1.

v-bind动态绑定的本地图片src需要放在static这个文件夹中

2.

<!-- elementui组件的prop不是string类型的时候需要v-bind绑定 不然会报错 -->

3.

 <!-- props有标题、背景颜色等 通过一个对象传进去-->

    <index-content v-bind="content\_01"></index-content>

    <index-content v-bind="content\_02"></index-content>

 data() {

    return {

      content\_01: {

        title\_01: "第一阶段课程",

        title\_02: "Java设计和编程思想",

        container\_bg: "#fff",

        title\_bg: "url('/static/image/strong02.png')",

        title\_01\_col: "#000",

        title\_02\_col: "#000"

      }

}

4.

<!-- 通过props、v-bind、style和data里的样式对象 控制组件的样式(也可写行内style解决) -->

    <div class="index-content-container" :style="bgColor">

 data() {

        return {

            bgColor: {

                backgroundColor: this.container\_bg

            }

        }

    }

 props: {

        container\_bg: {

            type: String,

            default: '#fff'

        }

1. 使用/deep/或>>>深度选择器改变element-ui的默认样式；使用scoped 后，父组件的样式将不会渗透到子组件中，使用深度选择器则可以从父组件渗透到子组件

/deep/ .el-carousel\_\_indicator--horizontal:hover

              .el-carousel\_\_button

                transition: all 0.2s

                height: 7px

6.

### **二、关于子组件的根元素**

　　使用了scoped属性之后，父组件的style样式将不会渗透到子组件中，然而**子组件的根节点元素**会**同时被设置了scoped的父css样式和设置了scoped的子css样式影响**，这么设计的**目的是父组件可以对子组件根元素进行布局**。

//子组件

<template>

    <div class="bottom-container">

        <p>Copyright </p>

    </div>

</template>

//调用子组件

<bottom></bottom>

//父组件中的样式修改子组件样式

.bottom-container

    background-color #ccc

    /deep/ p

            font-size: 30px