



TaskBuilder 应用开发规范

v1.0

北京任讯信息技术有限公司

www.taskbuilder.org

2024 年 5 月

目 录

一、前言	1
二、项目或产品命名规范	1
三、数据模型设计规范	1
1、数据模型文件夹设计规范	1
2、数据模型文件名命名规则	2
2.1 总体原则	2
2.2 各子系统业务实体表的命名规则	2
2.3 业务实体扩展信息或关联信息表的命名规则	2
2.4 单据类一主多从关系的明细表的命名规则	3
2.5 实体数据分类表的命名规则	3
2.6 视图类数据模型命名规则	3
3、数据模型字段命名规则	3
4、数据模型字段数据类型设置规则	4
5、数据模型字段约束条件设置	5
6、数据模型字段默认值设置	6
7、数据模型字段备注设置	6
8、数据模型的索引设置	6
9、数据模型冗余字段设置	6
10、数据模型索引设置规范	6
四、后台服务设计规范	7
1、后台资源命名规则	7

1.1 后台文件夹命名规则	7
1.2 后台服务文件命名规则	7
2、后台服务基本信息设置规则	8
2.1 编辑模式选择	8
2.2 服务名称设置要求	8
2.3 服务说明设置要求	8
2.4 安全设置要求	8
2.5 请求参数设置要求	9
3、后台服务业务逻辑编排要求	9
3.1 数据校验要求	9
3.2 变量声明语句使用要求	10
3.3 变量赋值语句使用要求	11
3.4 判断语句使用要求	11
3.5 循环语句使用要求	12
3.6 返回语句使用要求	12
3.7 返回逻辑错误语句使用要求	12
3.8 终止服务语句使用要求	12
3.9 异步操作要求	13
4、常见后台语句使用规范	13
4.1 数据库操作相关要求	13
4.2 导入导出语句设置规范	15
4.3 附件操作语句设置规范	15

五、前端页面设计规范	16
1、前端文件夹设计规范	16
2、前端文件命名规范	16
3、前端页面设计规范	16
3.1 整体设计规范	16
3.2 页面类型的选择	17
3.3 列表页设计规范	18
3.4 添加修改页设计规范	22
3.5 详情页设计规范	26
4、前端页面常见布局参考	27
4.1 列表页常见布局	27
4.2 添加修改页面常见布局	33
4.3 详情页常见布局	35
六、代码编写规范	36
1、JavaScript 代码编写规范	36
1.1 变量和对象属性的声明	36
1.2 函数和对象方法命名	37
1.3 类的命名	37
1.4 注释	37
1.5 缩进	38
1.6 代码换行要求	38
1.7 行分隔符的使用	38

1.8 字符串值的设置和拼接	38
1.9 JSON 对象的定义	38
1.10 异步操作的使用	39
2、CSS 代码编写规范	39
2.1 样式规则命名	39

一、前言

为了规范我司开发人员在使用 TaskBuilder 进行软件开发时的各项行为，特制定此规范。

广大 TaskBuilder 开发者在开发自己的系统时，也可以借鉴和参考。

二、项目或产品命名规范

- (1) 一个业务子系统建一个项目，例如财务管理、人事管理、销售管理、采购管理、资产管理等。
- (2) 项目或产品代码只能是字母、数字或下划线，且必须字母开头，至少 2 个字符；
- (3) 项目代码不能使用 sys、system、dev、im、oa、om、demo、workflow 这几个单词；
- (4) 如果项目既有电脑端，也有手机端，建议单独为手机端建一个项目，数据模型和后台服务共用电脑端的，只需要在单独的项目中开发手机端的前端页面即可。

三、数据模型设计规范

1、数据模型文件夹设计规范

- (1) 如果子系统的数据模型比较少，可以不建文件夹，如果超过 20 个，建议按业务建文件夹，然后在文件夹内再建数据模型。
- (2) 文件夹名称只能是字母、数字或下划线，且必须以字母开头，建议用业务功能的英文单词，别用汉语拼音。

2、数据模型文件名命名规则

2.1 总体原则

(1) 数据模型文件名（包括表和视图）只能是小写字母和下划线，且必须以字母开头，禁止使用数据库关键字；

(2) 需要发布到应用市场的应用，数据模型文件名必须以在任讯应用开发平台注册的组织代码作为前缀，以便与其他开发者开发的应用系统的数据模型进行区分，避免因重名而产生冲突；

(3) 数据模型文件名中的子系统、业务实体、明细数据、扩展或关联信息等建议都用英文单词，不建议用汉语拼音；

(4) 子系统可以是英文单词全称，例如：financial（财务），也可以是缩写，例如：oa（办公管理）、fm（财务管理）、crm（客户关系管理）、hr（人力资源管理）、pm（项目管理）、scm（供应链管理）等，开发者根据业务系统的复杂度可以自行选择。除了子系统以外，其他部分不能用缩写。

2.2 各子系统业务实体表的命名规则

数据模型文件名格式：组织代码_子系统名称_业务实体名称

例如财务账户信息：taskmsg_fm_accounts

2.3 业务实体扩展信息或关联信息表的命名规则

数据模型文件名格式：组织代码_子系统名称_业务实体名称_扩展信息名称

例如客户联系人信息：taskmsg_crm_customer_contacts

2.4 单据类一主多从关系的明细表的命名规则

单据明细表数据模型文件名格式：单据主表名_details

例如采购订单明细表：taskmsg_purchase_order_details

2.5 实体数据分类表的命名规则

数据模型文件名格式：实体表名_type

例如物料信息分类：taskmsg_scm_material_type

2.6 视图类数据模型命名规则

视图数据模型文件名格式：组织代码_v 子系统名称_业务实体名称

例如客户信息视图：taskmsg_vcrm_customer

3、数据模型字段命名规则

- (1) 只能是小写字母、下划线或数字，且必须以字母开头，禁止使用数据库关键字；
- (2) 建议用英文单词全称，不建议用缩写和拼音；
- (3) 主键字段建议用 id ；
- (4) 外键字段命名规则：关联实体名_id，例如销售订单内的客户编号：customer_id
- (5) 常用字段命名约定：

编号：id

编码：code

名称：name

父类编号：parent_id

创建人编号: creator_id

创建时间: create_time

最后更新人编号: updater_id

最后更新时间: update_time

删除人编号: deleter_id

删除时间: delete_time

审核人编号: auditor_id

审核时间: audit_time

审核结果: audit_result

状态: status

备注: comment

经办人编号: operator_id

单据编号: bill_id

单据编码: bill_code

数量: quantity

单位: unit

单价: price

金额: amount

金额合计: total_amount

4、数据模型字段数据类型设置规则

- (1) 建议每个表建一个名为 id 的、整数类型主键，且设置为自动增长。如果需要通过跨库导

入导出或同步数据，则为了避免主键值与现有数据重复或顺序错乱，则可以采用任擎自带的编码规则生成主键值，或者采用 32 位 UUID 字符串。

(2) 只需要精确到年月日的日期数据用长度为 10 的“字符串”类型；需要精确到小时分钟秒的时间值用“日期时间”类型；需要精确到毫秒的时间值用“大整数”记录精确的时间戳；记录各种业务操作时间的数据，必须用“日期时间”类型，精确到秒。

(3) 存储金额数据用“金额”类型，默认精度（小数点保留位数）为两位，用来存人民币，如果要存其他币种，可以根据需要调整精度；

(4) 存储税率等对精确度要求高的小数时，类型选“数值”，然后可以设置具体的精度。

(5) 存字符串时，要尽量控制好长度，能确定最大长度的，就按最大长度来设置，例如身份证号、手机号等，不确定长度的、但不会特别长的，保证够用就行。

(6) 存储超过 5000 个字符的文本内容时，用“文本”类型；

(7) 手机号、身份证等数据不要用“整数”类型存储；

(8) 图片或附件代码用长度为 32 的“字符串”类型存储；

5、数据模型字段约束条件设置

(1) 非空约束：在 TaskBuilder 的数据模型设计器里将字段设置为非空后，在添加和修改时，如果这个字段的值为空，则任擎会自动返回“xx 字段不能为空！”的错误提示；

(2) 唯一性约束：在 TaskBuilder 的数据模型设计器里将字段设置为唯一后，目前任擎还没有实现自动校验，需要开发者在做添加和修改操作时，自行实现该字段值是否重复的校验，否则，数据重复时，会收到数据库软件抛出的错误提示，这种提示信息对于用户来说是不友好的，必须进行判断处理。

6、数据模型字段默认值设置

不建议在数据模型设计器里为字段设置默认值, 而是应该去添加或修改数据的后台服务里进行判断, 是否要为该字段设置默认值。

7、数据模型字段备注设置

请务必为每个字段设置备注(可以用汉字), 因为后续在创建数据增删改查页面时, 都需要用字段备注作为列表页面数据表格的列名或添加修改页面内输入项的文本标签, 如果不设置备注, 就会用字段名(英文字母), 对国内用户不友好。

8、数据模型的索引设置

对于需要作为查询条件供用户查询的字段, 建议为其创建索引, 但 TaskBuilder 的数据模型设计器目前不支持设置索引, 需要使用第三方的数据库管理工具设置。

9、数据模型冗余字段设置

为了提高数据查询速度, 对于数据量特别大、查询非常频繁的表, 建议可以适当创建一些冗余字段, 尽量减少多表关联查询, 例如创建人、审核人等这种信息, 除了存这些人员的编号外, 可以直接将其姓名存储在业务表里, 后续查询就不需要再关联系统用户表进行查询。

10、数据模型索引设置规范

- (1) 对于作为查询或排序条件频繁使用的字段, 建议创建索引;
- (2) 索引命名格式: 组织代码_i 表名_索引字段名, 例如客户信息表里的客户代码字段的索引:
taskmsg_icrm_customer_code

四、后台服务设计规范

1、后台资源命名规则

1.1 后台文件夹命名规则

(1) 一个功能模块一个文件夹，如果功能比较复杂，还可以创建子模块的文件，但不建议超过 3 级。同一个功能模块的前后端文件夹名称要一致。

(2) 文件夹名称只能是字母、数字或下划线，且必须以字母开头，建议用业务功能的英文单词，最好小写，不建议用汉语拼音。

1.2 后台服务文件命名规则

一个业务操作对应一个后台服务，服务名用业务操作对应的英文单词，并且要求小写，常见的后台服务文件名如下：

- **翻页查询：** `getDataPage.tbs`
- **不翻页查询：** `getList.tbs`
- **单条记录查询：** `get.tbs`
- **添加：** `add.tbs`
- **修改：** `update.tbs`
- **删除：** `delete.tbs`
- **禁用：** `disable.tbs`
- **启用：** `enable.tbs`
- **导入：** `import.tbs`

- **导出：** export.tbs
- **统计：** stat.tbs

2、后台服务基本信息设置规则

2.1 编辑模式选择

默认采用图形化格式（tbs），如果有复杂的业务逻辑，可以采用 js 格式。

2.2 服务名称设置要求

服务名称用来作为后台服务的类名，要求跟服务文件名一致，但首字母需要大写。

2.3 服务说明设置要求

服务说明需要说清楚当前服务的具体功能，以便其他开发者调用该服务时了解其用途。

2.4 安全设置要求

- **不校验是否登录：**新建的后台服务默认需要登录才能访问，如果某个服务可以对外公开，不需要登录就可以访问，则可以在后台服务的基本信息里，将“不校验是否登录”选中。
- **不检查 SQL 注入：**新建的后台服务默认会检查所有请求参数中是否有可能造成 SQL 注入的特殊字符或单词，如果发现有，会自动返回相应的错误信息。如果某个服务允许在请求参数中出现特殊字符，则可以在后台服务的基本信息里，将“不检查 SQL 注入”选中。一般不建议启用该选项，因为该选项选中后，提交给该后台服务的所有请求参数都不会进行 SQL 注入检查，建议在请求参数列表里，设置仅针对特定的参数不检查 SQL 注入，这样会更加安全。

2.5 请求参数设置要求

对于非字符串类型或者不能为空的请求参数，必须在后台服务的基本信息设置内，将其添加到请求参数列表里，并设置好参数的备注，然后在前端请求该服务时，引擎会根据这些设置自动对参数进行类型和必填校验，如果发现不满足要求，会返回相应的错误提示，并且提示中会采用参数备注来标识具体是哪个参数有问题。

3、后台服务业务逻辑编排要求

3.1 数据校验要求

对于客户端提交的请求数据，要进行必要的校验，包括：

- **数据类型校验：**校验客户端提交的请求参数是否是指定的数据类型，例如是否是日期、是否是整数等，如果不满足要求，则需要提示用户参数数据类型错误，请提供正确的数据。如果在后台服务的基本信息里已经将该参数添加到请求参数列表，并设置正确的数据类型，则在客户端请求该服务时，引擎会自动进行校验，否则，需要开发者在后台服务内添加相应的校验语句。
- **数据格式校验：**校验客户端提交的请求参数是否满足指定的数据格式，例如是否是正确的手机号、身份证号、邮政编码、邮箱地址等，另外，有些数据可能还存在一些特殊的格式要求，例如 TaskBuilder 的项目代码只能包含字母、数字和下划线，且必须以字母开头，对于这种情况，需要开发者在业务逻辑设计器内使用相应的语句进行校验，如果对正则表达式熟悉，可以用正则表达式进行校验。
- **数据长度校验：**校验客户端提交的请求参数是否满足特定的长度要求，长度包括最小长度和最大长度，也可以是固定长度，例如国内手机号必须是 11 位，不能长也不能短，用户

名最短不能少于 2 位，最长不能超过 20 位。

- **不能为空校验：**校验客户端提交的请求参数是否为空，如果不能为空，但客户端提供的参数值为空，则需要提示用户。如果在后台服务的基本信息里已经将该参数添加到请求参数列表，并设置了不能为空，则在客户端请求该服务时，引擎会自动进行校验，否则，需要开发者在后台服务内添加相应的校验语句。另外，如果请求参数是数组类型，且要求不能为空，则不能简单地判断该参数是否为空，还需要判断该参数的长度是否为 0，如果为 0，也视同为空。
- **数据重复性校验：**如果要添加或修改数据的表中，某个字段的值不能重复，则在添加或修改之前，需要查询该表中这个字段是否已经存在客户端传过来的值，如果已经存在，则需要提示用户，该字段的值已经存在。
- **数据合理性校验：**在增删改相关业务数据时，需要校验数据的合理性，例如付款操作，如果账户余额不足，则不能支付。例如出库操作，如果用户输入的出库数量大于库存数量，则不允许出库。
- **数据完整性校验：**在删除某条业务数据时，需要判断该数据是否已经被其他业务引用，如果已经被引用，则不允许直接物理删除，否则，在其他业务与该数据进行关联查询时，会造成查询结果中部分数据丢失。

3.2 变量声明语句使用要求

- (1) 变量名只能是字母、数字和下划线，且必须以字母开头；
- (2) 变量名建议使用英语单词，不建议用汉语拼音；
- (3) 如果变量包含多个英语单词，则需要采用驼峰式命名法，即第一个单词所有字母小写，从第二个单词开始，首字母大写，其他字母小写，例如：isExistsCustomer；

(4) 变量名不能使用系统关键字，包括 JavaScript 语言的关键字、node.js 的关键字和任擎后台服务的关键字，JavaScript 语言的关键字和 node.js 的关键字请自行用搜索引擎搜索，任擎后台服务的关键字有：self、server、req、res 这几个。

(5) 如果能确定变量的类型，则在声明变量时最好设置一下类型，以便在后续操作中能根据变量类型自动做一些相关处理，例如赋值时，如果是字符串或日期类型，可以自动添加单引号；

(6) 如果变量有默认值，也建议设置好默认值，默认值的设置要求参考下面的变量赋值要求；

3.3 变量赋值语句使用要求

(1) 如果是字符串或日期类型的值（不是其他变量或表达式），则在赋值给变量时，需要在值两边加上英文的单引号；

(2) 如果要给对象的属性赋值，则在变量名输入框内输入的内容格式为：对象名.属性名；

(3) 只有数值类型支持自增（++）和自减（--）赋值；

3.4 判断语句使用要求

(1) 等于的运算符为：==，不等于的运算符为：!=；

(2) 判断表达式里可以有多个与或非运算，与运算的符号为：&&，或运算的符号为：||，非运算的符号为：!，多个与或非运算可以用括号进行分组，例如：(var1>=10 && var1<=15) || var2 in (1,2)；

(3) 在判断语句的每个分支内，最后都需要终止服务，否则会造成服务无法正常结束，一直占用服务器资源。可以使用“终止服务”+“返回”语句终止，也可以使用“返回逻辑错误”语句终止。

3.5 循环语句使用要求

不能在循环语句中执行异步操作，因为异步操作不是阻断式、一个一个依次执行的，循环时，不一定哪一次循环的异步操作先执行完，很可能造成不可预测的错误。如果循环内要执行数据库操作、文件读写、HTTP 请求等操作，请使用其对应的同步操作语句。

3.6 返回语句使用要求

在后台的方法内，可以使用返回语句退出当前方法的执行，一般用在判断语句内，例如经过判断，不满足相应条件，则可以用返回语句退出当前方法的执行，如果该方法是入口方法 process 或没有返回值的方法，则在返回语句前面，需要拖拽一个终止服务的方法，否则可能会造成当前服务无法正常结束，一直占用服务器资源。

3.7 返回逻辑错误语句使用要求

- (1) 在后台服务内进行各种校验时，如果有不满足条件的情况，则需要用“返回逻辑错误”语句向客户端返回明确的错误信息；
- (2) 如果同一个后台服务内，会返回多个逻辑错误，则每个逻辑错误要设置不同的错误代码，以便客户端进行判断，错误代码只能是整数，且不能为 0；
- (3) 提示、警告、或错误说明应精简、明确且易于理解；
- (4) 提示中不要包含后台表字段名或数据库表名等敏感信息；

3.8 终止服务语句使用要求

- (1) 如果有多个异步操作层层嵌套，必须在最里面的异步操作的回调函数里结束服务，否则该服务会一直在后台运行，已经打开的网络和数据库连接也不会关闭，多次请求该服务后，会

造成数据库连接超过最大连接数，导致其他服务无法连接数据库进行相关操作。

(2) 如果后台服务内有多个程序分支，例如有判断语句，那么每个分支的最里面的回调函数里，都必须有结束服务语句。

3.9 异步操作要求

(1) 如果服务内有多个异步操作，例如数据库增删改查、文件读写、服务调用、HTTP 请求等，必须层层嵌套，即后一个异步操作要放在前一个异步操作的回调函数里面，作为其子语句运行，不能将多个异步操作上下（不是里外）平级摆放，否则会造成服务执行失败。

(2) 如果需要循环执行数据库操作，请使用同步方式的数据库操作，不要在循环中进行异步方式的数据库操作，因为异步操作不是按顺序一个接一个依次执行的，是多个操作同时执行，不一定哪一个先执行完，也不是等所有操作都执行完才结束循环，所以，无法决定什么时候结束服务。

4、常见后台语句使用规范

4.1 数据库操作相关要求

4.1.1 在 SQL 语句中插入 js 变量的要求

(1) 在后台服务的数据库操作的 SQL 语句中，如果要插入请求参数或后台服务中的变量值，建议使用 {变量名} 的形式，例如修改数据记录的条件：id={req.id}，采用这种方式，变量的值会当做参数值传给数据库，由数据库在执行相应 SQL 语句时，将 {变量名} 转换为对应的值，即使变量值里有特殊符号或 SQL 关键字，数据库也只会当做一个字符串值进行处理，不会被拼接成 SQL 语句，可以有效避免被 SQL 注入；

(2) 如果有特殊情况，需要将请求参数或后台变量作为字符串拼接到 SQL 语句中，则可以采用：\${变量名} 的形式，例如批量删除时，前端传给后台服务的 id 值是以英文逗号分隔的多个 id 值，这时就可以用这种形式：id in (\${id}) 。

4.1.2 数据查询语句设置规范

- (1) 前端需要哪些字段就查哪些字段，用不上字段的不要查；
- (2) 对于数据较多的查询必须启用翻页，不要一次性查询大量数据返回给前端；
- (3) 如果需要多个表关联查询，建议创建视图通过 join 表连接进行关联查询，尽量不要使用子查询；
- (4) 如果只需要返回单条数据，则将查询结果保存到响应对象里时，只能取结果数组中的第一条记录，而且，需要先判断查询结果数组的长度是否>0，否则返回未找到数据的错误。

4.1.3 数据添加和修改语句设置规范

- (1) 记录操作时间的字段值要在后台设置为服务器系统的当前时间，不要用前端用户电脑的系统时间；
- (2) 记录操作人信息的字段值，例如操作人编号、操作人姓名等，也要在后台服务的数据添加设置里，该字段的值设置为当前用户编号或姓名；
- (3) 要详细记录执行数据添加或修改操作的人和具体时间，以备后续查证；

4.1.4 数据删除语句设置规范

- (1) 删除时，需要校验是否有关联数据，如果有，则不能删除。（比如：删除某个分类，如果分类下有数据，则不能删除）；

- (2) 建议用一个字段标识某条数据记录是否已被删除，删除时修改该字段的值即可，尽量不要物理删除；
- (3) 要详细记录执行删除操作的人和具体时间，以备后续查证；；

4.2 导入导出语句设置规范

4.2.1 数据导入语句设置规范

- (1) 如果要对导入的每条数据进行复杂的校验和判断，然后再决定是否导入或动态计算生成某一系列值，这种情况，建议先批量导入到某个临时表，然后再进行相关处理；
- (2) 单次导入的数据行数不要超过 3 万条，超过 3 万条记录的，请分多次导入；
- (3) 导入的 Excel 文件格式必须为.xlsx；

4.2.2 数据导出语句设置规范

- (1) 不要一次性导出超过 10 万条记录；
- (2) 仅导出需要的字段，不要所有字段都导出；

4.3 附件操作语句设置规范

- (1) 附件表内目前记录应用数据 id 值字段只支持整数，所以在添加修改附件时，请提供整数类型的“应用数据记录编号”；
- (2) 修改应用的附件信息时，如果原来已经有附件信息，且这些信息还需要保留，则在修改时，原来的附件信息也要提供，否则会视为要被删除；

五、前端页面设计规范

1、前端文件夹设计规范

(1) 一个功能模块一个文件夹，如果功能比较复杂，还可以创建子模块的文件，但不建议超过 3 级。同一个功能模块的前后端文件夹名称要一致。

(2) 文件夹名称只能是字母、数字或下划线，且必须以字母开头，建议用业务功能的英文单词，最好小写，不建议用汉语拼音。

2、前端文件命名规范

文件名只能是小写字母、数字或下划线，且必须以字母开头，建议用可以体现页面具体功能的英语单词，例如 add、update 等，不建议用汉语拼音。

常见页面类型命名规则：

- **模块首页**：index.tfp
- **添加页面**：add.tfp
- **修改页面**：update.tfp
- **详情页面**：detail.tfp
- **弹窗选择列表页**：list.tfp

3、前端页面设计规范

3.1 整体设计规范

(1) 页边距固定为 20 像素；

(2) 默认字体为微软雅黑，字号为 14px；

- (3) 分组标题的文字字号为 16px，加粗；
- (4) 输入项文本标签的颜色默认为：#666666；
- (5) 输入项内容的文字颜色默认为：#333333；
- (6) 普通页面背景颜色为浅灰色，最底层放一个背景颜色为白色的面板组件，然后再在上面放其他组件；
- (7) 对话框和抽屉页的背景为白色，不需要加背景面板；
- (8) 普通页面和对话框页面的确定取消按钮放在页面底部，抽屉页不需要。
- (9) 布局要合理，内容不宜过于密集，也不能过于空旷，合理的利用空间；
- (10) 对话框应尽量在不借助水平或垂直滚动条的情况下显示所有内容；

3.2 页面类型的选择

TaskBuilder 支持以下三种类型的页面，各自有其适用的场景，请根据需要选择：

	普通页面	抽屉页	对话框
显示方式	显示在后台管理系统主界面区域	显示在后台管理系统主界面右侧	显示在后台管理系统中间
打开方式	1、点击系统菜单打开 2、通过程序代码打开	通过程序代码打开	通过程序代码打开
适用场景	1、应用模块首页； 2、内容很多的添加修改页； 3、内容复杂的详情页； 4、统计分析报表页面；	1、添加修改页面默认方式； 2、查看详情页默认方式；	1、内容很少的添加修改页； 2、弹窗数据选择页面；
打开方法	tfp.openPage()	tfp.openDrawer()	tfp.openDialog()

3.3 列表页设计规范

3.3.1 查询条件设置

- (1) 列表页面提供的查询条件要尽量完备充足，并按使用频次依次排列；
- (2) 能列表选择的尽量用下拉列表或数据字典组件供用户选择；只有两个值可以选的，用单选框或开关组件；
- (3) 能模糊查询的尽量支持模糊查询；
- (4) 查询日期时间时，需要提供起止时间范围查询；
- (5) 如果有多个查询条件，查询条件文本标签、输入项和按钮之间要保留 10 像素的间距；
- (6) 查询条件超过 5 个时，需要放在页面左侧，不要在数据列表上面放置过多查询条件，导致出现横向滚动条；
- (7) 如果页面左侧放置查询条件的区域还有很多空白，可以将一些数据字典类的查询条件用静态列表组件列表显示；
- (8) 如果有查询条件是多级树形结构，则需要放在列表页面左侧用树形组件显示，默认展开第一层节点，然后用户可以层层展开，点击某个节点查询相应数据。

3.3.2 显示字段的选择

- (1) 如果数据模型的字段比较多，尽量选最关键、用户最关心的字段，不要把所有字段都显示在列表里；
- (2) 不要在列表里显示 id、operator_id、supplier_id 这种没有业务含义的字段；
- (3) 不要在列表里显示内容非常多的文本内容，尤其是富文本，例如备注、简介等；
- (4) 如果有些敏感数据不是所有人都可以看，例如合同金额、工资薪酬等，则需要为不同身

份的人提供不同的列表，有权限的人打开的列表里可以显示这些敏感信息，没有权限的人打开的列表里不要显示这些信息。

3.3.3 列宽设置

(1) 字段值长度比较短或者长度固定的，列宽比列值宽 20 像素即可，两边不要留太多空白，但至少要保证列标题文字不串行；

(2) 长度不固定的，而且可能比较长的，宽度可以设置宽一些，但不建议超过 300 像素，可以设置这一列最多显示的字符数，然后设置鼠标放到单元格上自动显示全部内容；

(3) 常用字段推荐宽度设置：

字段名	宽度 (像素)	备注
姓名	70	如果有少数民族用户，姓名宽度可以加宽
性别	50	
电话	120	
身份证号	150	
日期	90	
日期时间	150	
金额	100	如果金额可能过亿，则宽度可以再加宽

3.3.4 列内容对齐设置

(1) 数据表格内的数据默认居中对齐；

(2) 内容长度不固定、而且可能比较长的左对齐

- (3) 金额右对齐。
- (4) 图片或文件大小值右对齐。

3.3.5 操作按钮或链接设置

- (1) 按钮默认采用默认风格，不要使用圆角矩形和椭圆形；
- (2) 重要的按钮与使用较频繁的按钮要放在界面上醒目的位置；
- (3) 修改和删除按钮默认放最后面两列，建议加上小图标会更加美观；
- (4) 查看详情的链接建议加在 1-3 列的某一关键列上，颜色要设置为蓝色，表示可以点击；
- (5) 点击删除按钮时，需要显示确认对话框，经过用户确认后才能删除，避免误删；
- (6) 数据表格上部工具栏上如果有多个按钮，最多 2-3 个按钮设置背景颜色，其他按钮不要设置，常见按钮的颜色和图标参考下表：

按钮类型	背景颜色和图标	宽度	备注
添加		默认即可	
查询		默认即可	放在数据列表上部时
		100%	放在页面左侧边栏时
修改		默认即可	
删除		默认即可	
导入		默认即可	
导出		默认即可	

3.3.6 内容格式设置

- (1) 日期格式默认使用：yyyy-MM-dd；
- (2) 日期时间格式默认使用：yyyy-MM-dd HH:mm:ss；
- (3) 如果不需要查看具体时间，则日期字段设置为只需要显示年月日即可；
- (4) 金额字段要设置为显示千位分隔符的金额格式。

3.3.7 列内容类型设置

- (1) 对于可选值少于 5 个的数据字典列，建议设置显示条件，不同的值用不同颜色的椭圆按钮显示，提升美观度；
- (2) 点击后可以打开其他页面或对话框的列，用按钮或链接类型；
- (3) 只有两个可选值的列可以采用开关类型；
- (4) 可以计算出百分比的列可以采用进度条显示；

3.3.8 数据加载设置

- (1) 通过选项卡可以切换显示多个数据列表的页面，默认仅加载第一个选项卡内的数据，在切换到其他选项卡后，再加载这个选项卡里的数据，这样可以避免因为同时加载多个列表的数据而导致页面打开缓慢。
- (2) 数据加载时默认显示加载动画，提示用户正在加载数据；
- (3) 数据加载失败默认弹出对话框显示错误提示；
- (4) 页面里数据表格组件的翻页设置和后台服务里数据查询的翻页设置要一致。

3.3.9 其他设置

- (1) 查询结果可能超过 100 行的，必须启用翻页功能；
- (2) 第 1-3 列的某一列建议加一个小图标提升美观度；
- (3) 金额、数量类的列可以设置计算和显示合计；
- (4) 如果列比较多，可以将左侧关键列设置为固定，不随横向滚动条滚动；
- (5) 存在先后顺序的列可以启用点击标题栏进行排序，例如时间、单据编码、金额、数量等；

3.4 添加修改页设计规范

3.4.1 页面类型选择

- (1) 对于输入项少于或等于 16 个的，可以用对话框实现；
- (2) 对于输入项超过 16 个的，建议用抽屉页；

3.4.2 输入项文本标签位置

- (1) 建议放在输入项上面。

3.4.3 输入项文本标签颜色

- (1) 建议用浅灰色（#666666），以便跟输入项内容的深灰色进行区分。

3.4.4 每行显示输入项数量

- (1) 对话框建议每行显示两个输入项；
- (2) 抽屉页建议每行显示 3-4 个输入项；
- (3) 内容较多的输入项要占 2-3 个输入项的位置，例如地址、备注、简介等；

3.4.5 输入项类型选择

(1) 输入项类型应根据业务需要和数据库中字段的数据类型灵活选择，常见字段类型与输入项类型的对应表：

字段数据类型	输入项类型	备注
短字符串 (VARCHAR)	单行文本输入框	字符串长度小于等于 200
长字符串 (VARCHAR)	多行文本输入框	字符串长度大于 200
文本 (TEXT)	富文本输入框	
整数 (INT、TINYINT、SMALLINT、BIGINT)	单行文本输入框 计数器组件 滑块组件	数据类型设置为：整数
小数 (FLOAT、DOUBLE、REAL、NUMERIC)	单行文本输入框	数据类型设置为：小数
金额 (DECIMAL)	单行文本输入框	数据类型设置为：金额
日期时间 (DATETIME、SMALLDATETIME)	日期时间组件	
日期字符串 (VARCHAR(10))	日期组件	日期格式为：yyyy-MM-dd
时间字符串 (VARCHAR(5-8))	时间组件	精确到秒时 8 位，否则 5 位
仅有两个可选值的字符 (CHAR) 或微整数 (TINYINT)	开关组件	
单选整数值 (INT)	单选组件 下拉列表组件 弹出列表组件	选中值为整数

单选字符串值 (VARCHAR)	单选组件 下拉列表组件 弹出列表组件	选中值为字符串
多选字符串 (VARCHAR)	复选框组件	
数据字典编码 (VARCHAR(20))	数据字典组件	
编码规则字符串 (VARCHAR(20))	编码规则组件	
用户编号 (INT)	用户选择组件	值类型：用户编号
用户编码 (VARCHAR(20))	用户选择组件	值类型：用户编码
用户姓名 (VARCHAR(20))	用户选择组件	值类型：用户姓名
用户对象 JSON 字符串 (VARCHAR(100))	用户选择组件	值类型：用户对象
部门编号 (INT)	部门选择组件	值类型：部门编号
部门编码 (VARCHAR(20))	部门选择组件	值类型：部门编码
部门姓名 (VARCHAR(20))	部门选择组件	值类型：部门姓名
部门对象 JSON 字符串 (VARCHAR(100))	部门选择组件	值类型：部门对象
角色编号 (INT)	角色选择组件	值类型：角色编号
角色编码 (VARCHAR(20))	角色选择组件	值类型：角色编码
角色姓名 (VARCHAR(20))	角色选择组件	值类型：角色姓名
角色对象 JSON 字符串 (VARCHAR(100))	角色选择组件	值类型：角色对象
附件代码字符串 (VARCHAR(32))	图片上传组件 文件上传组件	仅保存附件的代码 (code)
附件 JSON 字符串 (VARCHAR(500))	图片上传组件 文件上传组件	保存附件 JSON 对象的完整信息

二维码字符串 (VARCHAR)	二维码组件	
条形码字符串 (VARCHAR)	条形码组件	

(2) 单选框、复选框、下拉列表框的选项应按使用频率由高到低排序，界面空间较小时，单选推荐使用下拉列表组件，复选使用弹出列表组件；

(3) 只有两个选项的单选使用开关组件；

(4) 图片上传和文件上传组件需要设置应用代码；

3.4.6 输入项宽度设置

(1) 页面默认采用流式定位，用弹性栅格进行组件布局，则输入项宽度默认为 100%，即填充整个父组件；

(2) 如果采用绝对定位，则可以根据实际需要设置其宽度；

(3) 单行输入框和多行输入框要设置最大可输入的字符数量，不能无限制，具体长度应与数据库中字段的长度一致；

3.4.7 内容分组显示

(1) 如果输入项比较多，建议分成几组，用分组标题隔开，标题前面要加蓝色竖条，字体加粗，字号为 16 像素。

3.4.8 必填项提示

(1) 对于必填项，其文本标签后面要显示红色的星号，以便与非必填项进行区分。

3.4.9 输入项备注设置

(1) 表单内的输入项需要设置好备注信息，备注建议用汉字，以便在提交表单时，如果输入项数据不满足相关要求，可以根据备注提示用户，而不是组件编号。

3.5 详情页设计规范

3.5.1 页面类型选择

- (1) 对于输入项少于或等于 16 个的，可以用对话框实现；
- (2) 对于输入项超过 16 个的，建议用抽屉页；
- (3) 内容很多，且存在跟其他多个业务有关联的，建议用普通页面，里面用选项卡分页显示；

3.5.2 输入项文本标签位置

- (1) 建议放在输入项上面。

3.5.3 输入项文本标签颜色

- (1) 建议用浅灰色（#666666），以便跟输入项内容的深灰色进行区分。

3.5.4 每行显示输入项数量

- (1) 对话框建议每行显示两个输入项；
- (2) 抽屉页建议每行显示 3-4 个输入项；
- (3) 内容较多的输入项要占 2-3 个输入项的位置，例如地址、备注、简介等；

3.4.5 输入项宽度

(1) 页面默认采用流式定位，用弹性栅格进行组件布局，则输入项宽度默认为 100%，即填充整个父组件；

(2) 如果采用绝对定位，则可以根据实际需要设置其宽度；

3.4.6 内容分组显示

(1) 如果输入项比较多，建议分成几组，用分组标题隔开，标题前面要加蓝色竖条，字体加粗，字号为 16 像素。

4、前端页面常见布局参考

4.1 列表页常见布局

4.1.1 标准布局

分上下两个部分，上部显示添加按钮、导入导出按钮、查询条件和查询按钮等，下面显示数据表格。

适用于以下两种情况：

(1) 查询条件 and 操作按钮比较少，不超过 5 个，参考界面如下图所示：

编码	名称	类型	开户银行	银行账号	余额	修改	删除
01	现金	01			186,410.00	修改	删除
04	支付宝	03			60,700.00	修改	删除

参考页面地址：/web/app/demo/page/list/index1.tfp

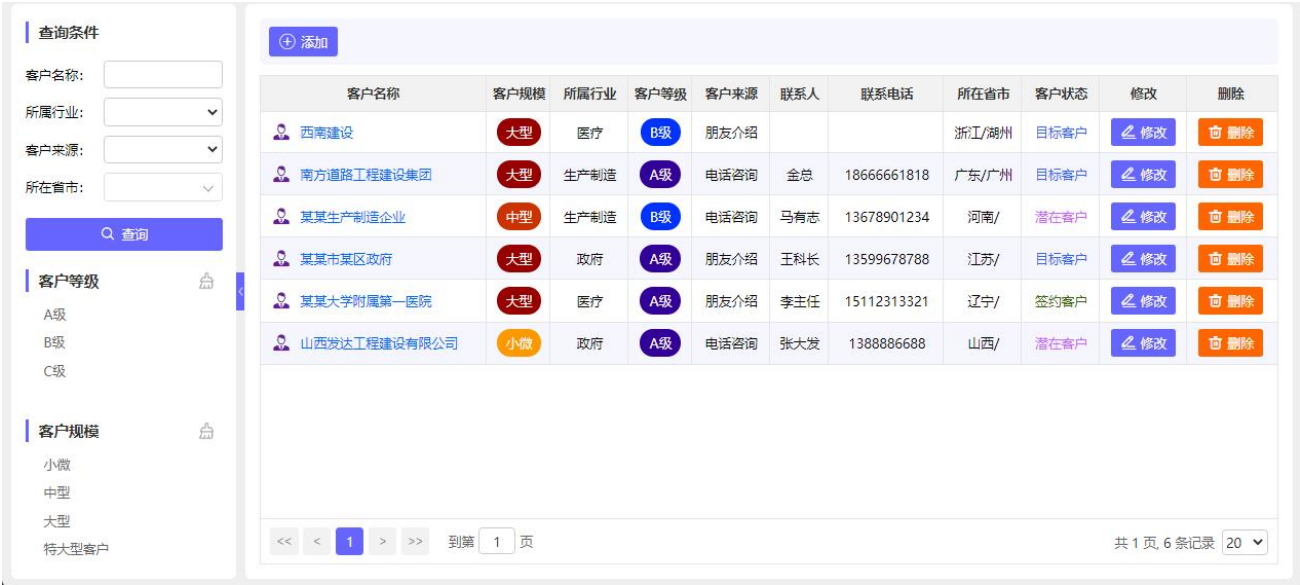
(2) 在其他页面里用对话框形式打开，用来选择列表内的数据；

4.1.2 左右两栏布局

内容分为左右两栏显示，左侧显示查询条件，右侧显示数据表格。

适用于以下两种情况：

(1) 查询条件比较多，放在表格上部可能会出现横向滚动条，参考界面如下图所示：



参考页面地址: /web/app/demo/page/list/index3.tfp

(2) 需要按树形结构层层展开多级分类查询, 参考界面如下图所示:



参考页面地址: /web/app/demo/page/list/index2.tfp

4.1.3 选项卡布局

如果列表数据可以按某个字段的值分别查询, 且该字段等于某个值的数据对于用户来说是更加关注、需要经常查看, 而其他值的数据不需要经常看, 这种情况就可以使用选项卡形式,

将这个字段的可选值作为选项卡的标题，默认显示用户最关心的数据，然后点击其他选项卡可以查看满足其他值的数据。例如工作列表，默认显示待办工作，然后要查看已转交、办理中、已办结的工作可以点选项卡上的相应标题进行查看。

参考界面如下图所示：



参考页面地址：/web/app/demo/page/list/index4.tfp

4.1.4 主明细联动列表页面

需要通过弹窗选择的数据还存在明细信息时，则需要提供主明细联动显示页面，常见就是各种单据的选择，例如销售付款时，列表选择多张销售订单时，不仅需要查看订单主表的信息，还需要查看订单的明细信息，否则很容易操作出错。参考界面如下图所示，点击上面的主表信息，下面就会联动显示其明细数据：

订单编码:
客户名称:
经办人:

☐	订单编码	客户名称	订单金额	折扣金额	实际金额	经办人	下单时间	备注
☐	XS20230021	某某大学附属第一医院	400.00	0.00	400.00	管理员	2023-05-29 08:00:00	
☐	XS20230017	西南建设	900.00	0.00	900.00	管理员	2023-05-12 08:00:00	
☐	XS20230016	某某生产制造企业	4,500.00	0.00	4,500.00	管理员	2023-04-28 08:00:00	
☐	XS20230018	某某生产制造企业	11,100.00	1,100.00	10,000.00	管理员	2023-03-26 08:00:00	
☐	XS20230016	某某生产制造企业	25,200.00	0.00	25,200.00	管理员	2023-03-25 08:00:00	士大夫
☐	XS20230014	某某生产制造企业	196,000.00	6,000.00	190,000.00	管理员	2023-03-25 08:00:00	士大夫
☐	XS20230013	某某生产制造企业	33,200.00	200.00	33,000.00	管理员	2023-03-25 08:00:00	
☐	XS20230012	某某市某区政府	19,600.00	600.00	19,000.00	管理员	2023-03-25 08:00:00	
☐	XS20230010	某某生产制造企业	750.00	50.00	700.00	管理员	2023-03-25 08:00:00	

<< < 1 > >> 到第 1 页
共 1 页, 14 条记录 20

服务项目	数量	单位	单价	金额	备注
背胶+冷裱板	10	平方米	30.00	300.00	
UV灯片	10	平方米	10.00	100.00	
合计					400.00

参考页面地址: /web/app/demo/page/list/index6.tfp

4.1.5 自定义单元格内容的列表页

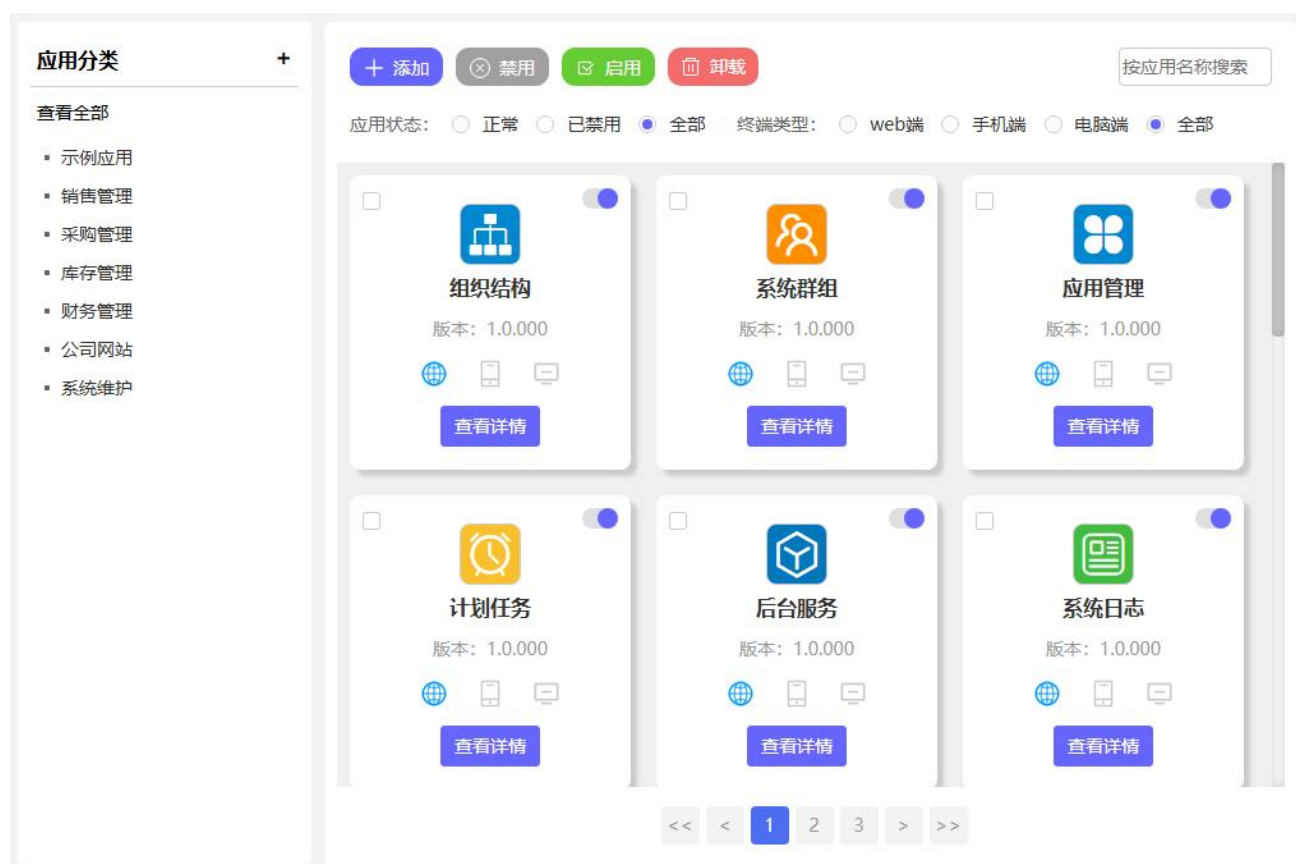
对界面美观度有较高要求, 不想用千篇一律的、一条记录用一行显示的形式, 则可以使用自定义单元格, 可以随意拖拽任意组件到自定义单元格内, 设计出复杂美观的列表界面。参考界面如下图所示:



参考页面地址: /web/app/demo/page/list/index5.tfp

4.1.6 数据卡片列表页

对于数据量不大（如果有多级层次关系，某一级的数据不是很多），有图标的数据记录，如果想让列表界面比较美观，可以采用数据卡片进行显示，参考界面如下图所示：



参考页面地址: </web/app/sys/app/index.tfp>

4.2 添加修改页面常见布局

4.2.1 对话框形式

对于内容比较少的添加和修改页面，建议使用对话框形式显示，如下图所示：

参考页面地址: /web/app/demo/page/add/add1.tfp

4.2.2 抽屉页形式

对于内容较多的添加或修改页面, 建议使用抽屉页, 参考界面如下图所示:

参考页面地址: /web/app/demo/page/add/add1.tfp

4.3 详情页常见布局

4.3.1 抽屉页形式

详情页默认建议使用抽屉页，参考界面如下图所示：

The screenshot displays a web application interface for '查看客户详情' (View Customer Details). On the left, a dark sidebar contains a list of '示例应用' (Example Applications) from 1 to 6. A drawer is open, showing '查询条件' (Search Conditions) with fields for '客户名称' (Customer Name), '所属行业' (Industry), '客户来源' (Source), and '所在省市' (Region), along with a '查询' (Search) button. Below the search conditions are sections for '客户等级' (Customer Level) with options A, B, and C, and '客户规模' (Customer Scale) with options 小微, 中型, 大型, and 特大型客户. The main content area is titled '基本信息' (Basic Information) and contains fields for '客户编码' (1000), '客户名称' (山西发达工程建设有限公司), '所属销售' (周瑾), '客户规模' (小微), '所属行业' (政府), '客户等级' (A级), '客户来源' (电话咨询), '联系人' (张大发), '联系电话' (1388886688), '所在省市' (山西/), '公司地址' (empty), '详细地址' (asdf), '财税信息' (Tax Information) with fields for '税号' (asdf), '开户银行' (asdf), and '银行账号' (asdf), and a '备注' (Remarks) field containing 'asdf'. At the top right, there are '确定' (Confirm) and '取消' (Cancel) buttons.

参考页面地址：/web/app/demo/page/detail/detail1.tfp

4.3.2 普通页面形式

对于内容很多，且存在多个关联信息的详情页面，需要使用普通页面来显示，且里面用选项卡分别显示基本信息和其他关联信息，参考界面如下图：



参考页面地址：/web/app/demo/page/detail/detail2.tfp

六、代码编写规范

1、JavaScript 代码编写规范

1.1 变量和对象属性的声明

- (1) 常量声明使用 `const` 关键字；
- (2) 变量声明使用 `let` 关键字；
- (3) 变量和对象属性名只能是字母或数字，且必须以字母开头；
- (4) 变量和对象属性名建议使用英语单词，不建议用汉语拼音；
- (5) 如果变量或对象属性名包含多个英语单词，则需要采用驼峰式命名法，即第一个单词所有字母小写，从第二个单词开始，后面的所有单词，首字母大写，其他字母小写，例如：
`isExistsCustomer`；

(6) 变量和对象属性名不能使用 JavaScript 语言的关键字，前端代码内的变量名还不能使用浏览器的关键字和 tfp 框架的关键字，后台服务内的变量名不能使用 node.js 的关键字和任擎后台服务的关键字。

1.2 函数和对象方法命名

(1) 变量和对象属性名只能是字母或数字，且必须以字母开头；

(2) 变量和对象属性名建议使用英语单词，不建议用汉语拼音；

(3) 如果变量或对象属性名包含多个英语单词，则需要采用驼峰式命名法，即第一个单词所有字母小写，从第二个单词开始，后面的所有单词，首字母大写，其他字母小写，例如：
isExistsCustomer；

(4) 变量和对象属性名不能使用 JavaScript 语言的关键字，前端代码内的变量名还不能使用浏览器的关键字和 tfp 框架的关键字，后台服务内的变量名不能使用 node.js 的关键字和任擎后台服务的关键字。

1.3 类的命名

(1) 类名只能是字母或数字，且必须以大写字母开头；

(2) 类名建议使用英语单词，不建议用汉语拼音；

(3) 类名如果包含多个英语单词，所有单词的首字母大写，其他字母小写；

(4) 类名不能使用 JavaScript 语言的关键字，前端代码内的类名还不能使用浏览器的关键字和 tfp 框架的关键字，后台服务的类名不能使用 node.js 的关键字和任擎后台服务的关键字。

1.4 注释

(1) 代码注释包括类注释（必须写）、方法注释（必须写）、块注释（关键位置、特殊含义）；

- (2) 类注释要简要介绍该类的主要功能;
- (2) 方法注释包含方法功能介绍、参数和返回值的含义及数据类型, 参数是对象的, 最好把关键的属性也写一下注释;

1.5 缩进

- (1) js 代码采用 Tab 键进行缩进
- (2) 在 TaskBuilder 或 VS Code 内, 可以使用 Shift+Alt+F 快捷键实现代码的格式化

1.6 代码换行要求

- (1) 一行一条语句, 不要一行放多条语句;
- (2) 判断、循环、函数语句的开始大括号{ 后面要换行;
- (3) 一行代码最多不要超过 100 个字符, 太长了后要换行;

1.7 行分隔符的使用

- (1) 每一行的结尾需要加分号;
- (2) 判断、循环、函数或方法最后一行的大括号后面不需要加分号;

1.8 字符串值的设置和拼接

- (1) 优先使用单引号定义字符串;
- (2) 优先使用模版字符串代替字符串拼接;

1.9 JSON 对象的定义

- (1) 要么所有属性不加引号, 要么都加, 如果有一个属性名需要加引号 (例如带减号的样式

名)，则所有属性都需要加引号；

1.10 异步操作的使用

- (1) 异步操作要层层嵌套执行，后一个异步操作要放在前一个异步操作的回调函数里；
- (2) 不要在循环中执行异步操作，例如循环请求后台服务；

2、CSS 代码编写规范

2.1 样式规则命名

- (1) CSS 样式规则名称只能是小写字母、数字或下划线，且必须字母开头（不包括前缀里的#和. 符号）；