|  |  |
| --- | --- |
| **URL** | **说明** |
| 基本界面 | |
| r'^ $' | 主界面 |
| r'^ test/$' | 测试界面 |
| 基本模块 | |
| r'^ admin/' | 后台管理模块（django自带） |
| r'^ account/' | 账号模块 |
| r'^ widget/’ | 组件模块 |
| r'^ scene/’ | 场景模块 |
| r'^ theme/’ | 主题模块 |
| 账号模块 | |
| r'^ account / $’ | 登录界面 |
| r'^ account /login/$’ | 登录界面 |
| r'^ account /register/$’ | 注册页面 |
| 组件模块 | |
| r'^ widget/ $’ | 组件列表/管理 |
| r'^ widget/list/$’ | 组件列表/管理 |
| r'^ widget/ create / $’ | 组件增加 GET/PUT |
| r'^ widget/delete/ $’ | 组件删除 |
| r'^ widget/edit/ $’ | 组件编辑 POST |
| 场景模块 | |
| r'^ scene/ $’ | 场景列表/管理 |
| r'^ scene/ list/ $’ | 场景列表/管理 |
| r'^ scene /create / $’ | 场景增加 |
| r'^ scene /delete/ $’ | 场景删除 |
| r'^ scene /edit/ $’ | 场景编辑 |
| 主题模块 | |
| r'^ theme/$’ | 主题列表/管理 |
| r'^ theme / list/ $’ | 场景列表/管理 |
| r'^ theme / create / $’ | 主题增加 |
| r'^ theme /delete/ $’ | 主题删除 |
| r'^ theme /edit/ $’ | 主题编辑 |
| r'^ theme /view/ $’ | 主题浏览 |

|  |  |
| --- | --- |
| **名词** | **英文** |
| 组件 | widget |
| 场景 | scene |
| 主题 | theme |
| 列表 | list |
| 增加 | create |
| 删除 | delete |
| 编辑 | edit |
| 浏览 | view |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

说明:

* 参数解释栏中的M表示必须项目，O表示可选项
* 服务器所有的返回数据都分为标记部分和数据部分/信息部分，即{succc: true, data: …/msg: ….}；其中标记部分不可少，而数据部分可以不存在。返回类型中的d表示是默认情况，默认情况是 {succ: true/false, msg: ‘……’}；对于其他的返回（比如是data结构），补上标记部分，实际返回结构就是 {succ: true/false, data: data}

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **URL** | **type** | **数据类型** | **参数名称** | **参数解释** | **返回类型** | **参数名称** | **功能解释** |
| 账号模块 ^account/ | | | | | | | |
| register | POST | String | username |  | D |  | 注册 |
| String | password |  |
| int | Identity | 浏览者/设计者 |
| array | themes | 有权限的主题 |
| GET |  |  |  | html |  | 注册页面 |
| Login | POST | String | Username |  | D |  | 登陆 |
| String | Password |  |  |
| GET |  |  |  | html |  | 登陆页面 |
| 组件模块 ^widget/ | | | | | | | |
| '$’ | GET |  |  |  | html |  | 组件列表页面 |
| String | xxx | 某某查询条件 |
| ‘{widget\_id}/$’ | GET |  |  |  | html |  | 组件详情页面，分设计和浏览 |
| 'delete/$’ | POST | Int | widget\_id |  | D |  | 删除 |
| 'draw/basic/$’ | POST | int | widget\_id | M | Json  数据内容在下面补充说明部分有 |  | 得到画图的基本数据，用这些即可画出图 |
| array | x | M [{kind: }, {cmd: }, {attr: }]  kind值为0、1，前者表示维度，后者表示度量  cmd值为’rgl’，表示不做运算，’avg’表示做平均值，‘sum’表示求总和  attr就是所选的属性名 |  |
| array | y | M 结构同x |  |
| string | graph | M |  |
| String | color | 对应的属性名称 |  |
| string | size | 对应的属性名称 |  |
| …… | …… |  |  |
| ‘draw/addtional/$’ | POST | int | widget\_id | M | d |  | 画图的辅助描述性数据，并不影响图本身 |
| string | Pic\_name | O |
| string | Xlab\_name | O |
| supply | bool | O允许场景使用 |
| …… | …….. |  |
| 场景模块 ^scene/ | | | | | | | |
| '$’ | GET |  |  |  | html |  | 场景列表页面 |
| '^{scene\_id}/ $’ | GET |  |  |  | html |  | 某场景中组件列表页面 |
| 'create / $’ | GET |  |  |  | html |  | 创建，成功后重定向到上面url，id为新生成的 |
| 'delete/$’ | POST | int | Scene\_id |  | D |  | 删除 |
| 'save /$’ | POST | int | scene\_id | M | D |  |  |
| array | widgetid\_list | O |  |  |  |
| string | layout | O(但上面出现，就必须出现) |  |  |  |
| bool | supply | O 允许主题使用 |  |  |  |
| ……. | …….. |  |  |  |  |
| 主题模块 ^theme/ | | | | | | | |
| '$’ | GET |  |  |  | html |  | 主题列表页面 |
| '{theme\_id}/$’ | GET |  |  |  | html |  | 某主题中场景列表 |
| ' create / $’ | GET |  |  |  | html |  | 创建，成功后重定向到上面url，id为新生成的 |
| 'delete/ $’ | POST | int | theme\_id |  | D |  | 删除 |
| ‘view/$’ | GET | Int | Scene\_Id |  | Html |  | 浏览（发布后） |
| 'save / $’ | POST | int | theme\_id | M | D |  | 保存 |
| array | Sceneid\_list | O Array本身就代表了顺序 |  |  |
| string | effect | O |  |  |
| bool | distribute | O |  |  |
|  | | | | | | | |
| 数据库连接模块 ^db/ | | | | | | | |
| conn/$ | POST | Int | Db\_kind |  | html |  | 连接数据库，成功生成新的组件id，重定向到 ^widget/id/$ |
| string | ip |  |
| String | port |  |
| string | Db\_name |  |
| String | password |  |
| string | username |  |
| GET |  |  |  | html |  |  |
| disconn/$ | POST |  |  |  | D |  | 断开连接 |
| view/{table}$ | GET |  |  |  | Json |  | 预览表数据 |

**补充说明：**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **URL** | **type** | **返回类型** | **参数名称** | **参数解释** | **功能解释** |
| 'draw/basic/$’ | POST | json |  |  |  |
|  | type | 图表类型 |  |
| If Type in (bar, line, scatter’) | data | X: [ {type: category/value  , data: [] } ]  Y: [ {type: category/value  , data: [] } ]  Legend\_series: [{‘series’:[], ‘legend’: []}] |  |
| If type = map | data |  |  |
|  |  |  |  |