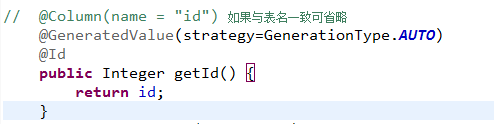
# @Column

## 概念

@Column标注也可置于属性的getter方法之前

结合@id使用

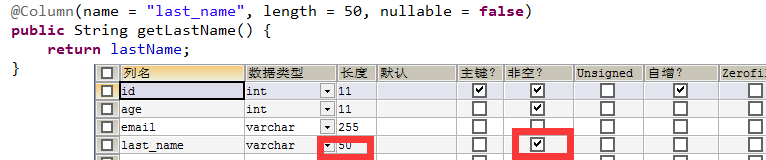
当实体的属性与其映射的数据库表的列不同名时需要使用@Column 标注说明，该属性通常置于实体的属性声明语句之前，还可与 @Id 标注一起使用。

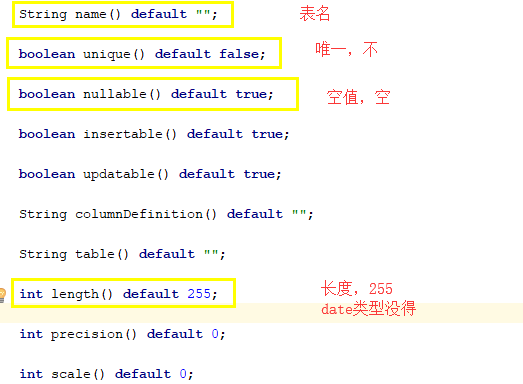


## 常规属性

### 概述

@Column 标注的常用属性是 name，用于设置映射数据库表的列名。此外，该标注还包含其它多个属性，如：unique 、nullable、length 等。





### name

定义了被标注字段在数据库表中所对应字段的名称；

### unique

表示该字段是否为唯一标识，默认为false。如果表中有一个字段需要唯一标识，则既可以使用该标记，也可以使用@Table标记中的@UniqueConstraint。

### nullable

表示该字段是否可以为null值，默认为true。

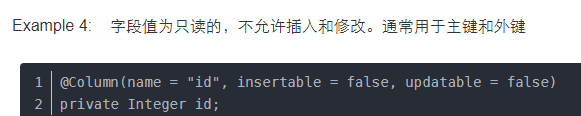
### insertable

表示在使用“INSERT”脚本插入数据时，是否需要插入该字段的值。

### 只读updatable

updatable

表示在使用“UPDATE”脚本插入数据时，是否需要更新该字段的值。insertable和updatable属性一般多用于只读的属性，例如主键和外键等。这些字段的值通常是自动生成的。

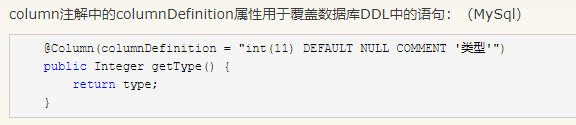


### columnDefinition（几乎不用）--注释

表示创建表时，该字段创建的SQL语句，一般用于通过Entity生成表定义时使用。（也就是说，如果DB中表已经建好，该属性没有必要使用。）

可替代所有

@Column 标注的 columnDefinition 属性: 表示该字段在数据库中的实际类型.通常 ORM 框架可以根据属性类型自动判断数据库中字段的类型,但是对于Date类型仍无法确定数据库中字段类型究竟是DATE,TIME还是TIMESTAMP.此外,String的默认映射类型为VARCHAR, 如果要将 String 类型映射到特定数据库的 BLOB 或TEXT 字段类型.



### table

表示当映射多个表时，指定表的表中的字段。默认值为主表的表名。

### length

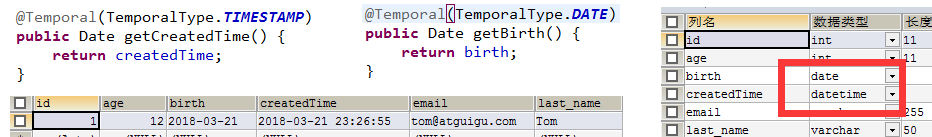
表示字段的长度，当字段的类型为varchar时，该属性才有效，默认为255个字符。

### precision和scale

precision属性和scale属性表示精度，当字段类型为double时，precision表示数值的总长度，scale表示小数点所占的位数

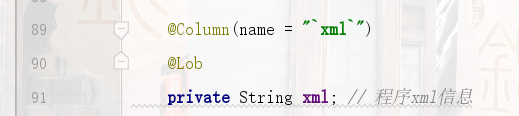
## @Temporal（日期）

在核心的 Java API 中并没有定义 Date 类型的精度(temporal precision). 而在数据库中,表示 Date 类型的数据有 DATE, TIME, 和 TIMESTAMP 三种精度(即单纯的日期,时间,或者两者 兼备). 在进行属性映射时可使用@Temporal注解来调整精度.

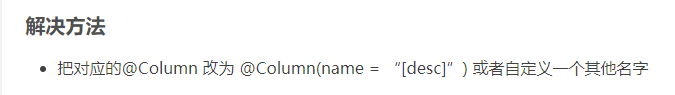


## (重点)兼容关键字

https://blog.csdn.net/qq\_19671173/article/details/76782618



或者

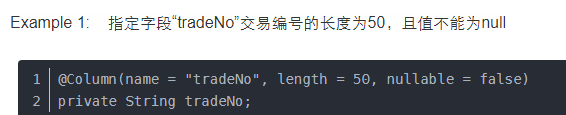


# 类型

主键和外键

## String(2+1+1)

Name,length, nullable unque



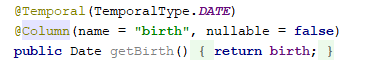
## Integer（2+1+1）



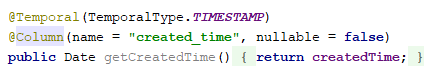
## 日期时间（2）

不要length

### Date

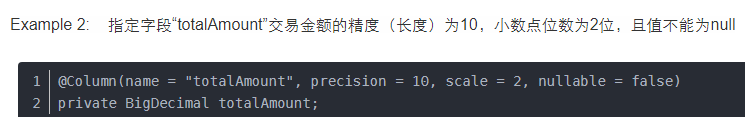


### Datetime



## 小数

### BigDecimal（了解）



## 大文本

### Text（了解）

