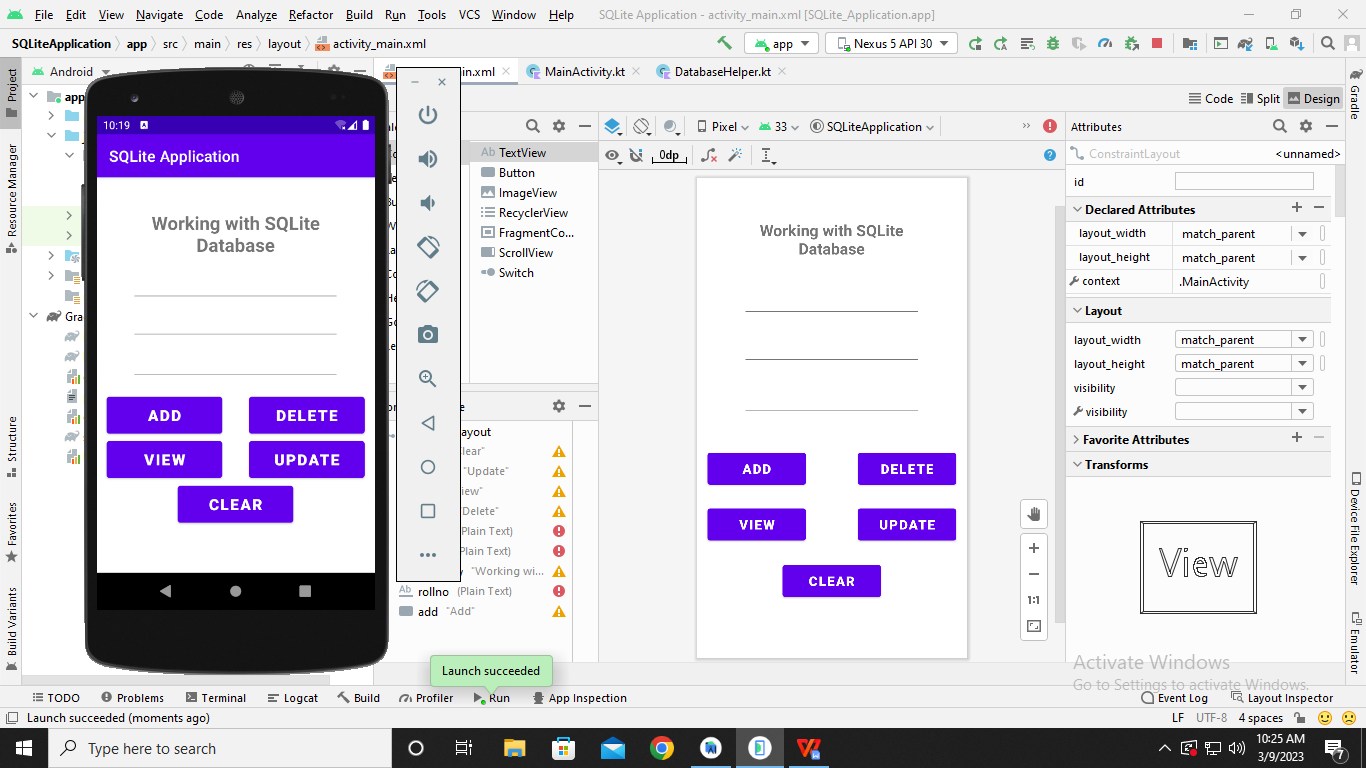
**Practical No. 13**

1. **Working with SQLite Database in Android**

**Designing UI:**



**Code:**

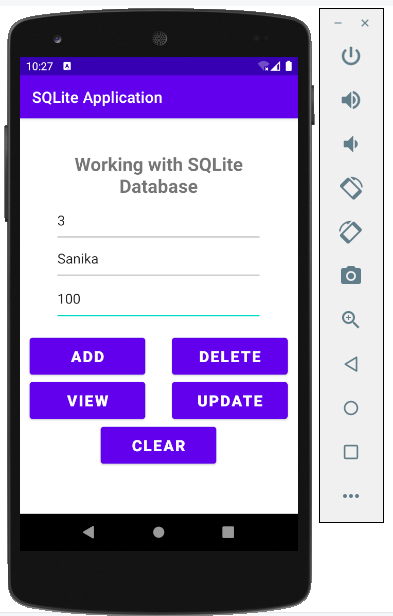
**MainActivity.kt**

**package** com.example.sqliteapplication  
  
**import** android.app.AlertDialog  
**import** androidx.appcompat.app.AppCompatActivity  
**import** android.os.Bundle  
**import** android.view.View  
**import** android.widget.Button  
**import** android.widget.EditText  
**import** android.widget.Toast  
  
**class** MainActivity() : AppCompatActivity() {  
 **var MyDb**: DatabaseHelper? = **null  
 var editrollno**: EditText? = **null  
 var editname**: EditText? = **null  
 var editmarks**: EditText? = **null  
 var button\_adddata**: Button? = **null  
 var button\_viewdata**: Button? = **null  
 var button\_delete**: Button? = **null  
 var button\_update**: Button? = **null  
 var button\_clear**: Button? = **null  
  
 override fun** onCreate(savedInstanceState: Bundle?) {  
 **super**.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_main*)  
  
 **MyDb** = DatabaseHelper(**this**)  
 **editrollno** = findViewById<View>(R.id.*rollno*) **as** EditText  
 **editname** = findViewById<View>(R.id.*name*) **as** EditText  
 **editmarks** = findViewById<View>(R.id.*marks*) **as** EditText  
 **button\_adddata** = findViewById<View>(R.id.*add*) **as** Button  
 **button\_viewdata** = findViewById<View>(R.id.*view*) **as** Button  
 **button\_update** = findViewById<View>(R.id.*update*) **as** Button  
 **button\_delete** = findViewById<View>(R.id.*delete*) **as** Button  
 **button\_clear** = findViewById<View>(R.id.*clear*) **as** Button  
  
 AddData()  
 ViewData()  
 UpdateData()  
 DeleteData()  
 ClearData()  
 }  
  
 **fun** AddData() {  
 **button\_adddata**?.setOnClickListener**{  
 var** result: Boolean? = **MyDb**?.insertdata(  
 **editrollno**?.*text*.*toString*(), **editname**?.*text*.*toString*(),  
 **editmarks**?.*text*.*toString*()  
 )  
 **if** (result==**true**) {  
 Toast.makeText(*applicationContext*, **"Data Inserted successfully"**, Toast.*LENGTH\_LONG*).show()  
 }  
 **else** {  
 Toast.makeText(*applicationContext*, **"Data Not Inserted"**, Toast.*LENGTH\_LONG*).show()  
 }  
 **}** }  
  
 **fun** ViewData() {  
 **button\_viewdata**!!.setOnClickListener(**object** : View.OnClickListener {  
 **override fun** onClick(view: View) {  
 **val** res = **MyDb**!!.getdata()  
 **if** (res.*count* == 0) {  
 *//show error message* showdata(**"Error"**, **"No data Found.."**)  
 }  
 **else** {  
 **val** buffer = StringBuffer()  
 **while** (res.moveToNext()) {  
 buffer.append(**"Rollno : "** + res.getString(0) + **"\n"**)  
 buffer.append(**"Name : "** + res.getString(1) + **"\n"**)  
 buffer.append(**"Marks : "** + res.getString(2) + **"\n\n"**)  
 }  
 *//showdata* showdata(**"Data"**, buffer.toString())  
 }  
 }  
 })  
 }  
  
 **fun** showdata(title: String?, message: String?) {  
 **val** builder = AlertDialog.Builder(**this**)  
 builder.setCancelable(**true**)  
 builder.setTitle(title)  
 builder.setMessage(message)  
 builder.show()  
 }  
  
 **fun** UpdateData() {  
 **button\_update**!!.setOnClickListener(**object** : View.OnClickListener {  
 **override fun** onClick(view: View) {  
 **val** result = **MyDb**!!.updatedata(  
 **editrollno**!!.*text*.toString(),  
 **editname**!!.*text*.toString(),  
 **editmarks**!!.*text*.toString()  
 )  
 **if** (result == **true**) {  
 Toast.makeText(*applicationContext*, **"Data Updated successfylly"**, Toast.*LENGTH\_LONG*).show()  
 }  
 **else** {  
 Toast.makeText(*applicationContext*, **"Data not Updated.."**, Toast.*LENGTH\_LONG*).show()  
 }  
 }  
 })  
 }  
  
 **fun** DeleteData() {  
 **button\_delete**!!.setOnClickListener(**object** : View.OnClickListener {  
 **override fun** onClick(view: View) {  
 **val** result = **MyDb**!!.deletedata(  
 **editrollno**!!.*text*.toString(),  
 **editname**!!.*text*.toString(),  
 **editmarks**!!.*text*.toString()  
 )  
 **if** (result == **true**) {  
 Toast.makeText(*applicationContext*, **"Data deleted successfylly"**, Toast.*LENGTH\_LONG*).show()  
 }  
 **else** {  
 Toast.makeText(*applicationContext*, **"Data not deleted..."**, Toast.*LENGTH\_LONG*).show()  
 }  
 }  
 })  
 }  
  
 **fun** ClearData() {  
 **button\_clear**!!.setOnClickListener(**object** : View.OnClickListener {  
 **override fun** onClick(v: View) {  
 **editrollno**!!.setText(**""**)  
 **editname**!!.setText(**""**)  
 **editmarks**!!.setText(**""**)  
 }  
 })  
 }  
}

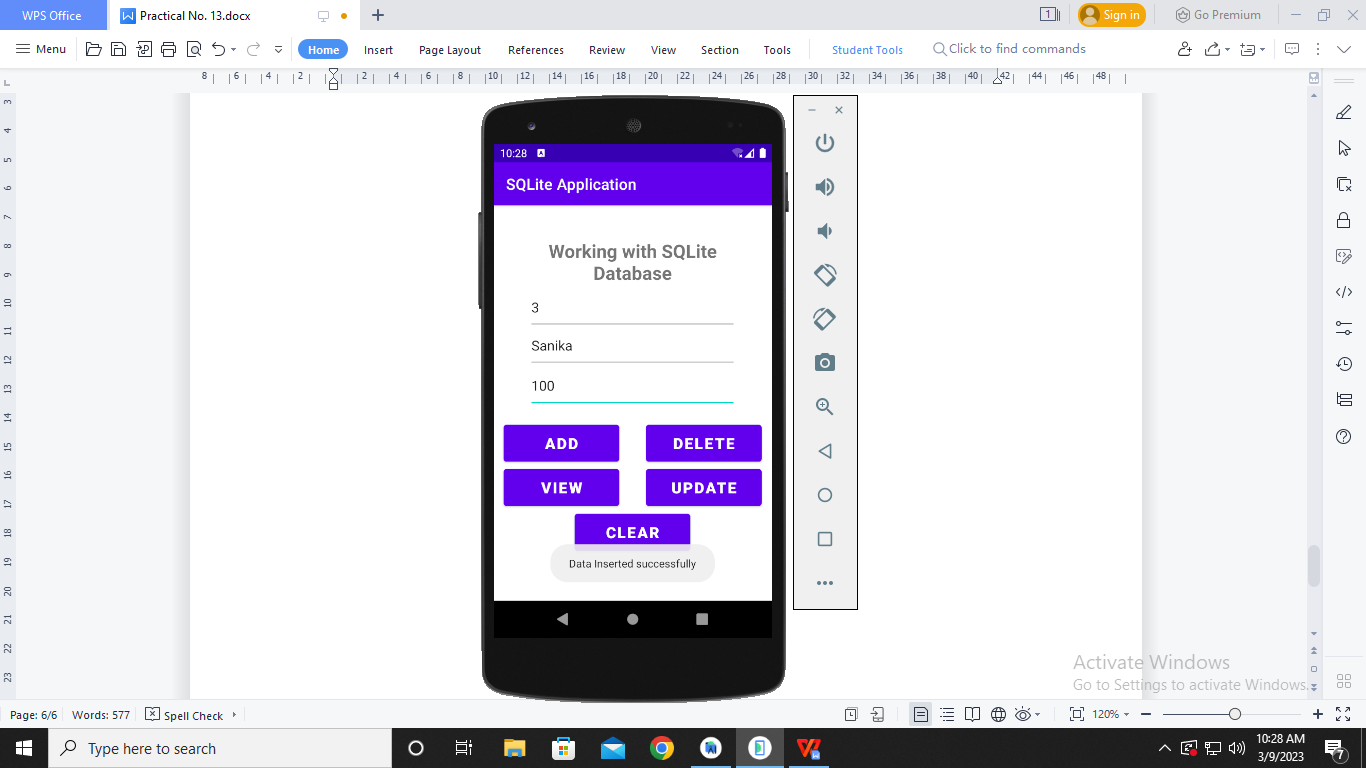
**DatabaseHelper.kt**

**package** com.example.sqliteapplication  
  
**import** android.database.sqlite.SQLiteOpenHelper;  
**import** android.database.sqlite.SQLiteDatabase;  
**import** android.content.ContentValues;  
**import** android.content.Context;  
**import** android.database.Cursor;  
  
**class** DatabaseHelper(context: Context?) : SQLiteOpenHelper(context, **DATABASE\_NAME**, **null**, 1) {  
 **override fun** onCreate(db: SQLiteDatabase) {  
 db.execSQL(**"create table "** + **TABLE\_NAME** + **"(Rollno integer,Name text,Marks text)"**)  
 }  
  
 **override fun** onUpgrade(db: SQLiteDatabase, i: Int, i1: Int) {  
 db.execSQL(**"drop table if exists "** + **TABLE\_NAME**)  
 onCreate(db)  
 }  
  
 **fun** insertdata(rollno: String?, name: String?, marks: String?): Boolean {  
 **val** db = *writableDatabase* **val** contentValues = ContentValues()  
 contentValues.put(**"Rollno"**, rollno)  
 contentValues.put(**"Name"**, name)  
 contentValues.put(**"Marks"**, marks)  
 **val** result = db.insert(**TABLE\_NAME**, **null**, contentValues)  
  
 *// db.execSQL("insert into student\_data values("+rollno+",'"+name+"',"+marks+")");* **return if** (result == -1L) {  
 **false**;  
 }  
 **else** {  
 **true** }  
 }  
  
 **fun** getdata(): Cursor {  
 **val** db = *writableDatabase* **return** db.rawQuery(**"select \*from $TABLE\_NAME"**, **null**)  
 }  
  
 **fun** updatedata(rollno: String, name: String?, marks: String?): Boolean {  
 **val** db = **this**.*writableDatabase* **val** contentValues = ContentValues()  
 contentValues.put(**"Rollno"**, rollno)  
 contentValues.put(**"Name"**, name)  
 contentValues.put(**"Marks"**, marks)  
 **val** result = db.update(  
 **"student\_data"**, contentValues,  
 **"Rollno=?"**, *arrayOf*(Integer.toString(rollno.*toInt*()))  
 )  
 **return if** (result != 0) {  
 **true** } **else** {  
 **false** }  
 }  
  
 **fun** deletedata(rollno: String, name: String?, marks: String?): Boolean {  
 **val** db = **this**.*writableDatabase* **val** result = db.delete(**"student\_data"**, **"Rollno=?"**, *arrayOf*(Integer.toString(rollno.*toInt*())))  
 **return if** (result != 0) {  
 **true** }  
 **else** {  
 **false** }  
 }  
  
 **companion object** {  
 **const val DATABASE\_NAME** = **"student.db"  
 const val TABLE\_NAME** = **"student\_data"  
 const val COL\_1** = **"Rollno"  
 const val COL\_2** = **"Name"  
 const val COL\_3** = **"Marks"** }  
}

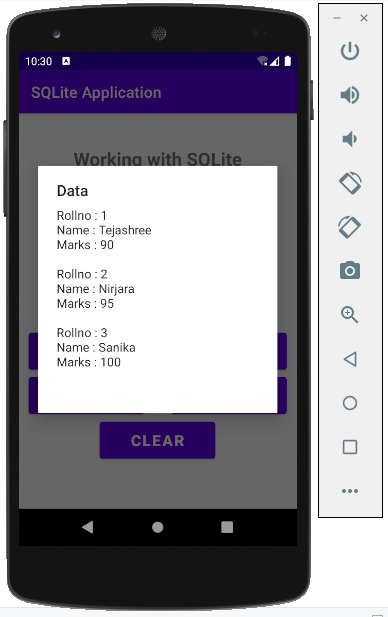
**Output:**



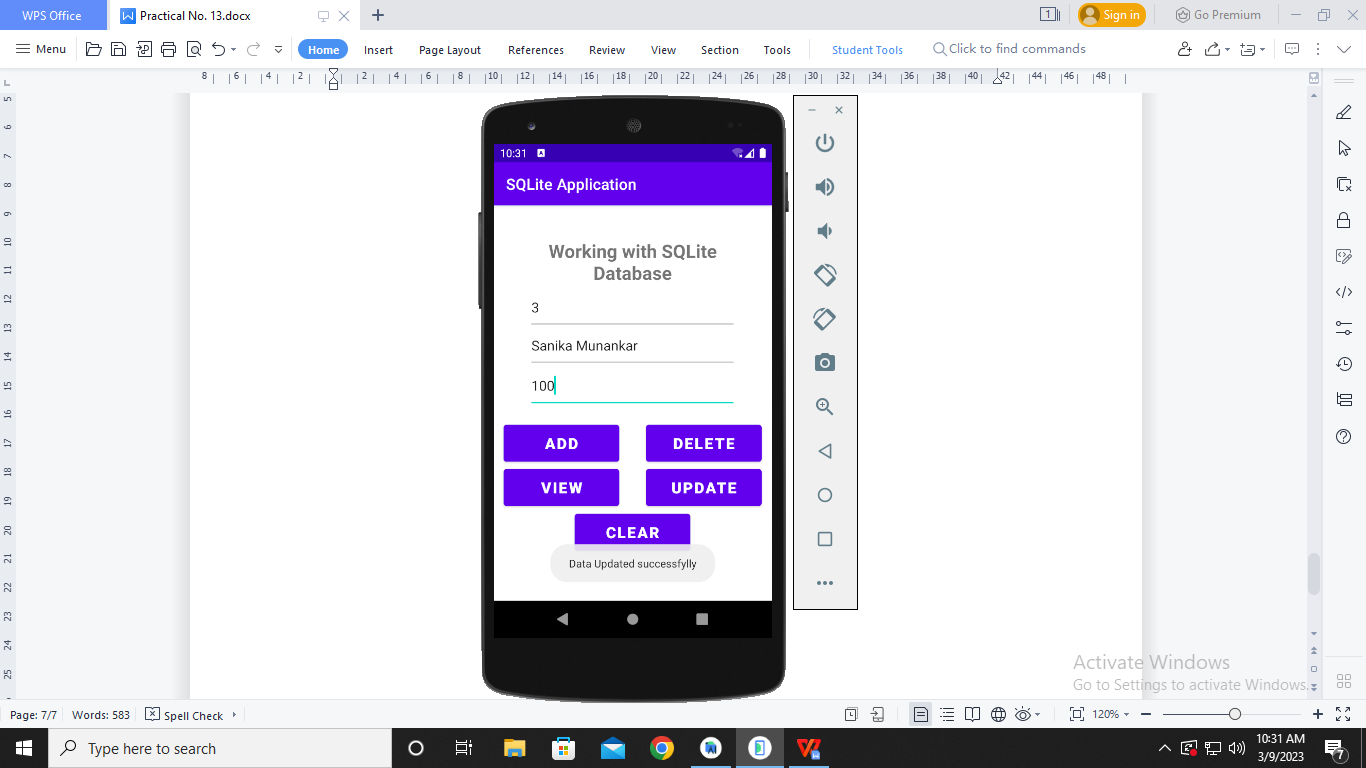
**After Clicking ‘ADD’**

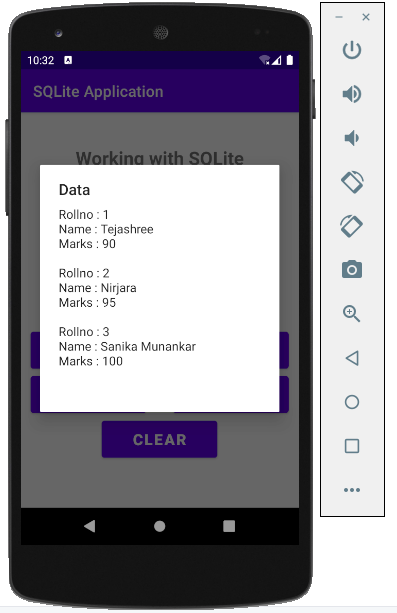


**After Clicking ‘VIEW’**

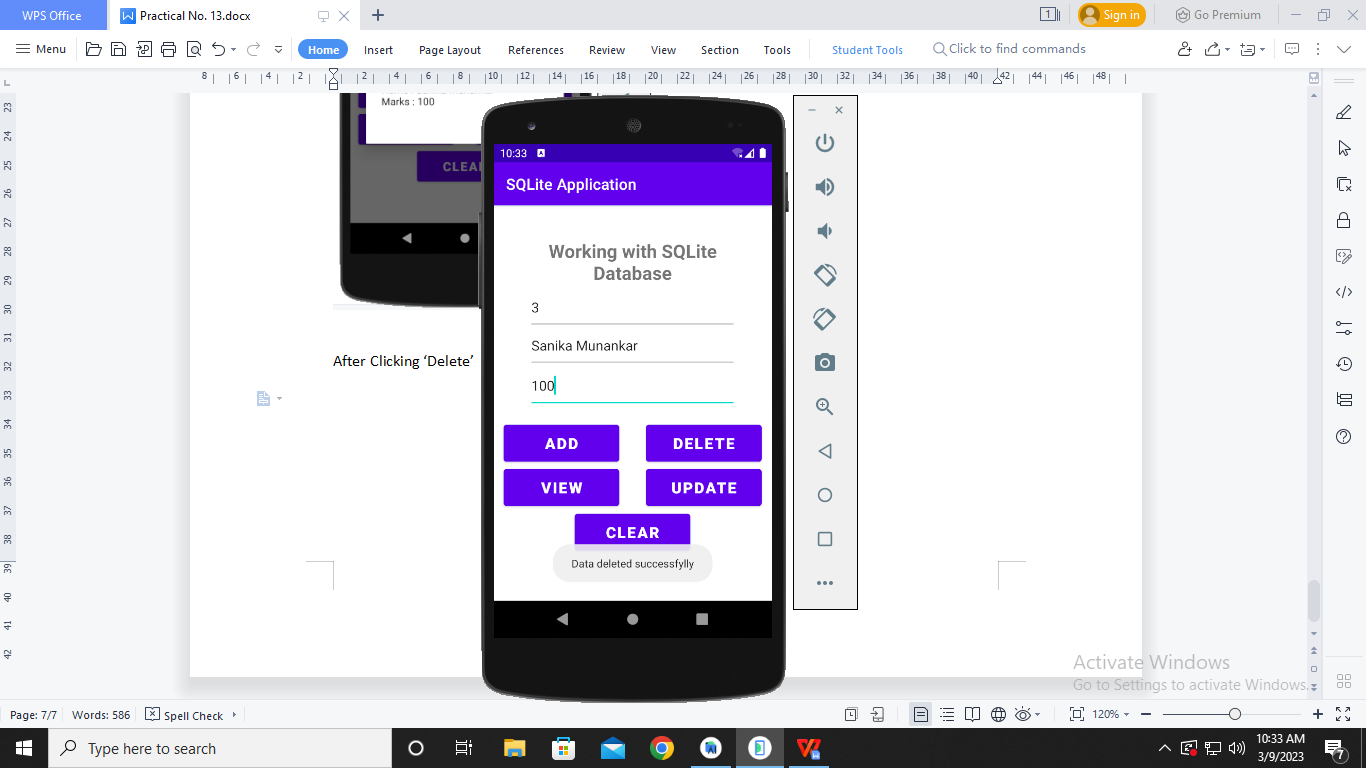


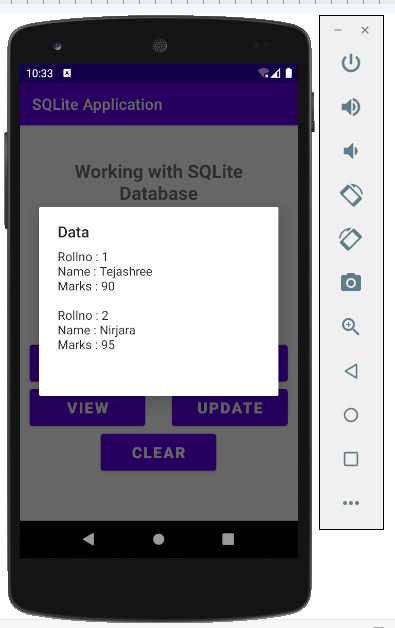
**After Clicking ‘UPDATE’**





**After Clicking ‘DELETE’**





**After Clicking ‘CLEAR’**

