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 5月 22 日

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 院系 | 计算机院信息工程学院 | 年级 | 21级 | 学号 | **2021122156404** |
| 专业 | 计算机科学与技术 | | | 姓名 | **周烨** |
| 实验名称 | 实验五 关系型数据库实验 | | | 实验室 | 414 |
| 实  验  目  的 | 1. 掌握数据库和表的创建。 2. 掌握数据库表数据的增删改查。 3. 掌握数据库的删除。 | | | | |
| 实验题目 | 关系型数据库实战练习  扩展内容：1.可以根据自己的想法设计数据库和数据表。 | | | | |
| 实验代码 | import relationalStore from '@ohos.data.relationalStore'  @Extend(Button) function myButton() {  .backgroundColor(0x0080ff)  .width('90%')  .height('5%')  .borderRadius(50)  .type(ButtonType.Normal) }  const STORE\_CONFIG={  name:'student.db',  securityLevel:relationalStore.SecurityLevel.S1 }; const SQL\_CREATE\_TABLE = 'CREATE TABLE IF NOT EXISTS student(id INTEGER PRIMARY KEY AUTOINCREMENT, name TEXT NOT NULL, age INTEGER)'; const MyTAG = '----->';  @Entry @Component struct Index {  @State rdbStore:relationalStore.RdbStore=null  @State message: string =""  @State MyTAG:string ='----->'   aboutToAppear(){  relationalStore.getRdbStore(getContext(),STORE\_CONFIG,(err,store)=>{  if (err) {  this.message+=`${this.MyTAG}初始化数据库失败,Failed to get RdbStore.Code :${err.code},message:${err.message}\n`  console.error(this.message)  return;  }   this.message+=`${this.MyTAG}成功初始化数据,Succeeded in getting RdbStore.\n`  console.info(this.message)  this.rdbStore=store;  })  }   build() {  Scroll(){  Column({space:15}){  Text('DataBase').fontSize(40).fontColor(0x0080ff)  Button('创建数据表').myButton().onClick(()=>{  this.rdbStore.executeSql(SQL\_CREATE\_TABLE);  this.message+=`${this.MyTAG}创建数据表成功\n`  console.info(this.message)  })  Button('插入数据表').myButton().onClick(()=>{  const valueBucket = {  'name': '张三',  'age': 18  };  this.rdbStore.insert('student', valueBucket, (err, rowId) => {  if (err) {  this.message += `${MyTAG} 插入数据失败. Code:${err.code}, message:${err.message}\n`  console.error(this.message)  return;  }  this.message += `${MyTAG} 插入数据成功. rowId:${rowId}\n`  console.info(this.message)  })  })  Button('批量插入').myButton().onClick(()=>{  let valueBucketList = []  for (let i = 0;i < 10; i++) {  let valueBucket = {  'name': '张三' +i,  'age': 18+i  };  valueBucketList[i] = valueBucket  }  this.rdbStore.batchInsert("student", valueBucketList, (err, rowId) => {  if (err) {  this.message += `${MyTAG} 批量插入数据失败. Code:${err.code}, message:${err.message}\n`  console.error(this.message)  return;  }  this.message += `${MyTAG} 批量插入数据成功. rowId:${rowId}\n`  console.info(this.message)  })  })  Button('修改数据').myButton().onClick(()=>{  const newValueBucket = {  'name': '李四',  'age': 20  };  let predicates = new relationalStore.RdbPredicates('student');  predicates.equalTo('name', '张三');  this.rdbStore.update(newValueBucket, predicates, (err, rows) => {  if (err) {  this.message += `${MyTAG} 更新数据失败. Code:${err.code}, message:${err.message}\n`  console.error(this.message)  return;  }  this.message += `${MyTAG} 更新数据成功. row count: ${rows}\n`  console.info(this.message)  })  })  Button('删除数据（先修改）').myButton().onClick(()=>{  let predicates = new relationalStore.RdbPredicates('student');  predicates.equalTo('name', '李四');   this.rdbStore.delete(predicates, (err, rows) => {  if (err) {  this.message += `${MyTAG}删除数据失败. Code:${err.code}, message:${err.message}\n`  console.error(this.message);  return;  }  this.message += `${MyTAG}删除数据成功，Delete rows: ${rows}\n`  console.info(this.message)  })  })  Button('查询').myButton().onClick(()=>{  let predicates =new relationalStore.RdbPredicates('student')  predicates.equalTo('name','张三')  this.rdbStore.query(predicates,['id','name','age'],(err,resultSet)=>{  if (err) {  this.message+=`${MyTAG}条件查询失败. Code:${err.code},message:${err.message}\n`  console.error(this.message)  return  }  this.parseData(resultSet)  })  })  Button('查询全部').myButton().onClick(()=>{  let predicates=new relationalStore.RdbPredicates('student')  this.rdbStore.query(predicates,['id','name','age'],(err,resultSet)=>{  if (err) {  this.message+=`${MyTAG}查询全部失败. Code:${err.code},message:${err.message}\n`  console.error(this.message)  return  }  this.parseData(resultSet)  })  })  Button('删除数据库').myButton().onClick(()=>{  AlertDialog.show({  title: "提示",  message: "确认要删除数据库吗?",  primaryButton: {  value: "确认",  action: () => {  relationalStore.deleteRdbStore(getContext(), 'student.db', (err) => {  if (err) {  this.message += `${MyTAG}删除数据库失败. Code:${err.code}, message:${err.message}\n`  console.error(this.message);  return;  }  this.message += `${MyTAG}删除数据库成功，Succeeded in deleting RdbStore.\n`  console.info(this.message);  });  }  },  secondaryButton: {  value: "取消",  action: () => {  this.message += `${MyTAG}您点击了取消按钮，取消删除数据库\n`  console.info(this.message)  }  }  })  })  TextArea({text:'执行结果：\r\n'+this.message}).margin({top:10})  }.width('100%').alignItems(HorizontalAlign.Center)  }  }  parseData(resultSet: relationalStore.ResultSet) {  console.info(`${this.MyTAG}ResultSet column names:${resultSet.columnNames}\n`);  console.info(`${this.MyTAG}ResultSet column count:${resultSet.columnCount}\n`);   let count = resultSet.rowCount  if (count == 0) {  this.message += `${this.MyTAG}查询结果为空，count=${count}\n`  console.info(this.message)  return  }  this.message += `${this.MyTAG}查询成功，count=${count}，开始解析......`  console.info(this.message)   resultSet.goToFirstRow() //开始第一条  for (let i = 0; i < count; i++) {  let id = resultSet.getDouble(resultSet.getColumnIndex("id"))  let name = resultSet.getString(resultSet.getColumnIndex("name"))  let age = resultSet.getDouble(resultSet.getColumnIndex("age"))   //输出结果看一下  let msg = `id=${id}, name=${name}, age=${age}`  this.message += "\r\n" + msg+'\n'  console.info(this.message)  //继续下一条  resultSet.goToNextRow()  }   this.message += `${this.MyTAG}查询成功，count=${count}，解析结束.\n`  console.info(this.message)  } } | | | | |

|  |  |
| --- | --- |
| 实  验  步  骤 | 9e550c3d7121951e4be5e464e667e07运行截图：d8f49f20d972acefd907a3916871220 |
| 心  得  体  会 | 在实验过程中，我学会了使用调试工具和查看日志信息来定位和解决问题。这对于开发过程中快速发现问题并修复非常有帮助。  同时，更深一步理解了数据库的操作和使用，对SQL语言的掌握更为熟练，对课上学习的抽象数据库知识有了更深层次的理解  能将SQL数据库的知识运用到实际的数据操作中，进行数据表的创建和数据的增删查改。 |
| 成  绩  评  定 | 教师签名：  年 月 日 |