

layuiAdmin std v1.x 【iframe 常规版】 开发者文档

[更新日志](#)[常见问题](#)[在线演示](#)[切换到：单页面应用【专业版】文档](#)

layuiAdmin.std 是完全基于 layui 架构而成的通用型后台管理模板系统，采用传统的 iframe 多页面开发模式，可更快速直接地开发网页后台应用程序，无需过多地学习成本，简单高效，撸起袖子直接干。

题外

- 该文档适用于 **layuiAdmin.std 常规版 (iframe)**，阅读之前请务必确认是否与你使用的版本对应。
- 熟练掌握 layuiAdmin 的前提是熟练掌握 layui，因此除了本篇文档，[layui 的文档](#) 也是必不可少的存在。

快速上手

目录说明

解压文件后，你将看到以下目录：

- **src/**

layuiAdmin 源代码，通常用于开发环境（如本地）。

- **src/layuiadmin/**：layuiAdmin 的静态资源（JS、CSS、模块碎片等）

- ****src/layuiadmin/json/**：模拟接口返回的 JSON 相应信息，实际应用时可无视。

- ****src/layuiadmin/layui/**：layui 基础框架，功能一般领先于官网提供下载的版本。

- ****src/layuiadmin/lib/**：layuiAdmin 的核心模块，通常不建议修改。

- ****src/layuiadmin/modules/**：layuiAdmin 的业务模块

- ****src/layuiadmin/style/**：layuiAdmin 的样式文件

- ****src/layuiadmin/tpl/**：layuiAdmin 部分区块可能用到的模版碎片

- ****src/layuiadmin/config.js**：全局配置文件

- **src/views/**：layuiAdmin 提供的视图文件，你可以将其移动到你服务端工程的 view 层中，注意修改 JS/CSS 的路径即可。

- **dist/**

通过 gulp 将资源包的 src 目录的源代码进行构建后生成的目录（即：将 JS 和 CSS 文件进行了压缩等处理），通常用于线上环境。关于 gulp 的使用，下文也有介绍。

部署到服务端

1. 将 `src` 目录中的 `views` 文件夹整个复制到你 `服务端工程` 的 `view` 层中,
2. 确保可以访问后, 修改好 HTML 文件中的 JS/CSS 路径, 即可正常运行页面。

`iframe 常规版` 相比于 `单页面模式的专业版`, 无论是在目录结构还是开发模式上都要简单很多。因为单页版是接管了服务端 MVC 的视图层, 而 `iframe 版` 则将视图交给了服务端来控制 and 输出, 可以避免鉴权的复杂程度, 直接可衔接好新老项目 (因为你们的大部分老项目都是采用 `iframe 模式`)。

全局配置

当你已经顺利在本地预览了 `layuiAdmin` 后, 你一定迫不及待关注更深层的结构。假设你页面引入的是 `src` 目录的 JS, 你还需要关注的是 `src/layuiadmin/`

目录中的 `config.js`, 它里面存储着所有的默认配置。你可以按照实际需求选择性修改:

```
layui.define(['laytpl', 'layer', 'element', 'util'], function(exports){
  exports('setter', {
    container: 'LAY_app' //容器ID
    ,base: layui.cache.base //记录静态资源所在路径
    ,views: layui.cache.base + 'tpl/' //动态模板所在目录
    ,entry: 'index' //默认视图文件名
    ,engine: '.html' //视图文件后缀名
    ,pageTabs: true //是否开启页面选项卡功能。iframe 常规版推荐开启

    ,name: 'layuiAdmin'
    ,tableName: 'layuiAdmin' //本地存储表名
    ,MOD_NAME: 'admin' //模块事件名

    ,debug: true //是否开启调试模式。如开启, 接口异常时会抛出异常 URL 等信息

    //自定义请求字段
    ,request: {
      tokenName: false //自动携带 token 的字段名 (如: access_token)。可设置 false
      不携带。
    }

    //自定义响应字段
    ,response: {
      statusName: 'code' //数据状态的字段名称
      ,statusCode: {
        ok: 0 //数据状态一切正常的状态码
        ,logout: 1001 //登录状态失效的状态码
      }
      ,msgName: 'msg' //状态信息的字段名称
      ,dataName: 'data' //数据详情的字段名称
    }

    //扩展的第三方模块
    ,extend: [
      'echarts', //echarts 核心包
      'echartsTheme' //echarts 主题
    ]
  })
})
```

```
]

//主题配置
,theme: {
  //配色方案, 如果用户未设置主题, 第一个将作为默认
  color: [{
    main: '#20222A' //主题色
    ,selected: '#009688' //选中色
    ,logo: '' //logo区域背景色
    ,alias: 'default' //默认别名
  }] //为了减少篇幅, 更多主题此处不做列举, 可直接参考 config.js
}
});
});
```

基础方法

- **config 模块**

你可以在任何地方通过 `layui.setter` 得到 `config.js` 中的配置信息。如: `layui.setter.base`

- **admin 模块**

```
var admin = layui.admin;
```

- **admin.req(options)**

Ajax 请求, 用法同 `$.ajax(options)`, 只是该方法会进行错误处理

- **admin.screen()**

获取屏幕类型, 根据当前屏幕大小, 返回 0 - 3 的值

0: 低于768px的屏幕

1: 768px到992px之间的屏幕

2: 992px到1200px之间的屏幕

3: 高于1200px的屏幕

- **admin.sideFlexible(status)**

侧边伸缩。status 为 null: 收缩; status为“spread”: 展开

- **admin.on(eventName, callback)**

事件监听, 下文会有讲解

- **admin.popup(options)**

弹出一个 layuiAdmin 主题风格的 layer 层, 参数 options 跟 `layer.open(options)` 完全相同

- **admin.popupRight(options)**

在屏幕右侧呼出一个面板层。options 同上。

```
admin.popupRight({
  id: 'LAY-popup-right-new1' //定义唯一ID, 防止重复弹出
,success: function(){
  //将 views 目录下的某视图文件内容渲染给该面板
  layui.view(this.id).render('视图文件所在路径');
}
});
```

- **admin.resize(callback)**

窗口 resize 事件处理, 我们推荐你使用该方法取代 jQuery 的 resize 事件, 以避免多页面标签下可能存在的冲突。

- **admin.events**

- admin.events.refresh()
刷新当前右侧区域
- admin.events.closeThisTabs()
关闭当前标签页
- admin.events.closeOtherTabs()
关闭其它标签页
- admin.events.closeAllTabs()
关闭全部标签页

- **view 模块**

用法同 专业版: <http://fly.layui.com/docs/5/#base-view>

实用组件

Hover 提示层

通过对元素设置 `lay-tips="提示内容"` 来开启一个 hover 提示, 如:

```
<i class="layui-icon layui-icon-tips" lay-tips="要支持的噢" lay-offset="5">
</i>
```

其中 `lay-offset` 用于定于水平偏移距离 (单位px), 以调整箭头让其对准元素

兼容性

layuiAdmin 使用到了 layui 的栅格系统, 而栅格则是基于浏览器的媒体查询。ie8、9不支持。

所以要在宿主页面 (如 start/index.html) 加上下面这段保证兼容:

```
<!-- 让IE8/9支持媒体查询, 从而兼容栅格 -->
<!--[if lt IE 9]>
```

```
<script src="https://cdn.staticfile.org/html5shiv/r29/html5.min.js">
</script>
<script src="https://cdn.staticfile.org/respond.js/1.4.2/respond.min.js">
</script>
<![endif]-->
```

升级事项

从官网更新资源包后，以 src 目录为例（dist 由于是 src 构建后生成的目录，所以本质是和 src 一样的）
src 目录下可以直接覆盖的有：

- src/json/
- src/layui/
- src/lib/
- src/style/

需要灵活调配的有：

- src/modules/
 - src/tpl/
 - src/config.js
- 如果没有改动默认配置，事实上 config.js 也可以覆盖升级

源码构建

当你在 src 目录完成开发后，你可通过 gulp 对 src 源码进行自动化构建，以生成用于线上环境的 dist 目录。并在 HTML 源代码中引入该目录的 JS/CSS。

在资源包中根目录下看到的 gulpfile.js 是 layuiAdmin 写好的任务脚本，package.json 是任务配置文件，你只需按照以下步骤：

1. step1：确保你的电脑已经安装好了 [Node.js](#)，如果未安装，可去官网下载安装
2. step2：命令行安装 gulp：npm install gulp -g
3. step3：切换到 layuiAdmin 项目根目录（即 gulpfile.js 所在目录），命令行安装任务所依赖的包：npm install

安装完成后，后续只需直接执行命令：`gulp` 即可完成 src 到 dist 目录的构建

关于版权

layuiAdmin 受国家计算机软件著作权保护，未经官网正规渠道授权擅自公开产品源文件、以及直接对产品二次出售的，我们将追究相应的法律责任。

© 2018 layui.com 版权所有