**实 验 报 告**

**课程名称：** **移动应用开发**

**学 院： 计算机科学与工程学院**

**专 业：计算机科学与技术** **班 级： 5班**

**姓 名： 尹浩男 学 号：201801051827**

**年 月 日**

**山 东 科 技 大 学 教 务 处 制**

**实 验 报 告**

**页**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **组 别** |  | **姓 名** | **尹浩男** | **同组实验者** |  |
| **实验项目**  **名称** | Activity | | **实验日期** |  | |
| **教师评语** |  | | | | |
| **实验成绩：** | | | **指导教师（签名）：**  **年 月 日** | | |
| 一、实验目的  * 熟练掌握Android开发环境搭建和Android Studio的使用； * 了解Android Studio中项目的文件目录结构； * 掌握基于Activity的Android APP的设计和编码方法，根据需求完成APP界面和Activity之间的数据传递功能。  二、实验内容  * Android Studio安装与基本使用方法； * 设计实现一个判断题题库Android APP，其基本功能包括：1）启动后显示第一个题目的界面，如图1-1所示，界面中包含了题目内容、“正确”、“错误”、“查看答案”和“下一个”按钮。点击“正确”或“错误”表示用于给出的答案，点击“下一个”进入下一个判断题；2）用户点击“查看”答案后，显示如图1-2所示的界面，其中显示提示信息“确定看答案？”和“答案”按钮。用户点击“答案”后，界面显示该题目的标准答案，如图1-3所示；3）用户如果在题目界面未选择答案而直接进入答案界面并查看了答案，再次回到题目界面（图1）点击“正确”或“错误”按钮，界面提示信息，如图1-4所示的“作弊是不对的”。  |  |  | | --- | --- | | 图1-1 题目界面 | 图1-2 查看“答案”界面 | | 图1-3 查看答案 | 图1-4 提示信息界面 |  三、实验要求  * 实现上述的基本功能要求。说明：题库内容可以写在代码中。在此基础上，可以进一步优化界面，也可以进一步添加其他功能。 * 实验报告提交内容：1）结合系统界面（截图）介绍APP功能；2）核心实现代码（Java或Kotlin代码）和布局文件等。  四、实验代码4.1 MainActivity4.1.1 功能介绍   **主界面的布局如左图所示 采用线性布局**  **由1个textView和4个button组成**  **其中 textView用于显示题目**  **而4个按钮分别代表着正确、错误、**  **查看答案、和下一个题目**  **点击查看答案按钮会启动AnswerActivity**  **显示题目的答案**  **而当点击下一个到达题目的最后一个时，会启动**  **FinishActivity 会显示最后的得分** 4.1.2 layout代码 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"      android:layout\_width="match\_parent"      android:layout\_height="wrap\_content"      android:layout\_gravity="center\_vertical"      android:orientation="vertical">      <LinearLayout          android:layout\_width="match\_parent"          android:layout\_height="wrap\_content"          android:orientation="horizontal">          <TextView              android:gravity="center\_horizontal"              android:id="@+id/question"              android:layout\_width="match\_parent"              android:layout\_height="wrap\_content"              android:text="Hello World!" />      </LinearLayout>      <LinearLayout          android:gravity="center\_horizontal"          android:layout\_width="match\_parent"          android:layout\_height="wrap\_content">          <Button              android:id="@+id/right"              android:layout\_width="wrap\_content"              android:layout\_height="wrap\_content"              android:text="正确"/>          <Button              android:id="@+id/wrong"              android:layout\_width="wrap\_content"              android:layout\_height="wrap\_content"              android:text="错误"/>      </LinearLayout>      <LinearLayout          android:gravity="center\_horizontal"          android:layout\_width="match\_parent"          android:layout\_height="wrap\_content"          android:orientation="vertical">          <Button              android:id="@+id/answer"              android:layout\_width="wrap\_content"              android:layout\_height="wrap\_content"              android:text="查看答案"/>          <Button              android:id="@+id/next"              android:layout\_width="wrap\_content"              android:layout\_height="wrap\_content"              android:text="下一个"/>      </LinearLayout>  </LinearLayout> 4.1.3 mainActivity代码 **整个代码的核心思想就是处理四个按钮的点击事件，以下是核心代码** 4.1.3.1 正确或者错误按钮的点击事件 **选择正确或者错误会通过toast给用户以反馈，告诉用户的选择正确性**  **// 正确按钮的点击事件**  **right.setOnClickListener {**  **if(!checkAnswer[currentIndex]) { // 如果没有勾选过或者查看答案 才可以判断**  **if ( fileContentList[currentIndex].answer){**  **Toast.makeText(this,"你答对了",Toast.LENGTH\_SHORT).show()**  **listOfEachQuestion[currentIndex] = true**  **}**  **else{**  **this.checkAnswer[currentIndex] = true**  **Toast.makeText(this,"你答错了",Toast.LENGTH\_SHORT).show()**  **}**  **}else**  **Toast.makeText(this,"你已经知道答案了,作弊是不对的喔",Toast.LENGTH\_SHORT).show()**  **}**  **// 错误按钮的点击事件**  **wrong.setOnClickListener {**  **//question.text = "wrong"**  **if(!checkAnswer[currentIndex]) { // 如果没有勾选过或者查看答案 才可以判断**  **if (!fileContentList[currentIndex].answer) {**  **Toast.makeText(this, "你答对了", Toast.LENGTH\_SHORT).show()**  **listOfEachQuestion[currentIndex] = true**  **} else{**  **this.checkAnswer[currentIndex] = true**  **Toast.makeText(this,"你答错了",Toast.LENGTH\_SHORT).show()**  **}**  **}else**  **Toast.makeText(this,"你已经知道答案了,作弊是不对的喔",Toast.LENGTH\_SHORT).show()**  **}** 4.1.3.2 查看答案按钮的点击事件 如果点击查看答案，会启动一个dialog对话框，让用户选择是否确定查看答案。  如果点击确定 那么就携带答案数据跳转到answerActivity    // 查看答案的按钮点击事件  answer.setOnClickListener {      //question.text = "answer"      var answerIntent = Intent(this, Answer::class.java)      AlertDialog.Builder(this).apply {// 弹出对话框 如果点击确认会启动 answerActivity          setTitle("查看答案")          setMessage("你确定要查看答案吗？")          setCancelable(false)          setPositiveButton("确定", DialogInterface.OnClickListener { dialogInterface, i ->              // 传递数据 拿到答案传送给  answerActivity              answerIntent.putExtra("answer",fileContentList[currentIndex].answer.toString())              startActivity(answerIntent)          })          setNegativeButton("取消", DialogInterface.OnClickListener { dialogInterface, i ->          })          show()      }      this.checkAnswer[currentIndex] = true  // 将查看过答案的标志标记为true  } 4.1.3.3下一题按钮的点击事件 在这里判断是否以及做完了所有的题目，如果已经做完了所有题目就准备启动finishActivity来进行 展示最后得分。否则当前索引值currentIndex++   // 下一个题目的点击事件   next.setOnClickListener {      // 如果当前索引已经超过列表中题目数量了 就准备启动finishActivity来进行 展示最后得分      if(currentIndex >= fileContentList.size -1){          val finishIndent = Intent(this,Finish::class.java)          var count = 0          for(ele in listOfEachQuestion){              if(ele)                  count++          }          val score = count \* 20          finishIndent.putExtra("score",score.toString())          startActivity(finishIndent)      }else{          currentIndex++          val strTemp:String = "${currentIndex +1}. "+ fileContentList[currentIndex].question          question.text = strTemp      }  } 4.2 AnswerActivity4.2.1 功能介绍 **最主要的功能就是展示从MainActivity中传输过来的数据，监听返回按钮的点击事件，返回继续做题。** 4.2.2 layout代码     <LinearLayout          android:layout\_width="match\_parent"          android:layout\_height="wrap\_content">          <Button              android:id="@+id/returnButton"              android:layout\_width="wrap\_content"              android:layout\_height="wrap\_content"              android:text="返回" />      </LinearLayout>      <LinearLayout          android:orientation="horizontal"          android:layout\_width="match\_parent"          android:layout\_height="wrap\_content"          android:gravity="center\_horizontal"          >          <TextView              android:textColor="@color/teal\_700"              android:textSize="20dp"              android:text="答案为："              android:layout\_width="wrap\_content"              android:layout\_height="wrap\_content"/>          <TextView              android:id="@+id/answerText"              android:textColor="#FF1100"              android:textSize="30dp"              android:text="答案在这"              android:layout\_width="wrap\_content"              android:layout\_height="wrap\_content"/>      </LinearLayout> 4.2.3 answerActivity代码 // 获取mainActivity中传过来的答案数据  var Temp:String? = intent.getStringExtra("answer")  answerText.text = Temp  // 展示答案  // 监听返回按钮的点击事件 如果点击了就结束当前activity  returnButton.setOnClickListener {      this.finish()  } 4.3 FinishActivity4.3.1 功能介绍 **最主要的功能就是展示从MainActivity中传输过来的数据，展示最后的得分** 4.3.2 layout代码 <LinearLayout  android:orientation="vertical"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_gravity="center\_vertical"  >      <TextView          android:textColor="@color/teal\_700"          android:textSize="20dp"          android:gravity="center\_horizontal"          android:text="您的最后得分是："          android:layout\_width="match\_parent"          android:layout\_height="wrap\_content"/>      <TextView          android:id="@+id/scoreText"          android:textColor="#FF1100"          android:textSize="30dp"          android:gravity="center\_horizontal"          android:text="最后得分"          android:layout\_width="match\_parent"          android:layout\_height="wrap\_content"/>  </LinearLayout> 4.2.3 answerActivity代码 val score = this.intent.getStringExtra("score") // 接受score数据  scoreText.text = score  // 展示出来 | | | | | |