多语言方案

## 需求分析

前言：航空箱项目目前存在多个客户端，且都有多语言需求，但在实际开发和项目管理过程中存在一些问题亟待解决，这些问题与目前前端多语言的实现方式有关。目前航空箱多语言的实现方案是使用的vue-i18n插件，在业务逻辑中放置“锚点”，如：

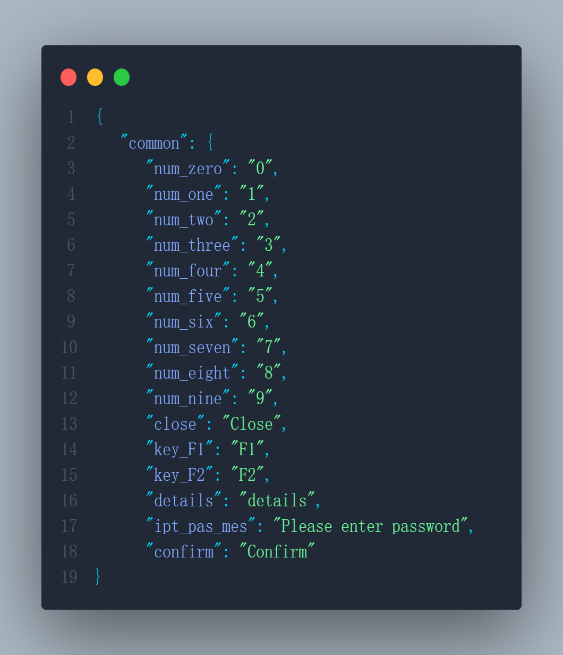


common.confirm可以理解为一个锚点，根据当前语言设置，会被替换为对应的文本，如：

中文



English



如上所示，UI中的锚点会被替换成当前语言环境下的对应json文件中的文本。

前端开发多语言时依赖于多语言文件（json、yml等等），该文件需要由翻译人员翻译成其他语种，开发过程中会有多种文件引入（全部or按需），衍生出如下问题：

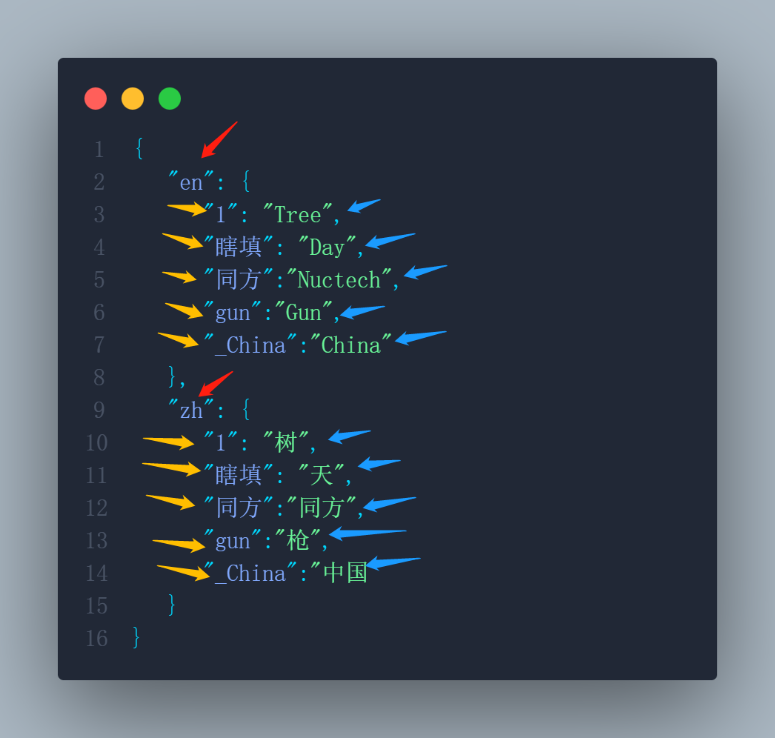
1.场景：如何将翻译人员返回的结果转换为可用的json（当待转换内容多时手动填写转换费时费力）。

翻译人员返回的文件多数为此种格式：

Excel 表：

|  |  |
| --- | --- |
| 中文 | 英文 |
| 树 | Tree |
| 中国 | China |
| 同方 | Nuctech |
| 枪 | Gun |
| 天 | Day |

前端使用方式：



红色箭头处表示的语言种类“zh”对应中文，“en”对应英文，字段可自定义；黄色箭头处表示的是“锚点”，可以不具有任何意义，但各个不同语种下锚点数量和名称应完全一致；蓝色箭头处表示的则是具体的文本信息，由专业翻译人员提供，不同语种下显示当前语种的文本信息。

那么如何从翻译人员返回的excel表转化成前端可用的json（其余格式类比实现）是本工具的实现目标。

2.场景：追加待翻译内容时如何将返回结果快速添加到现有json中。

3.场景：若要新增一种语言时如何快速导出一张excel表提供给翻译人员使用。