

yaml 与 Jinjia2语法语法

一．YAML编写技巧

为了能够很好的使用saltstack，我们有很多配置文件要编写，配置文件所使用的语法就是yaml语法．结构通过空格来展示，项目使用'-'来代表，键值对使用":"分割．

基本语法规则：

- 1．大小写敏感
- 2．使用缩进表示层级关系
- 3．缩进时不允许使用Tab键，只允许使用空格．
- 4．缩进的空格数目不重要，只要相同层级的元素左侧对齐即可
- 5．#表示注释，从这个字符一直到行尾都会被解析器忽略．

YAML支持的数据结构有三种

- 1．对象：键值对的集合，又称为映射（mapping） / 哈希（hashes） / 字典（dictionary）
- 2．数组：一组按次序排列的值，又称为序列（sequence） / 列表（list）
- 3．纯量：（scalars）单个的、不可再分的值

规则一：缩进

Yaml使用固定的缩进风格，表示数据的层级结构关系．saltstack需要每个缩进级别由两个空格组成，注意不要使用tabs．

规则二：冒号

Python的字典是键值对，字典的key在YAML中的表现形式是一个以:结尾的字符串．

- 1．key: value 对应的 Python字典格式为 {key: value},

另一种书写规则是：

- 2．key:
 value

转换成Python字典也是{key: value}

- 3．字典也是可以嵌套的

```
first:
  second:
    value
转换为Python字典形式为 {first: {second: value}}
```

规则三：短横线

想要表示列表项，使用短横线加一个空格，同一个列表里面的多个元素使用同样的缩进级别．

- first
- second
- third

以上转换为列表为 ["first", "second", "third"]

列表可以作为一个键值对的value，这在saltstack的配置文件中很常见，写法为：

```
my_directory:

- first

- second

- third
```

例：yaml与json转换

YAML文件内容

```
languages:
- Ruby
- Perl
- Python
websites:
  YAML: yaml.org
  Ruby: ruby-lang.org
  Python: python.org
  Perl: use.perl.org
```

转字典格式后为：

```
{ "languages": [
"Ruby",
"Perl",
"Python"
], "websites": {
"YAML": "yaml.org",
"Ruby": "ruby-lang.org",
"Python": "python.org",
"Perl": "use.perl.org"
}}
```

In []: