内蒙古农业大学

学生实验报告

院 系： 计算机与信息工程学院

专 业： 计算机科学与技术

年 级： 2021级

课程名称： 嵌入式应用开发

学 号： 2021122156404

姓 名： 周烨

指导教师： 阿斯雅

2024年 3 月 20 日

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 院系 | 计算机院信息工程学院 | 年级 | 2021级 | 学号 | **2021122156404** |
| 专业 | 计算机科学与技术 | | | 姓名 | **周烨** |
| 实验名称 | 实验二 组件字典 | | | 实验室 | 414 |
| 实  验  目  的 | 1. 了解常用基础组件 2. 了解容器组件 3. 数组的使用 4. ForEach用法 5. 数据源使用 6. List容器 7. Tabs用法 8. 组件封装 | | | | |
| 实验题目 | 组件字典 | | | | |
| 实验代码 | export const ComponentData: any[] =[  {  'catId': 1,  'catName':'基础组件',  'id': 1,  'compName':'Text'  },  {  'catId': 1,  'catName': '基础组件',  'id': 2,  'compName': 'TextInput'  },  {  'catId': 1,  'catName':'基础组件',  'id': 3,  'compName':'TimePicker'  },  {  'catId': 1,  'catName':'基础组件',  'id': 4,  'compName':'Button'  },  {  'catId': 1,  'catName':'基础组件',  'id': 5,  'compName':'Checkbox'  },  {  'catId': 1,  'catName':'基础组件',  'id': 6,  'compName':'CheckboxGroup'  },  {  'catId': 1,  'catName':'基础组件',  'id': 7,  'compName':'Radio'  },  {  'catId': 1,  'catName':'基础组件',  'id': 8,  'compName':'LoadingProgress'  },  {  'catId': 1,  'catName':'基础组件',  'id': 9,  'compName':'Image'  },  {  'catId': 1,  'catName':'基础组件',  'id': 10,  'compName':'Progress'  },  {  'catId': 1,  'catName':'基础组件',  'id': 11,  'compName':'Select'  },  {  'catId': 1,  'catName':'基础组件',  'id': 12,  'compName':'Slider'  },  {  'catId': 2,  'catName':'容器组件',  'id': 13,  'compName':'Column'  },  {  'catId': 2,  'catName':'容器组件',  'id': 14,  'compName':'Row'  },  {  'catId': 2,  'catName':'容器组件',  'id': 15,  'compName':'Panel'  },  {  'catId': 2,  'catName':'容器组件',  'id': 16,  'compName':'swiper'  },  {  'catId': 3,  'catName':'媒体组件',  'id': 17,  'compName':'Video'  },  {  'catId': 4,  'catName':'弹窗',  'id': 18,  'compName':'AlertDialog'  }  ,  {  'catId': 4,  'catName':'弹窗',  'id': 19,  'compName':'DatePickerDialog'  }  ,  {  'catId': 4,  'catName':'弹窗',  'id': 20,  'compName':'TextPickerDialog'  } ] export class ComponentBean {  catId: number; //组件分类 ID  catName: string; //组件分类名  id: number; //组件 id  compName: string ; //组件名  //构造方法  constructor(catId: number, catName: string, id: number, compName: string) {  this.catId = catId  this.catName = catName  this.id = id  this.compName = compName  } } export function initComponentsData():Array<ComponentBean> {  let componentArray: Array<ComponentBean> = []  //迭代原始数据封装成 bean 并存储到数组中，然后返回  ComponentData.forEach((item)=>{  componentArray.push(new ComponentBean(item.catId,item.catName,item.id,item.compName))  })  return componentArray; }  //import { ComponentBean, initComponentsData } from '../model/ComponentModel' import router from '@ohos.router'; import { initComponentsData } from '../model/ComponentModel'; import { ComponentBean } from '../model/ComponentModel'; @Entry @Component struct Index {  build() {  Row() {  Column() {  Text('ArkTS 常用组件字典')  .fontSize(30)  .fontWeight(700)  .margin({top:15,bottom:5})  Tabs() {  TabContent(){  BasicComponentList()  }.tabBar('基础组件')  TabContent(){  ContainerComponentList()  }.tabBar('容器组件')  TabContent(){  MediaComponentList()  }.tabBar('媒体组件')  TabContent(){  DailogComponentList()  }.tabBar('弹窗组件')  }  .barWidth('100%')  .barHeight(50)  .scrollable(true)  .height('70%')  .backgroundColor('#ffa9bacb')  }  .width('100%')  }  .height('100%')  } } @Component struct BasicComponentList {  private listArray: Array<ComponentBean> = []  aboutToAppear() {  var newList = initComponentsData()  this.listArray = newList.slice(0, 12)  }  build() {  Column(){  List({ space: 5 }) {  ForEach(this.listArray,item =>{  ListItem(){  Button(item.compName)  .type(ButtonType.Normal)  .width('100%')  .height(50)  .fontColor(0x000000)  .backgroundColor(0xffffff)  .onClick(() => {  router.pushUrl({  url: 'pages/ComponentDetails',  params: {  name: item.compName  }  })  })  }  },item=>item.id.toString())  }  }  } } //容器组件列表 @Component struct ContainerComponentList {  private listArray: Array<ComponentBean> = []  aboutToAppear() {  var newList = initComponentsData()  this.listArray = newList.slice(12, 16)  }  build() {  Column(){  List({ space: 5 }) {  ForEach(this.listArray,item =>{  ListItem() {  Button(item.compName)  .type(ButtonType.Normal)  .width('100%')  .height(50)  .fontColor(0x000000)  .backgroundColor(0xffffff)   }  },item=>item.id.toString())  }  }  } } @Component struct MediaComponentList {  private listArray: Array<ComponentBean> = []  aboutToAppear() {  var newList = initComponentsData()  this.listArray = newList.slice(16, 17)  }  build() {  Column(){  List({ space: 5 }) {  ForEach(this.listArray,item =>{  ListItem(){  Button(item.compName)  .type(ButtonType.Normal)  .width('100%')  .height(50)  .fontColor(0x000000)  .backgroundColor(0xffffff)  }  },item=>item.id.toString())  }  }  } } @Component struct DailogComponentList {  private listArray: Array<ComponentBean> = []  aboutToAppear() {  var newList = initComponentsData()  this.listArray = newList.slice(17, 20)  }  build() {  Column(){  List({ space: 5 }) {  ForEach(this.listArray,item =>{  ListItem(){  Button(item.compName)  .type(ButtonType.Normal)  .width('100%')  .height(50)  .fontColor(0x000000)  .backgroundColor(0xffffff)   }  },item=>item.id.toString())  }  }  } }  import { MyText } from '../common/components/MyText'; import router from '@ohos.router'; import { MyTextInput } from '../common/components/MyTextInput'; @Entry @Component struct ComponentDetails {  @State compnentName: string = router.getParams()['name']   build() {   Column() {  if(this.compnentName=='Text'){  MyText()  }  if(this.compnentName == 'TextInput'){  MyTextInput()  }   }  } } | | | | |

|  |  |
| --- | --- |
| 实  验  步  骤 | 运行截图：  585483ab52e0b53b97698c05e1711431ae39ec19cef1b196344d73f1e5b11d  ae794b4ba8012b72a43dc3afb3351bc |
| 心  得  体  会 | 1. 了解并学习了mvc模型的概念 2. @Entry（只能出现一次）后跟@Component才能创建页面，其余的@Component为普通渲染组件 3. 加上Export 关键字后，相关函数才可以对外使用 4. ForEach（）为页面循环渲染函数，与forEach()不同 5. 学习了组件中导入路由和接收参数的方法：使用import router from '@ohos.router'和@State compnentName: string = router.getParams()['name']配合onClick(() =>函数使用 6. 可以通过把界面上的文字字符串添加到资源文件中，通过引用资源的方式显示。 |
| 成  绩  评  定 | 教师签名：  年 月 日 |