

云朵数屏用户手册

目录

产品介绍	3
快速开始	3
术语说明	3
创建一个项目	4
编辑项目	5
预览项目	6
下载与应用	6
界面说明	7
项目管理	7
项目编辑	7
项目预览	8
创建项目	8
新建空项目	9
复制模板	9
复制已有项目	10
导入项目	10
项目设置	11
编辑项目	11
舞台设置	11
舞台尺寸	12
背景设置	12
网络设置	12
辅助设置	13
页面设置	13
新建页面	14
修改页面	14
删除页面	14
元素维护	14
元素添加	15
位置状态	15
外观设置	15
动效设置	16
配置数据	17
元素编组	17
内置组件	18
echarts 图表	19
图片素材	21
视频素材	21

文本内容	22
图形	22
盒子	22
选项卡	23
数据表格	23
iframe 框架	24
数据设置	24
添加数据	25
复制数据	25
应用数据	25
注意事项	25
事件动作	26
添加动作	26
动作类型	27
参数设置	28
辅助设计	28
排版对齐	28
复制黏贴	29
运行项目	29
项目预览	29
下载项目	30
运行项目	30
rdPlay 播放器	31
自定义	31

产品介绍

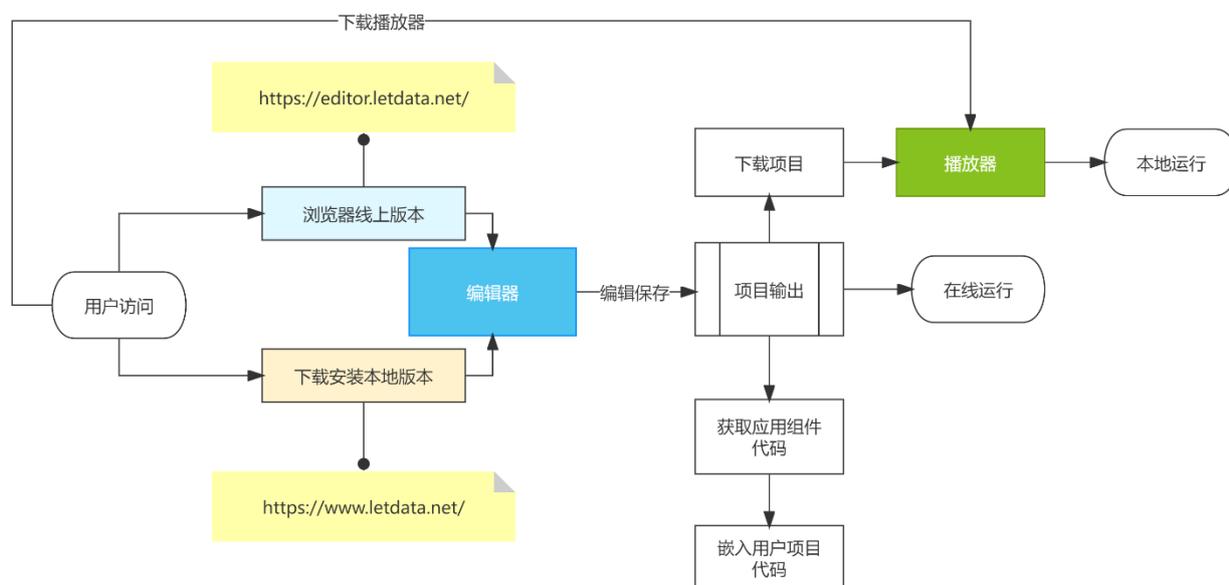
云朵数屏(LetData)由数屏编辑器 rdEditor，播放器 rdPlay，运行时 rdRuntime 几块组成，帮助用户创建丰富的信息数据可视化交互系统。

- **图形化**：使用图形化界面通过拖拉拽轻松搭建出专业的数据可视化应用。
- **多领域**：创建各类信息内容，无论是数据报表，大屏驾驶舱，车间看板，监控直播，3D 模型展示，操作终端等都可应对。
- **数据**：还有你编辑维护的内容**数据存储本地化**的，是否**云端同步**完全可以自主控制。
- **在线**：创建的内容可以直接**线上发布**，也可以通过 rdPlay 运行。
- **扩展**：可以可以基于 vue3 的基础上开发自己的**自定义组件**内容。
- **私有**：项目运行可**独立部署**私有环境，**无使用限制条件**。

更多信息可查看在新文档 <https://doc.letdata.net>

快速开始

云朵数屏（letData）默认存储是用户本机，如需要在线保存数据，需要开通会员服务。
rdEditor 编辑项目，rdPlay 运行项目，rdRuntime 扩展插件开发，具体流程如下：



术语说明

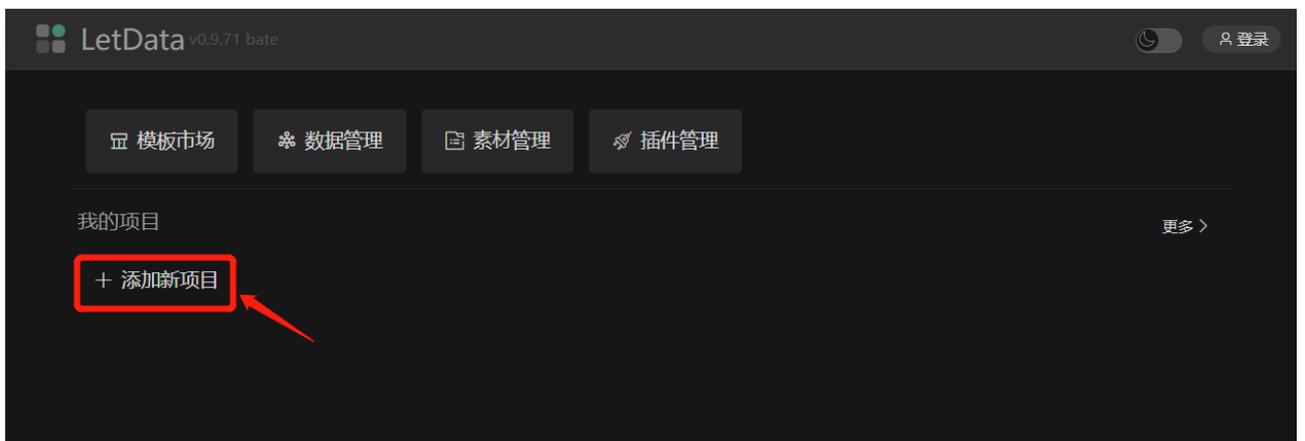
- **应用**：指的是项目编辑保存后，实际使用时播放运行的程序

- 舞台：所有元素需要显示到页面的对象
- 组件：指的是框架内置或用户开发的组件（当前只提供 vue 依赖的组件开发）
- 元素：指的是组件添加到舞台上后的对象
- 动作：系统内置或用户自定义需要操作控制目标内容的程序
- 事件：系统或元素与用户交互相关的响应

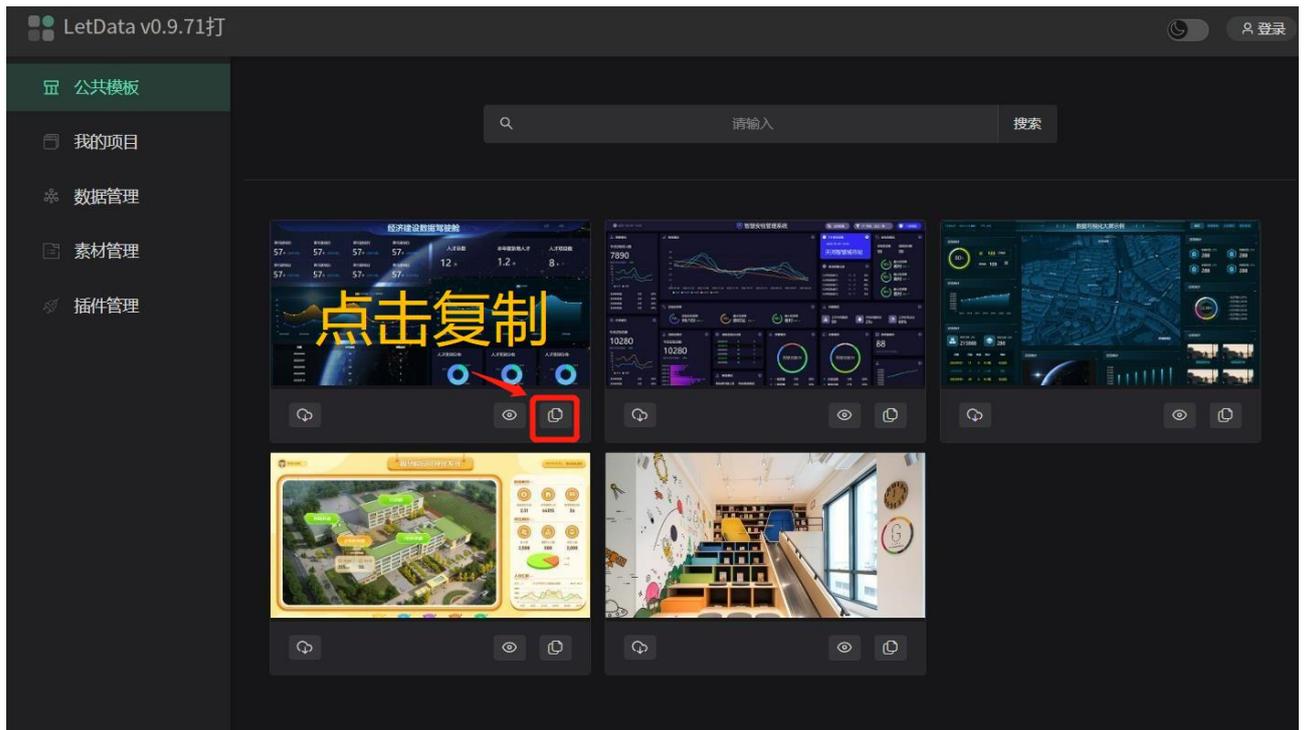
创建一个项目

创建项目没有任何过多的配置与流程，只需要一键完成项目的创建，具体操作有以下两种方式

1. 新建空白的项目



2. 复制已有项目（或模板项目）

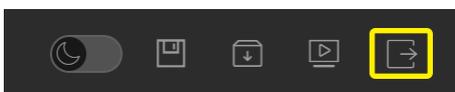


编辑项目

在编辑界面，通过基本的拖拉拽完成项目编辑



退出编辑



保存数据

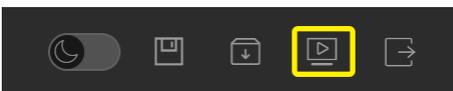


下载项目



预览项目

在编辑页面点击右上角预览按钮



项目管理界面，点击项目预览按钮



下载与应用

1. 点击项目下载按钮，将项目下载到本地



2. 通过在线窗口运行项目

运行地址: <https://run.letdata.net>

也可以下载 rdPlay 播放器，将项目文件放置其中运行

界面说明

编辑器分为管理窗口、编辑窗口、预览窗口三处主要的模块。

- 管理窗口：主要围绕项目相关信息的维护，包含数据，插件，素材等，同时也是开通线上账号的地方。
- 编辑窗口：项目编辑核心界面。
- 预览窗口：项目查看的独立界面，此界面是项目最终生成后运行结果界面。

项目管理

项目管理界面分三个区域，分别是：栏目导航、栏目内容和用户设置



项目编辑

编辑界面分为四个区域，分别是：内容面板、设置面板、编辑操作工具栏、编辑舞台



项目预览

项目预览是完全独立的一个项目运行窗口，此窗口无操作控制相关内容



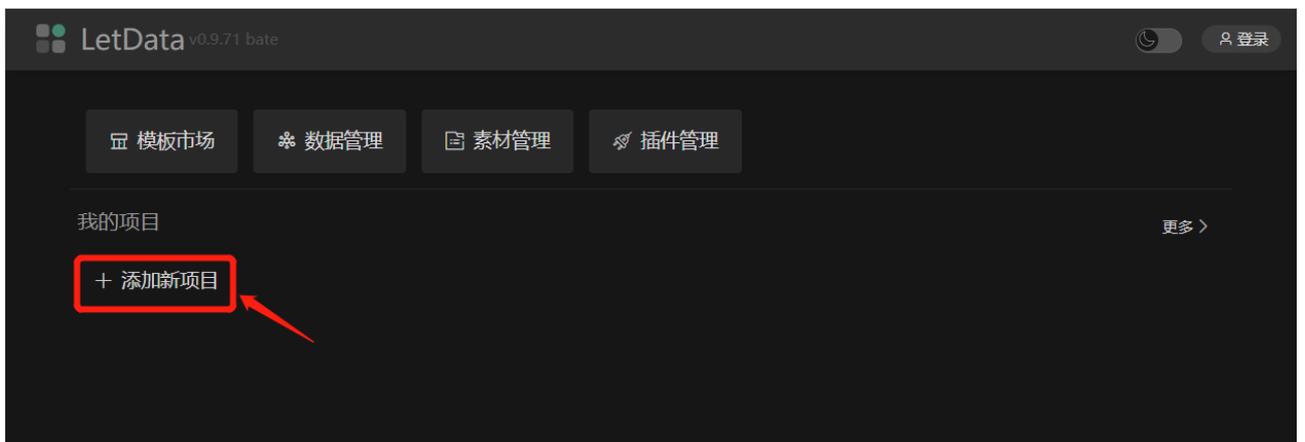
创建项目

项目的创建一共有四种方式

1. 完全新建空项目
2. 创建一个来自模板的项目
3. 创建一个来自已有项目的项目
4. 导入一个新项目

新建空项目

- 通过桌面“添加新项目”按钮
系统桌面 -> 点击添加新项目按钮

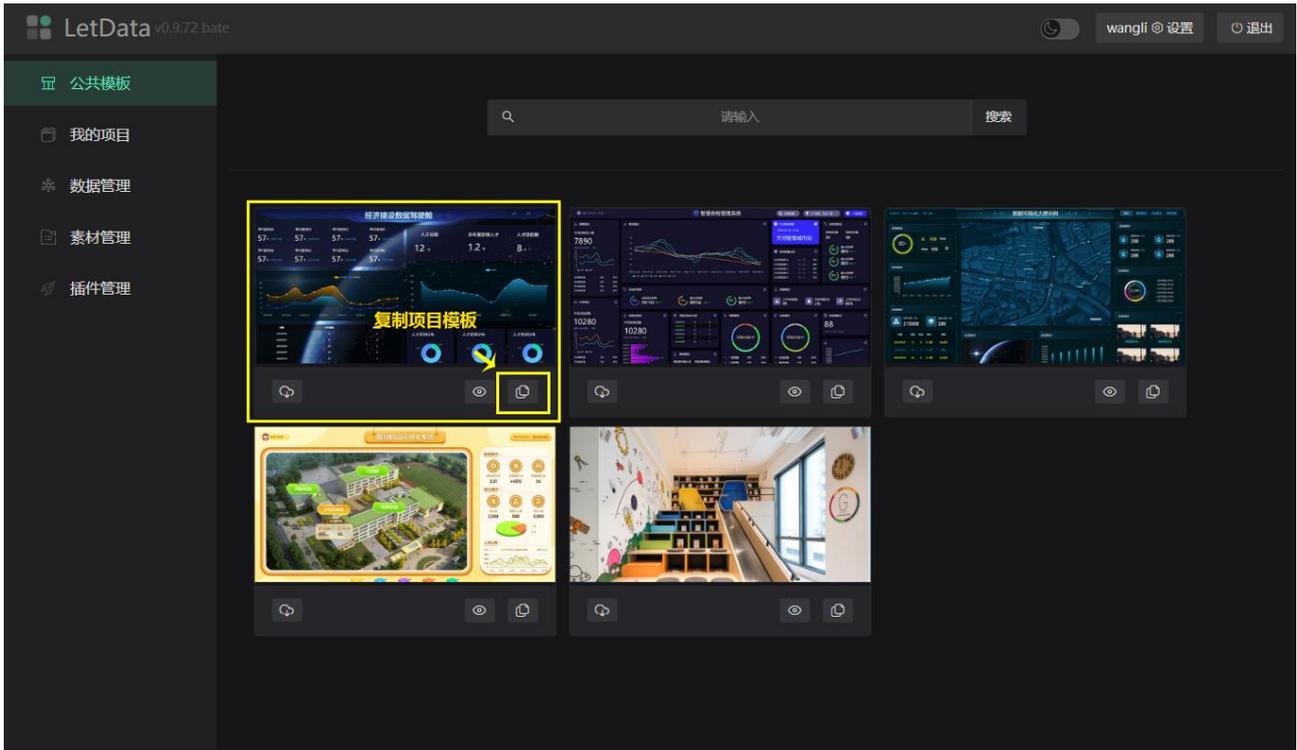


- 项目管理界面“新建项目”按钮
项目管理 -> 点击新建项目按钮



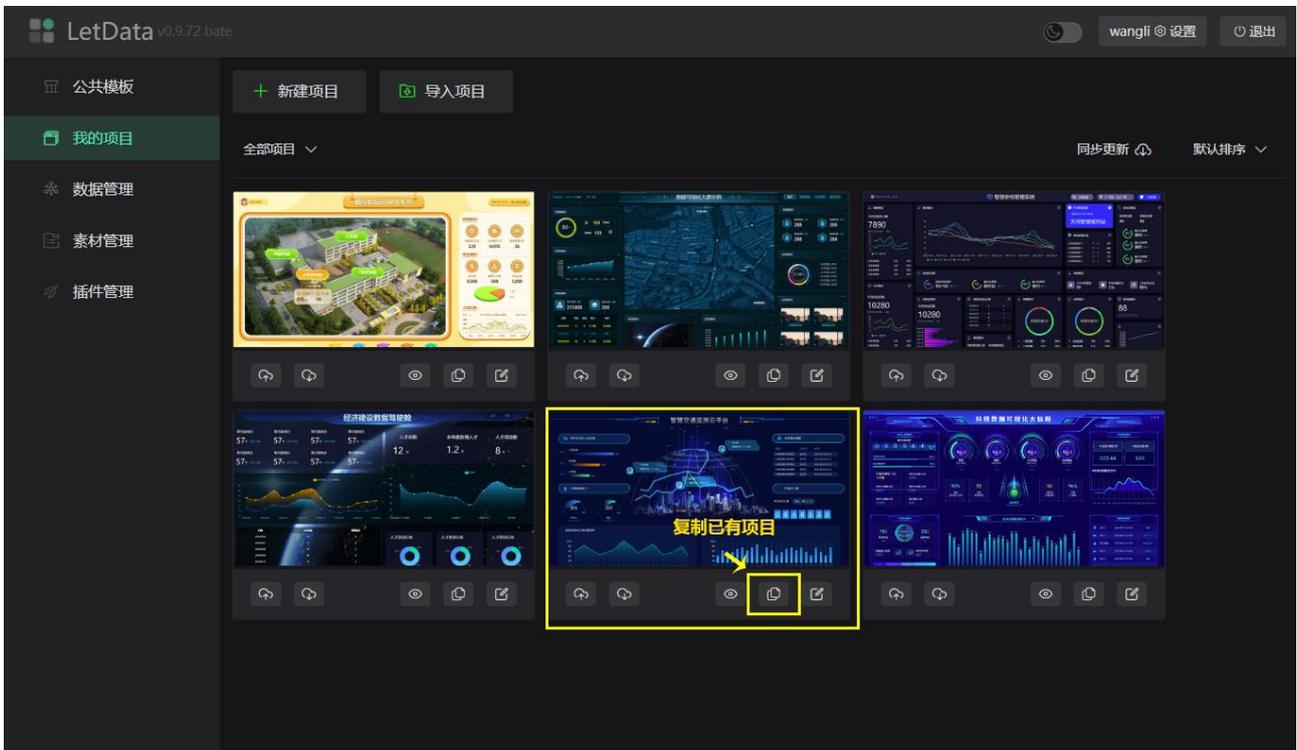
复制模板

公共模板 -> 选择模板-> 点击复制模板按钮



复制已有项目

我的项目 -> 选择已有项目-> 点击复制项目按钮



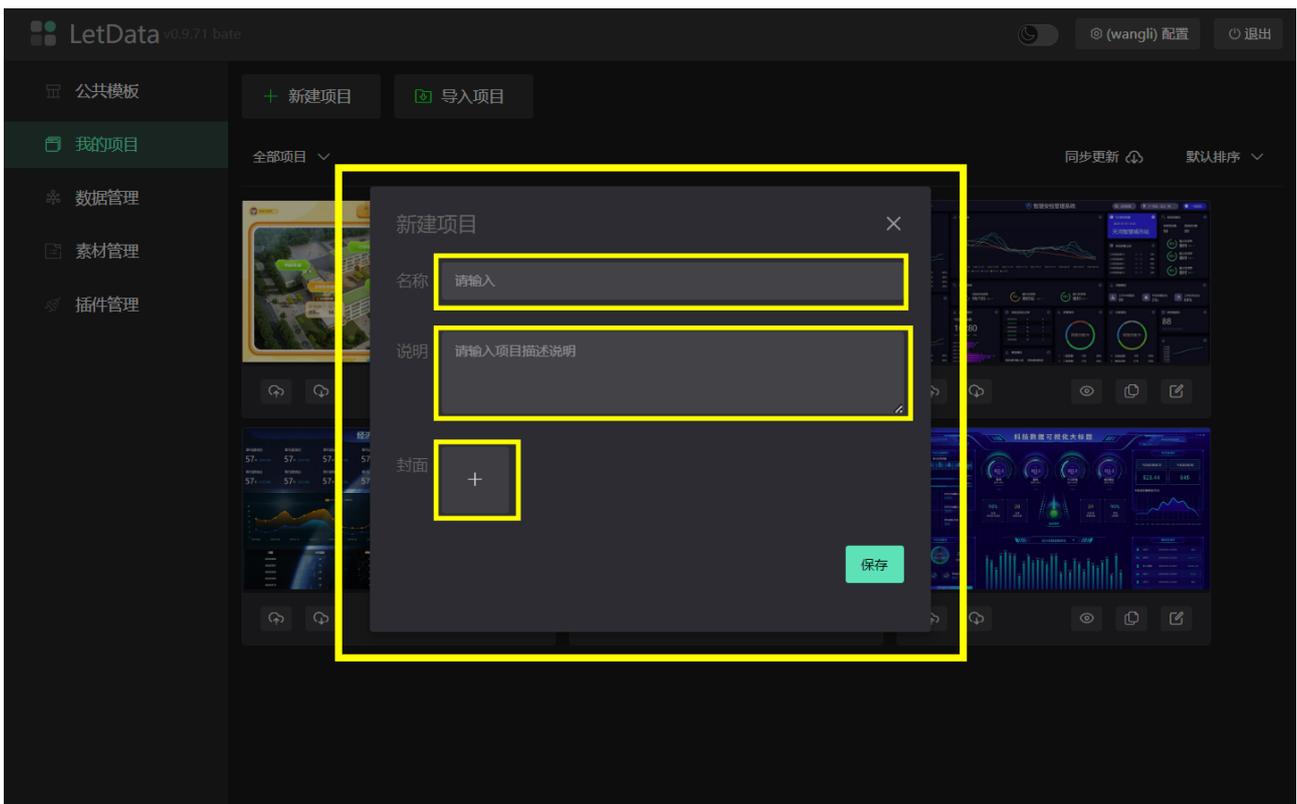
导入项目

我的项目 -> 点击导入项目按钮



项目设置

项目默认有三个参数可以设置，分别是：名称 说明 封面



编辑项目

舞台设置

一个项目只有一个编辑舞台，所有的页面和内容都在一个舞台上呈现，舞台相关设置内容是固定提前配置，除了引用的数据之外，不可动态变更。

主要包含：舞台尺寸 背景设置 网络设置 辅助设置 四项设置



舞台尺寸

舞台尺寸也就是项目应用实际呈现的尺寸大小，初始化的尺寸为 1920*1080 应用

- w: 宽度
- h: 高度
- s: 缩放模式 自动 填满 无（默认自动）

缩放自动的情况下：界面会根据实际的宽高比进行同比例的缩放，填满标识根据外部界面尺寸做满屏（会存在拉伸），无表示无缩放操作

背景设置

舞台整体的背景设置，可以通过颜色或图片配置
图片内容可以设置缩放形式，和平铺重复形式

网络设置

项目应用默认接口请求的配置

- 主机：服务器 host 地址
- 方法：get 与 post

- 头部: header 信息, 一个数组, 值以 key:Value 形式表现

辅助设置

- 配色: 项目编辑时默认颜色的快速选择

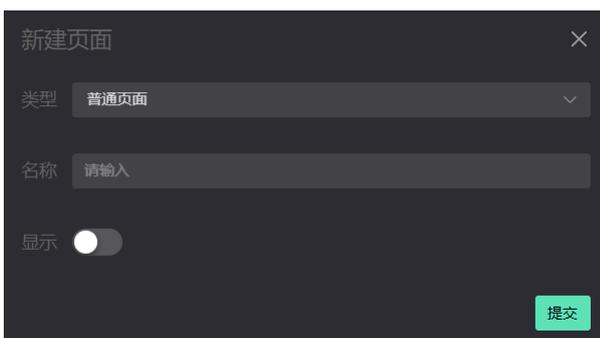


页面设置

项目的页面可以称为 项目模块 从项目编辑来说, 可以把一个模块做作为一个整体切换的页面。

每一个项目必须拥有一个 主页 并且不可以替换和删除。 页面的设置主要是三个参数:

- 类型: 页面的三种类型 (普通 固定 弹窗)
 - 普通表示通用的正常可切换显示的页面
 - 固定表示此页面模块将一直显示不会被隐藏
 - 弹窗表示此页面浮在其它页面上面, 显示的时候会有一个遮罩层。
- 名称: 页面管理区显示的名称
- 显示: 表示是否默认显示



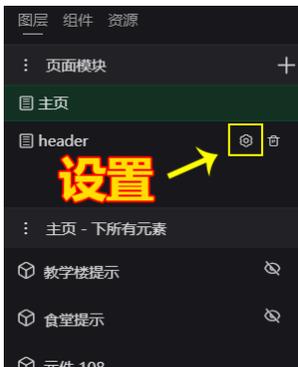
新建页面

内容面板 -> 点击新建+号按钮



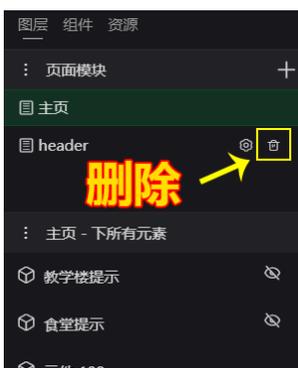
修改页面

内容面板 -> 页面列表项 -> 点击设置按钮



删除页面

内容面板 -> 页面列表项 -> 点击删除按钮



元素维护

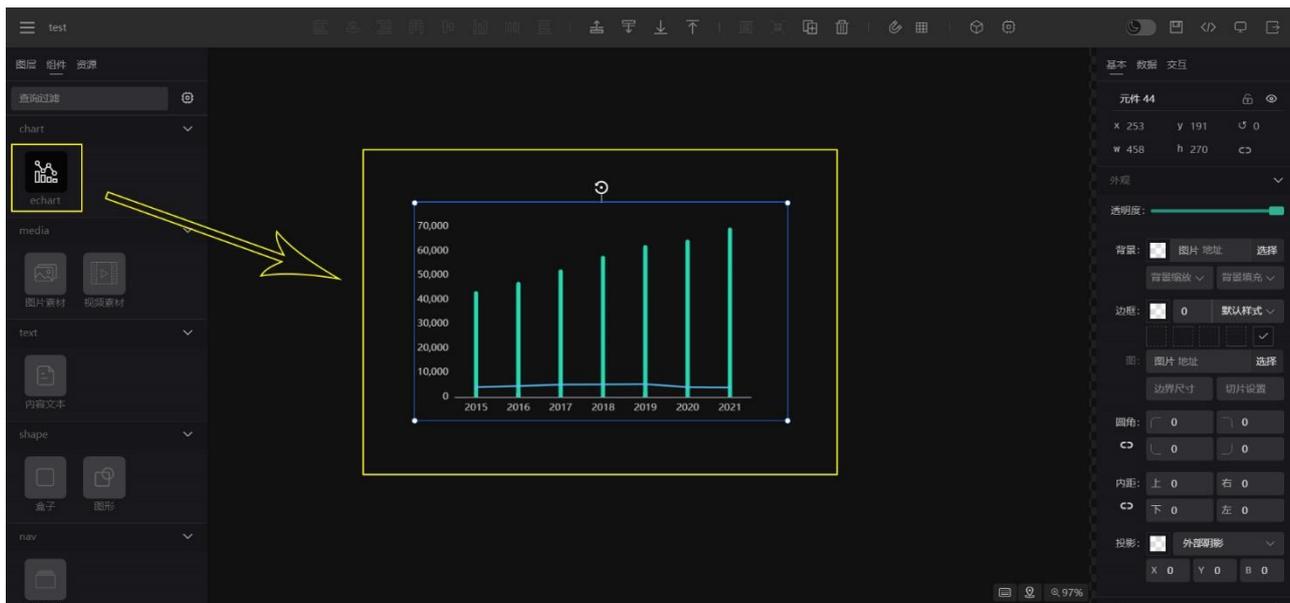
所有元素都来自用户的组件库，用户组件库是由系统自带的组件与用户自己开发组件组成。编辑的项目依赖用户 元素可以通过拖入，复制黏贴等方式添加，元素的尺寸编辑也是通过

拖拽完成，另外对齐、层级、吸附等特性也可通过相关动作指令完成。任何元素都包括的属性功能如下：

- 基本特性：位置状态、外观、动效
- 基本能力：数据绑定、事件交互

元素添加

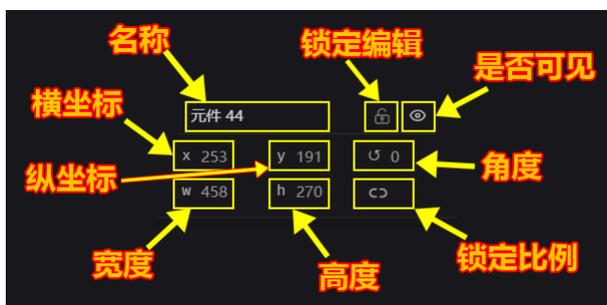
内容面板 -> 组件 -> 拖拽组件图标到舞台



舞台上的元素选中之后，可以直接拖拽移动，也可以通过选中的蓝色边框和四个点进行拖拽缩放尺寸。

位置状态

横坐标、纵坐标、宽度、高度、角度、同比例、锁编辑、是否可见



外观设置

透明度、背景、边框、圆角、内边距、投影



- 透明度：拖动滑杆调整，范围 0 至 100，默认为 100
- 边框：颜色、粗细、样式、四边
边框还有一种特殊的设置 slice 切片方式，切片方式需要在边框设置有宽度且为实体时生效
 - 边界图片：需要绘制的图片
 - 边界设置：具体设置参考 [borderImageSlice 设置](#)
 - 边界宽度：具体设置参考 [borderImageWidth 设置](#)
- 圆角：主要应用于边框、背景色、阴影
- 内距：元素的内边距，主要与文本相关
- 投影：分为内阴影与外阴影

动效设置

元素的动画效果



每一个动效都有四个基本参数

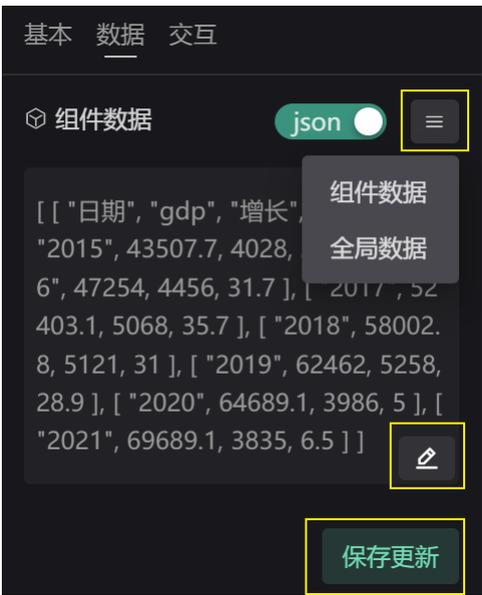
- 延时：元素显示之后延时执行动画

- 时长：动画时长，单位为秒
- 循环：动画执行次数，默认 1 次
- 状态：动画执行后的状态，默认情况将自动结束，停留状态表示，动画最后一帧的样子

配置数据

任何组件都可以通过两种方式配置数据

- 组件数据：如果是组件数据，那这个编辑的数据只能是静态的数据，是配置到项目内，并只和此组件相关，可以通过点击编辑按钮编辑
- 全局数据：全局数据是绑定项目设置的数据信息内容，[查看数据设置](#)。

组件数据	全局数据
	

元素编组

对舞台上的元素多选，可以通过编组按钮进行编组，编组后的元素可以作为一个单元进行拖动，也可以设置动效，设置整体的外观样式：

操作按钮	编组选中	整体样式
------	------	------



编组后的内部元素，可以通过双击组，选中对应的元素，此时的元素不能进行多拽操作，但可以通过属性面板设置其相关信息。如果比较难双击选中的元素，可以通过左侧元素列表，展开组信息单独选中。

内置组件

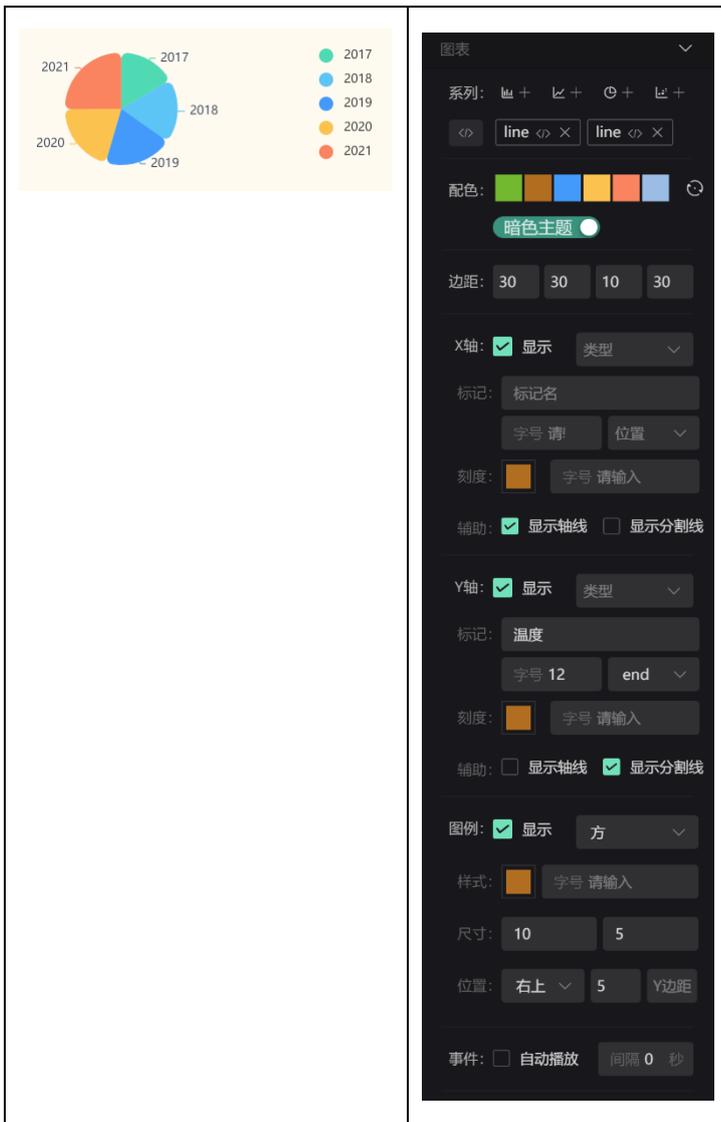
当前系统默认拥有九常用标准组件，分别是：echart 图表、图片素材、视频素材、内容文本、样式盒子、tab 选项卡、table 表格、iframe 容器 每个组件拥有不同的属性



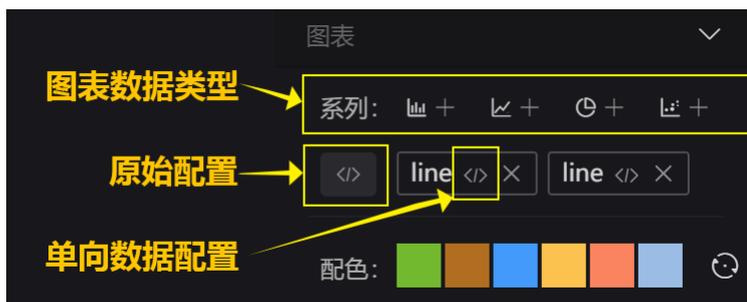
echarts 图表

此组件依赖 echarts 库，同时默认只提供了四种图表类型 line、bar、pie、scatter，更多其它用户可以自行开发对应的组件。echarts 图表属性面板提供了常用调节的参数，如果需要完成的配置可以参考 [echarts 官方文档](#)。从属性优先级来说，原始配置是最低层级，项目元素的设置会覆盖原始配置相同的参数值。

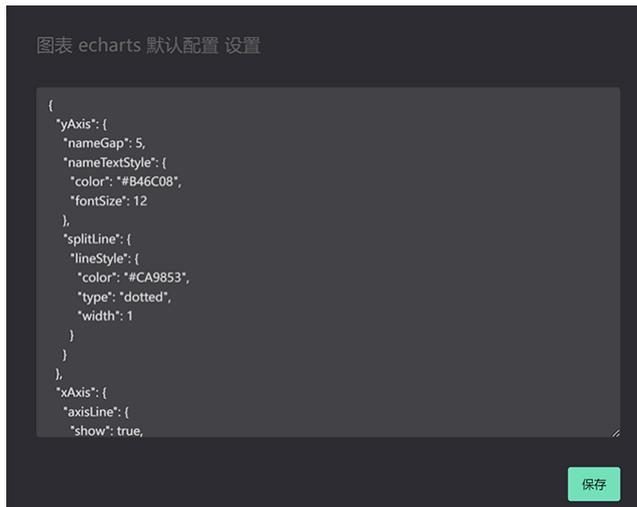
示例	属性
----	----



- **配色：**遵循 echarts 的色盘配置
- **边距：**绘制的内边界
- **x 轴：**包含类型、标记、字号、位置、刻度色、刻度字号、是否有辅助线、是否有分割线
- **y 轴：**同轴相同属性
- **图例：**形状、颜色、字号、图形尺寸、位置
- **事件：**是否自动播放，播放间隔时间
- **原始配置**



点击配置按钮弹出原始配置的文本编辑框



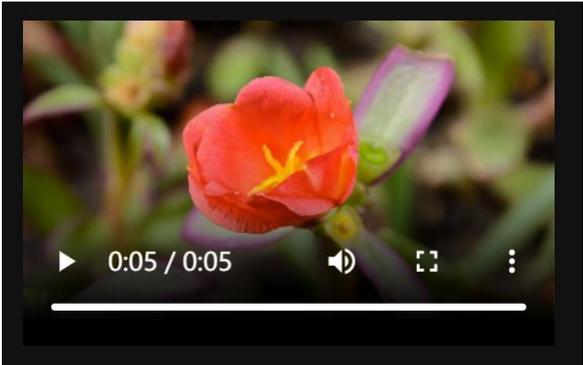
图片素材

- 平铺方式：自动、裁切铺满、完整显示、100%
- 对齐方式：左上、左中、左下、右上、右中、右下、中上、居中、中下

示例	属性
	<p>图片</p> <p>样式： 自动</p> <p>左中</p>

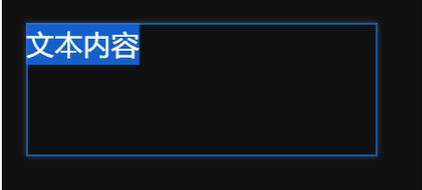
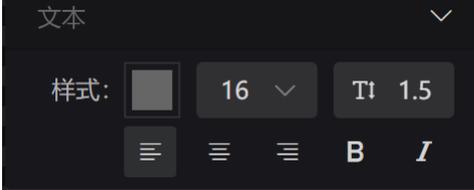
视频素材

是否自动播放、是否循环播放、是否显示控件

示例	属性
	<p>视频</p> <p>配置：<input checked="" type="checkbox"/> 自动播放 <input type="checkbox"/> 循环播放</p> <p><input type="checkbox"/> 显示控件</p>

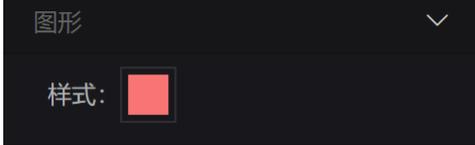
文本内容

样式：颜色、字号、行高、对齐、粗细、斜体

双击文本编辑示例	属性
	

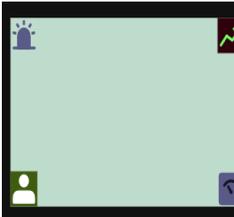
图形

图形组件主要用于显示 svg 格式的图形内容，非此格式字符串将不会显示
样式：颜色

示例	属性
	

盒子

盒子组件可以装入图片相关的内容，一般配合外观属性，制作出丰富的底图样式。
设置的每一项都一个和背景图相关的内容，具体属性有：
名称、横坐标、纵坐标、宽度、高度、背景色、背景图、缩放模式、重复情况

示例	属性	添加元素
		

选项卡

选项卡组件常用于页面的导航按钮

- 默认状态：颜色、字号、对齐方式、边界、圆角、背景色、背景图
- 选中状态：同默认属性
- icon：图标特定高度位置

示例	属性
	

数据表格

表格组件可满足常见表格数据展示

- 整体：颜色、字号、行数、行高、滚动行数
- 尺寸：这里的尺寸是定义每一列的宽度，使用都好分割
- 表头：颜色、字号、背景色
- 斑马：颜色、字号、背景色

示例	属性
----	----

日期	白天	夜晚
20230413	23	16
20230414	26	13
20230415	23	11

列表

整体: 字号 0

行数 3 行高 30 滚动 3

尺寸: 列宽 使用逗号,分割

列对齐 使用逗号,分割

表头: 显示

字号 0

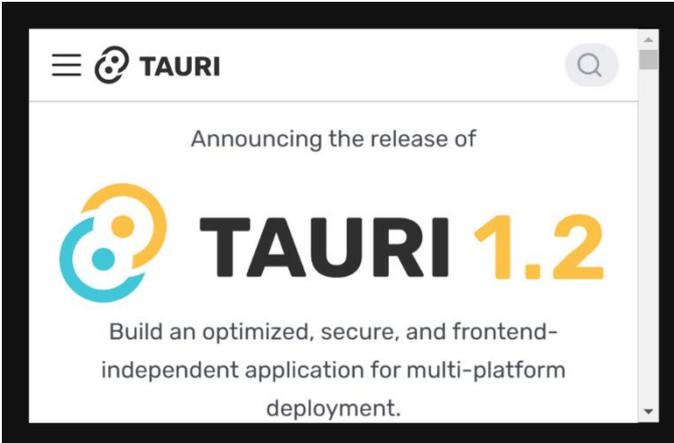
斑马: 显示

字号 0

iframe 框架

外部页面容器

- 权限: 自动播放、摄像头、麦克风、地理位置、全屏
- 策略: [参考 iframe 文档](#)

示例	属性
	<div style="border: 1px solid black; padding: 5px; background-color: #333; color: #eee;"> <p>框架</p> <p>安全: 可控权限</p> <p>内容安全策略</p> </div>

数据设置

项目的数据与项目是一体的，即时从外部复制过来，也是独立的一份。

数据主要有 原始数据、远程数据两种来源，添加与维护在编辑器右侧的数据面板上操作。

- 原始数据：表示写入项目配置文件内的数据信息，配置好后，在项目运行过程中是不会改变的。
- 远程数据：数据源来自显示接口返回的数据

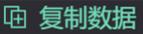
添加数据



具体的数据编辑操作可以查看 [数据管理](#)

复制数据

数据可以通过外部复制数据（这里的数据信息来自系统的数据管理）

点击 复制数据  按钮，通过参床选择 需要复制的数据到当前项目。



应用数据

具体的数据应用 可以在元素和系统时间种使用 [元素配置数据](#)

注意事项

- 添加后的数据，其来源不建议更改，更改后原数据信息将不存在，如果应用了数据，将造成信息都是错误。

- 数据删除，需要注意是否应用了，如果应用建议先删除相关的数据绑定在删除数据源。
- 原始数据的数据类型的更改，并不会自动调整数据内容格式，需要用户自己手动重新编辑。

事件动作

项目的用户交互主要由：事件、动作两部分组成。

- 事件：分为应用事件、元素事件、用户事件
 - 应用事件，项目启动后自主事件（应用启动、应用定时器）
 - 元素事件，元素显示后（定时任务、延迟任务），内部自定义事件
 - 用户事件，主要指的用户操作元素的事件（当前只有一个点击事件）
- 动作：当前系统内置（显示元素、开关元素、数据传递、模块切换、开关弹窗、新开窗口）与用户自定义动作

相关操作在 动作面板 、 交互面板 中完成



添加动作

任何元素都具备（点击、定时、延迟）三个事件，添加方式如下：

选中元素 -> 进入交互面板 -> 点击添加动作 -> 选择添加的动作并确认

勾选要添加 **点击** 事件动作 ✕

隐藏教学楼
 切换内容显示
 统计详情

打开用户详情窗口

动作类型

动作	目标	多项	参数
显示隐藏	元素	是	<div style="background-color: #333; color: white; padding: 5px; border-radius: 5px;"> <p>名称: <input type="text" value="显示元素"/></p> <p>动作: <input style="border-bottom: 1px solid #ccc;" type="text" value="显示隐藏"/></p> <p>元件: <input style="border-bottom: 1px solid #ccc;" type="text" value="组合1"/></p> <p>设置: <input checked="" type="checkbox"/> 显示</p> </div>
显示开关	元素	是	<div style="background-color: #333; color: white; padding: 5px; border-radius: 5px;"> <p>名称: <input type="text" value="开关"/></p> <p>动作: <input style="border-bottom: 1px solid #ccc;" type="text" value="显示开关"/></p> <p>元件: <input style="border-bottom: 1px solid #ccc;" type="text" value="组合2_c_c"/> <input style="border-bottom: 1px solid #ccc;" type="text" value="组合2"/> <input style="border-bottom: 1px solid #ccc;" type="button" value="+1"/></p> </div>
发送数据	元素	是	<div style="background-color: #333; color: white; padding: 5px; border-radius: 5px;"> <p>名称: <input type="text" value="数据"/></p> <p>动作: <input style="border-bottom: 1px solid #ccc;" type="text" value="发送数据"/></p> <p>元件: <input style="border-bottom: 1px solid #ccc;" type="text" value="组合2"/> <input style="border-bottom: 1px solid #ccc;" type="text" value="组合1"/></p> </div>
切换页面	模块	否	<div style="background-color: #333; color: white; padding: 5px; border-radius: 5px;"> <p>名称: <input type="text" value="页面"/></p> <p>动作: <input style="border-bottom: 1px solid #ccc;" type="text" value="页面切换"/></p> <p>页面: <input style="border-bottom: 1px solid #ccc;" type="text" value="主页"/></p> </div>
轮换显示	元素	是	<div style="background-color: #333; color: white; padding: 5px; border-radius: 5px;"> <p>名称: <input type="text" value="轮换"/></p> <p>动作: <input style="border-bottom: 1px solid #ccc;" type="text" value="轮换显示"/></p> <p>元件: <input style="border-bottom: 1px solid #ccc;" type="text" value="组合2"/> <input style="border-bottom: 1px solid #ccc;" type="text" value="组合1"/></p> </div>

开关弹窗	模块	是	
打开外链	窗口	否	

参数设置

事件参数： 点击定时任务或延迟任务右侧的参数配置按钮

配置按钮	参数设置

动作参数： 给添加的动作设置附加额外参数

配置按钮	参数设置

辅助设计

排版对齐

参考线

页面中拖动元素，元素会根据当前舞台情况呈现可以对齐的参考线（并进行吸附动作）



层级调整

选中元素，在工具栏操作按钮，分别有：
上一级、下一级、置底、置顶



对齐操作

对舞台上的元素进行对选，在工具栏操作按钮，分别有：
左侧对齐、垂直居中、顶部对齐、水平居中、底部对齐、水平分布、垂直分布



网格吸附

可以通过开启舞台的网格，在元素拖拽结束时，自动吸附在最近的网格线上，并且网格尺寸大小可以调节。



复制黏贴

可以通过快捷键

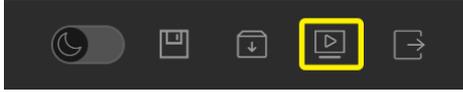
运行项目

项目预览

在编辑状态，项目预览时会自动保存项目数据

工具栏按钮

列表项按钮



下载项目

将项目下载到用户本地的设备上，下载的项目是编辑的数据信息，不包含配置引用的外部素材文件

工具栏按钮	列表项按钮

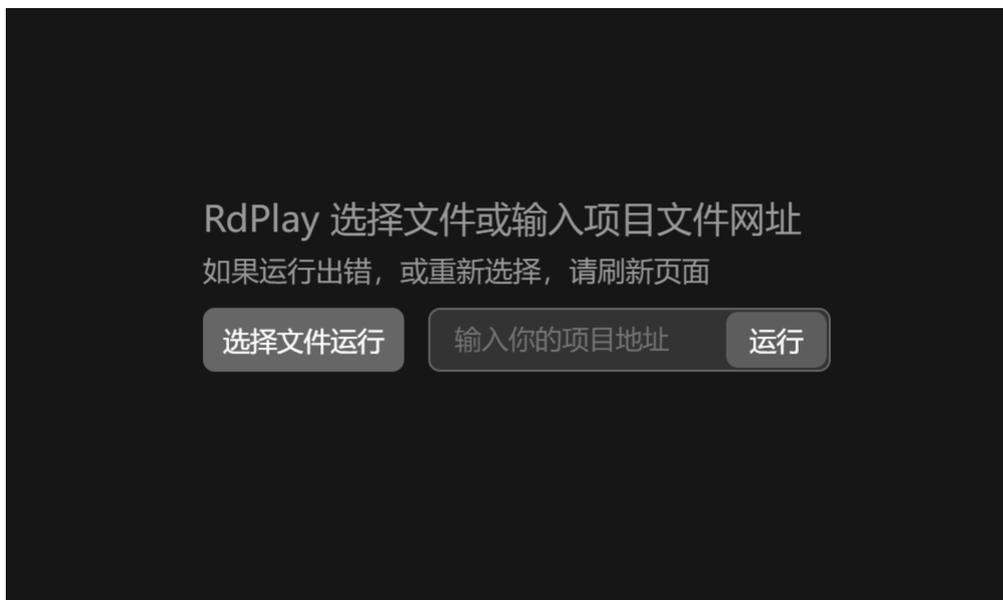
运行项目

项目的运行主要分为：rdPlay 播放器、自定义两种方式

rdPlay 播放器

播放器有在线版与本地安装版

两个版本的操作方式是相同的，都是通过选择运行文件或输入项目文件地址运行。



- 在线: <https://run.letdata.net>
- 本地: [下载播放器](#)

自定义

我们提供了，vue 插件与 javascript 代码嵌入两种选择

vue 插件

使用 npm 安装包

```
js
npm i rd-play
安装组件
js
import { createApp } from 'vue'
import rdPlay from 'rd-play'

const app = createApp(App)
app.use(rdPlay)
SFC 中使用组件
js
<template>
  <rd-play data="项目文件地址或项目文件数据" />
```



```
    </style>
</head>
<body>
  <div id="rdPlay" data-data="https://rde-1251496115.cos.ap-
shanghai.myqcloud.com/projects/A_nDTokkcSO-.json"></div>
</body>
</html>
```