

# Mongoose 数据校验

主讲教师：（大地）

合作网站：[www.itying.com](http://www.itying.com)

一、Mongoose 校验参数.....	1
二、Mongoose 自定义的验证器.....	2

## 一、Mongoose 校验参数

required: 表示这个数据必须传入

max: 用于 Number 类型数据, 最大值

min: 用于 Number 类型数据, 最小值

enum:枚举类型, 要求数据必须满足枚举值    enum: ['0', '1', '2']

match:增加的数据必须符合 match（正则）的规则

maxlength: 最大值

minlength: 最小值

```
var UserSchema = new mongoose.Schema({  
  name:{  
    type:String,  
    required: true,  
  
  },
```

```
age: {
  type: Number,
  // 是否必须的校验器
  required: true,
  // 数字类型的最大值校验器
  max: 120,
  // 数字类型的最小值校验器
  min: 0
},
status: {
  type: String,
  // 设置字符串的可选值
  enum: ['0', '1', '2']
},
phone: {
  type: Number,
  match: /^\\d{11}$/
},
desc: {
  type: String,
  maxlength: 20,
  minlength: 10
}
});
```

## 二、Mongoose 自定义的验证器

在缺省情况下创建的索引均不是唯一索引。下面的示例将创建唯一索引，如：

```
var UserSchema = new mongoose.Schema({
  name: {
    type: String,
    required: true,

  },
  age: {
    type: Number,
    // 是否必须的校验器
    required: true,
    // 数字类型的最大值校验器
```

```
max: 120,  
// 数字类型的最小值校验器  
min: 0  
,  
status: {  
  type: String,  
  // 设置字符串的可选值  
  enum: ['0', '1', '2']  
},  
phone: {  
  type: Number,  
  match: /^\\d{11}$/  
},  
desc: {  
  type: String,  
  // 自定义的验证器，如果通过验证返回 true，没有通过则返回 false  
  validate: function(desc) {  
    return desc.length >= 10;  
  }  
}  
});
```