.stringify(entries));
  })
  .on("err",function(err){
    var callback="<script>alert('+err+')</script>";
    res.end(callback);//这段文本发回前端就会被同名的函数执行
  }).on("abort",function(){
    var callback="<script>alert('+ttt+')</script>";
    res.end(callback);
  });

});

module.exports = router;

```

27 分钟前(2016-07-27)