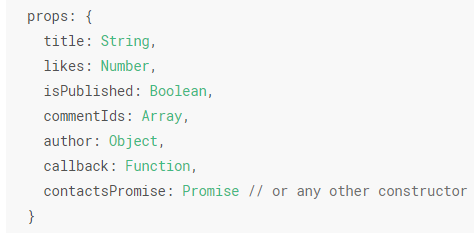
# Prop

## Prop类型

你可以使用对象形式列出prop，这些属性的名称和值分别是prop各自的名称和类型。



## **传递静态或动态prop**



## 传入一个数字

对于数字的传值，我们都需要使用v-bind进行绑定。



## 传入一个布尔值



## 传入一个数组



## 传入一个对象（同上）

# 单向数据流

所有的prop都使得父子prop之间形成了一个单向下行绑定：父级prop的更新会向下流动到子组件中，但是反过来则不行，这样会防止从子组件以外改变父组件的状态，从而导致你的应用的数据流向难以理解。

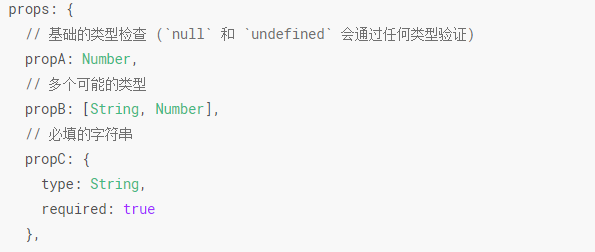
### 子组件处理传输过来的prop数据





## Prop验证

为组件的prop指定验证要求。





## 类型检查

为prop指定验证要求。

Type可以是下列原生构造函数的一个，String,Number,Boolean,Array,Object,Date,Function,Symbol

此外，type还可以是一个自定义的构造函数，并且可以通过instanceof来进行检查确认。例如，给定下列的构造函数。

## 非prop的特性

一个非prop特性是指传向一个组件，但是该组件并没有相应prop定义的特性。

因为显式定义的prop适用于向一个子组件传入信息，然而组件库的坐着并不总能遇见组件会被用于怎样的场景。这也是为什么组件可以接受任意的特性，而这些特性会被添加到这个组件的根元素上。

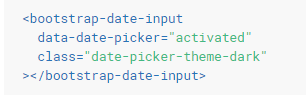


## 替换/合并已有的特性

例如一个组件内部模板是这样的

IMG_256

为了给我们的日期选择器插件制定一个主体，我们可能需要像这样添加一个特别的类名



这种情况下，我们定义了两个不同的class值，class和style特性稍微智能，会将两个值合并起来。

## 禁用特性继承

如果你不希望组件的根元素继承特性，可以设置，

