

CrudlessHub快速使用

1. 使用目的

通过配置编排`crudless.yml`文件内容并搭配NodeJS工具 (`crudless` 指令) 生成详细的**前后端与数据库**的配置文件, 最后再配合 `zero-json` (**前端代码生成工具**) 与 `cg-api-cli` (**后端代码生成工具**) 配合生成**可即时运行**的总代码。

2. 相关链接

⋮ Tips: 地址无法访问则需联系管理员接入授权。

- [crudless工具地址](#)
- [zero-json工具地址](#)
- [cg-api-cli工具地址](#)
- [dev-cli工具地址](#)

3. 构建crudless.yml

⋮ 直接基于业务理解定义应用服务, 各实体单元包括列表以详情字段, 还可以自动构建数据库SQL文件, 自动生成API代码。

可参考[crudless.yml](#)模板进行自定义编排, 其中**关键部分**解释如下所示:

- version: 版本号
- meta: 信息配置
 - author: 作者
 - createTime: 模块创建时间
 - updateTime: 模块更新时间
- ui: 视图配置
 - theme: 主题样式
 - layout: 布局格式
- auth: 授权配置
 - sso: 授权方式
 - reg: 是否可注册标识
 - bg: 背景图地址
- entries: 实体配置
 - menu_name (可自定义): 菜单栏配置
 - entry_name (可自定义): 菜单实体名称, 对应下述菜单入口
 - label: 菜单栏名称
 - sub: 菜单栏子选项
 - entry_name: 菜单实体名称, 对应下述菜单入口
 - label: 选项名称
- services: 服务配置
 - service_group_name: 服务组别名称
 - service_meta: 服务装配信息

- service_name: 服务名称
 - service_tag: 服务索引
 - service_title: 服务标题
 - service_category: 分类信息
 - target_domain: 服务装配地址
- api: 接口配置
 - pom: pom配置
 - version: 版本号
 - create_time: 创建时间
 - build_time: 更新时间
 - developer: 服务开发者名称
 - jar_repo_url: JAR包地址
 - jar_dependencies_url: JAR包依赖文件地址
- js: JS配置
 - create_time: 创建时间
 - build_time: 更新时间
 - developer: 前端开发者名称
 - js_page_url: 前端页面路径
- pages: 页面组配置
 - page (可自定义): 页面配置
 - api: 接口路径配置
 - description: 接口描述
 - entry_id: 菜单入口, 对应上述的entry_name
 - service_id: 服务标识名称
 - cg: 自动生成代码配置
 - master: 主表
 - slaves: 从表集
 - slaves_table_item: items
 - list: 列表配置
 - overall_actions: 列表操作
 - add_new: 新建列表项
 - export_xls: 导出Excel
 - import_xls: 导入Excel
 - print_pdf: 打印PDF
 - item_actions: 列表单项配置
 - edit: 编辑
 - delete: 删除
 - submit: 提交
 - form: 表单配置
 - columns: 表单列数
 - fields: 表单信息元素配置
 - sql_column_name: 对应sql列字段
 - lable: 对应名称
 - type: 表单填充类型
 - scope: 应用范围
 - sql: SQL配置

- type: 数据类型
- notnull: 是否不为空
- comment: 注释
- default: 默认值
- unique: 是否唯一
- props: 属性配置
 - autoSize: 大小配置
 - minRows: 最小行数
 - placeholder: 提示语
- rules: 规则
 - required: 是否必须
 - phone: 要求必须为手机

4. crudless 生成文件

根据 [2. 相关链接](#) 中的crudless工具地址拉取并使用其生成相应配置文件。

```
# 切换至crudless.yml同级目录下后执行crudless如下所示
$ crudless
outPath: ./test.json
outSQLPath: ./crudless.sql
$ crudless -h
crudless ## 默认找同目录的 crudless.yml, 并且默认输出前后端配置文件
crudless -f /path/to/crudless.yml ## 指定文件并输出所有配置文件
crudless -f /path/to/crudless.yml --json --sql --crud ## 按不同参数输出相应配置文件
```

执行完毕后将生成 `test.json` 和 `crudless.sql` 两份文件。

- 正式版本将生成三份文件:
 - test.json: 用于配合zero-json生成前端页面;
 - crudless.sql: 用于配合cg-api-cli生成后端代码;
 - *.crud.sql: 用于配合cg-api-cli生成SQL表之间的关系。

5. zero-json生成前端

Tips: 使用zero-json需先[初始化 web 后台管理项目](#)。

利用在上述 [4. crudless生成文件](#) 中生成 `*.json` 文件, 再配合zero-json即可快速生成前端代码。

假设现在需要生成一个名字叫 `myPage` 的 CRUD 页面, 同时新建一个目录存放 json 配置文件以方便管理, 例如 `./buildJSON`, 最后将上述生成的 `test.json` 文件放进里面

```
$ mv ./test.json ./buildJSON/test.json
```

因为后台管理项目的页面必须要放在 `web/src/pages` 目录下, 故而通过 `-o` 来将生成的文件重定向至 `./src/pages`

```
$ zero-json manage crud myPage -i ./buildJSON/test.json -o ./src/pages
```

至此页面就已经生成在 `./src/pages/myPage`

6. cg-api-cli生成后端

根据 [2. 相关链接](#) 中的cg-api-cli工具地址拉取并使用其生成相应配置文件，如下所示：

```
Usage: cg-api-cli <module> <sql> <cruds>
```

Example:

```
cg-api-cil test test.sql test.json
```

Parameters:

module	代码模块名
sql	sql文件路径
cruds	cruds文件路径

- Tips: 其中cruds.json文件目前仍需手动编排，具体请看下述。
- 执行命令后会新建一个项目名和module同名的maven项目，后续可用maven命令进行编译和测试。

• cruds.json编排

- 目前需要手动编写 `cruds.json` 文件，后续该文件将由crudless模块同时生成。

文件格式如下所示：

```
[
  {
    "master": "master", //主表名称
    "slaves": [ //从表名称，单表关系可不填。
      "slaveA",
      "slaveB"
    ]
  }
]
```

7. dev-cli运行后端

根据 [2. 相关链接](#) 中的dev-cli工具地址拉取、配置并使用其运行上述生成的后端代码。

- standalone
- used to run target/pack-1.0.0-standalone.jar locally first change path into the mvn project and type 'standalone'

```
# 在相同目录下执行standalone指令
# 使用init参数则运行SQL文件进行初始化
$ standalone init
```