

04、配置文件

1、文件类型

H2

1.1、properties

同以前的properties用法

1.2、yaml

1.2.1、简介

YAML 是 "YAML Ain't Markup Language" (YAML 不是一种标记语言) 的递归缩写。在开发的这种语言时，YAML 的意思其实是："Yet Another Markup Language" (仍是一种标记语言)。

非常适合用来做以数据为中心的配置文件

1.2.2、基本语法

- key: value; kv之间有空格
- 大小写敏感
- 使用缩进表示层级关系
- 缩进不允许使用tab, 只允许空格
- 缩进的空格数不重要, 只要相同层级的元素左对齐即可
- '#'表示注释
- 字符串无需加引号, 如果要加, "与""表示字符串内容 会被 转义/不转义

1.2.3、数据类型

- 字面量: 单个的、不可再分的值。date、boolean、string、number、null


```
1 k: v
```

YAML

复制代码

- 对象: 键值对的集合。map、hash、set、object


YAML

 复制代码

```
1 行内写法:  k: {k1:v1,k2:v2,k3:v3}
2  #或
3  k:
4      k1: v1
5      k2: v2
6      k3: v3
```

- 数组：一组按次序排列的值。array、list、queue

YAML

 复制代码

```
1 行内写法:  k: [v1,v2,v3]
2  #或者
3  k:
4      - v1
5      - v2
6      - v3
```

1.2.4、示例

Java

 复制代码

```
1  @Data
2  public class Person {
3
4      private String userName;
5      private Boolean boss;
6      private Date birth;
7      private Integer age;
8      private Pet pet;
9      private String[] interests;
10     private List<String> animal;
11     private Map<String, Object> score;
12     private Set<Double> salarys;
13     private Map<String, List<Pet>> allPets;
14 }
15
16 @Data
17 public class Pet {
18     private String name;
19     private Double weight;
20 }
```

```
1  # yaml表示以上对象
2  person:
3      userName: zhangsan
4      boss: false
5      birth: 2019/12/12 20:12:33
6      age: 18
7      pet:
8          name: tomcat
9          weight: 23.4
10     interests: [篮球,游泳]
11     animal:
12         - jerry
13         - mario
14     score:
15         english:
16             first: 30
17             second: 40
18             third: 50
19         math: [131,140,148]
20         chinese: {first: 128,second: 136}
21     salarys: [3999,4999.98,5999.99]
22     allPets:
23         sick:
24             - {name: tom}
25             - {name: jerry,weight: 47}
26         health: [{name: mario,weight: 47}]
```

2、配置提示

自定义的类和配置文件绑定一般没有提示。

```
1      <dependency>
2          <groupId>org.springframework.boot</groupId>
3          <artifactId>spring-boot-configuration-processor</artifactId>
4          <optional>true</optional>
5      </dependency>
6
7
8  <build>
9      <plugins>
10         <plugin>
11             <groupId>org.springframework.boot</groupId>
12             <artifactId>spring-boot-maven-plugin</artifactId>
13             <configuration>
14                 <excludes>
15                     <exclude>
16                         <groupId>org.springframework.boot</groupId>
17                         <artifactId>spring-boot-configuration-processor</a
18                     </exclude>
19                 </excludes>
20             </configuration>
21         </plugin>
22     </plugins>
23 </build>
```