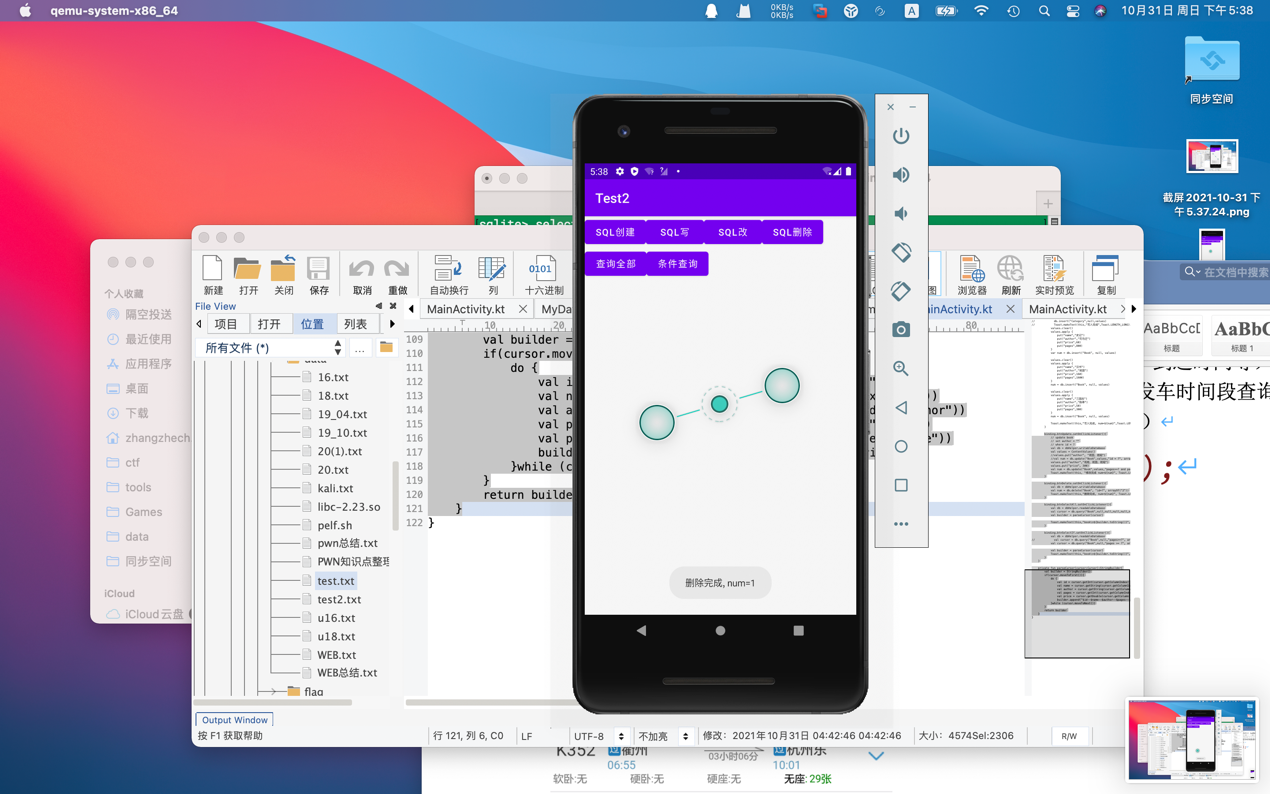
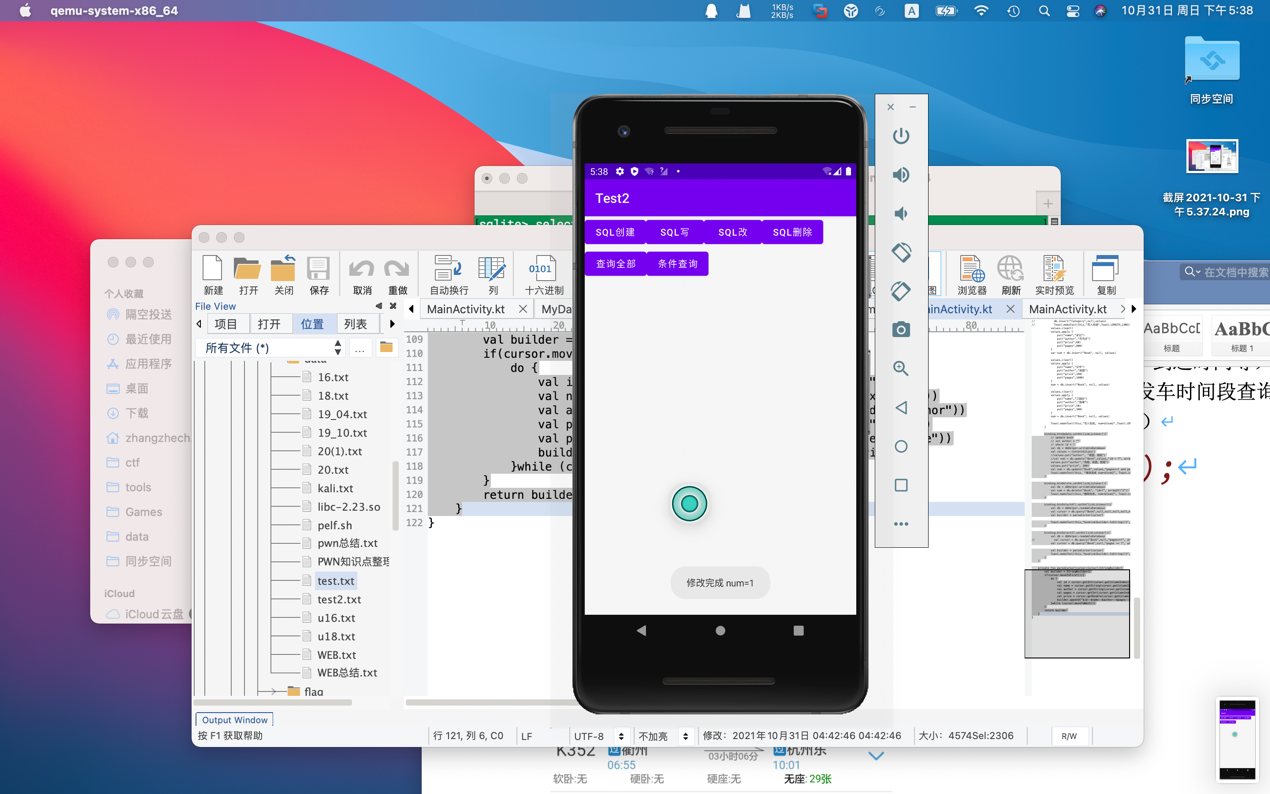
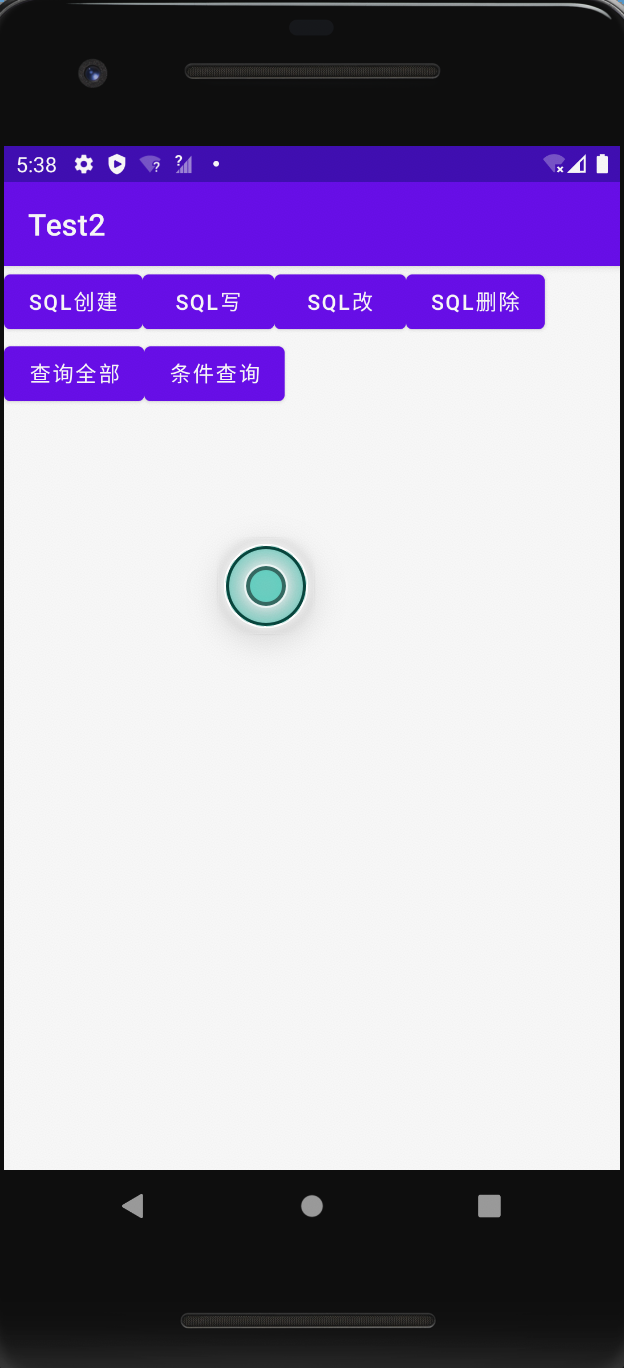
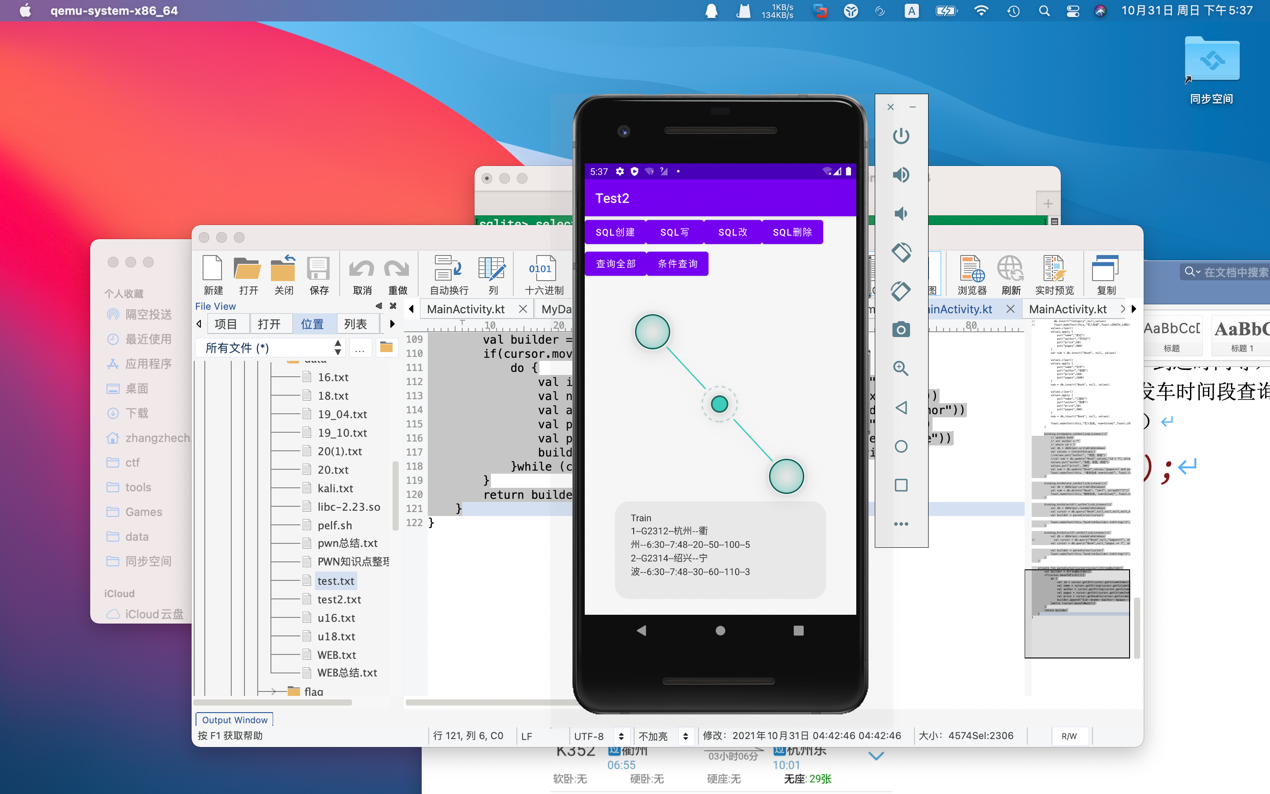
1 实验内容

在Sqlite中建表，保存车次基本信息（起点站、终点站、发车时间、到达时间等），

完成表的添加、修改、删除、查询

2 实验效果



3 实验实现过程

class MainActivity : AppCompatActivity() {

private val dbHelper = MyDataBaseHelper(this,"Train",1)

@SuppressLint("Range")

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

val binding = ActivityMainBinding.inflate(layoutInflater)

setContentView(binding.root)

//数据库创建

binding.btnCreateDB.setOnClickListener(){

dbHelper.writableDatabase

}

//添加数据

binding.btnInsert.setOnClickListener(){

//定位数据库

val db = dbHelper.writableDatabase

//准备数据

val values = ContentValues()

values.clear()

values.apply {

put("name","G2312")

put("startStation","杭州")

put("endStation","衢州")

put("startTime","6:30")

put("endTime","7:48")

put("shangWu",20)

put("firstNumber",50)

put("secondNumber",100)

put("wuZuo",5)

}

var num = db.insert("Train", null, values)

values.clear()

values.apply {

put("name","G2314")

put("startStation","金华")

put("endStation","宁波")

put("startTime","6:30")

put("endTime","7:48")

put("shangWu",30)

put("firstNumber",60)

put("secondNumber",110)

put("wuZuo",3)

}

num = db.insert("Train", null, values)

Toast.makeText(this,"写入完成, num=${num}",Toast.LENGTH\_LONG).show()

}

binding.btnUpdate.setOnClickListener(){

// update train

// set startStation = ""

// where id = ?

val db = dbHelper.writableDatabase

val values = ContentValues()

//values.put("author", "班固，班昭")

//val num = db.update("Book",values,"id = ?", arrayOf("2"))

values.put("startStation","绍兴")

val num = db.update("Train",values,"id=?", arrayOf("2"))

Toast.makeText(this, "修改完成 num=${num}", Toast.LENGTH\_LONG).show()

}

binding.btnDelete.setOnClickListener(){

val db = dbHelper.writableDatabase

val num = db.delete("Train", "id=?", arrayOf("2"))

Toast.makeText(this,"删除完成, num=${num}", Toast.LENGTH\_LONG).show()

}

binding.btnSelectAll.setOnClickListener(){

val db = dbHelper.readableDatabase

val cursor = db.query("Train",null,null,null,null,null,null,null)

val builder = parseCursor(cursor)

Toast.makeText(this,"Train\n${builder.toString()}",Toast.LENGTH\_LONG).show()

}

binding.btnSelectIf.setOnClickListener(){

val db = dbHelper.readableDatabase

// val cursor = db.query("Book",null,"pages>=?", arrayOf("100"),null,null,null)

val cursor = db.query("Train",null,"id = ?", arrayOf("1"),null,null,null)

val builder = parseCursor(cursor)

Toast.makeText(this,"Train\n${builder.toString()}",Toast.LENGTH\_LONG).show()

}

}

// put("name","G2314")

// put("startStation","金华")

// put("endStation","宁波")

// put("startTime","6:30")

// put("endTime","7:48")

// put("shangWu",30)

// put("firstNumber",60)

// put("secondNumber",110)

// put("wuZuo",3)

private fun parseCursor(cursor:Cursor):StringBuilder{

val builder = StringBuilder()

if(cursor.moveToFirst()){

do {

val id = cursor.getInt(cursor.getColumnIndex("id"))

val name = cursor.getString(cursor.getColumnIndex("name"))

val startStation = cursor.getString(cursor.getColumnIndex("startStation"))

val endStation = cursor.getString(cursor.getColumnIndex("endStation"))

val startTime = cursor.getString(cursor.getColumnIndex("startTime"))

val endTime = cursor.getString(cursor.getColumnIndex("endTime"))

val shangWu = cursor.getInt(cursor.getColumnIndex("shangWu"))

val firstNumber = cursor.getInt(cursor.getColumnIndex("firstNumber"))

val secondNumber = cursor.getInt(cursor.getColumnIndex("secondNumber"))

val wuZuo = cursor.getInt(cursor.getColumnIndex("wuZuo"))

builder.append("$id--$name--$startStation--$endStation--$startTime--$endTime--$shangWu--$firstNumber--$secondNumber--$wuZuo\n")

}while (cursor.moveToNext())

}

return builder

}

}