**Problem sheet -11**

activity\_main.xml

**<?xml version="1.0" encoding="utf-8"?>**

**<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"**

**xmlns:app="http://schemas.android.com/apk/res-auto"**

**xmlns:tools="http://schemas.android.com/tools"**

**android:id="@+id/main"**

**android:layout\_width="match\_parent"**

**android:layout\_height="match\_parent"**

**tools:context=".MainActivity">**

**<Button**

**android:id="@+id/buttoninsert"**

**android:layout\_width="180dp"**

**android:layout\_height="70dp"**

**android:backgroundTint="#928787"**

**android:onClick="insert"**

**android:text="INSERT"**

**android:textColor="#070707"**

**android:textSize="24sp"**

**android:textStyle="bold"**

**app:cornerRadius="0dp"**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.0"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.405" />**

**<Button**

**android:id="@+id/buttonDelete"**

**android:layout\_width="180dp"**

**android:layout\_height="70dp"**

**android:backgroundTint="#928787"**

**android:onClick="delete"**

**android:text="DELETE"**

**android:textColor="#070707"**

**android:textSize="24sp"**

**android:textStyle="bold"**

**app:cornerRadius="0dp"**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="1.0"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.405" />**

**<Button**

**android:id="@+id/buttonUpdate"**

**android:layout\_width="155dp"**

**android:layout\_height="69dp"**

**android:backgroundTint="#928787"**

**android:onClick="update"**

**android:text="UPDATE"**

**android:textColor="#000000"**

**android:textSize="24sp"**

**android:textStyle="bold"**

**app:cornerRadius="0dp"**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.0"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.577" />**

**<Button**

**android:id="@+id/buttonView"**

**android:layout\_width="179dp"**

**android:layout\_height="81dp"**

**android:backgroundTint="#928787"**

**android:onClick="view"**

**android:text="VIEW"**

**android:textColor="#000000"**

**android:textSize="24sp"**

**android:textStyle="bold"**

**app:cornerRadius="0dp"**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="1.0"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.577" />**

**<Button**

**android:id="@+id/buttonViewAll"**

**android:layout\_width="183dp"**

**android:layout\_height="79dp"**

**android:backgroundTint="#928787"**

**android:onClick="viewall"**

**android:text="VIEW ALL"**

**android:textColor="#000000"**

**android:textSize="24sp"**

**android:textStyle="bold"**

**app:cornerRadius="0dp"**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.445"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.735" />**

**<TextView**

**android:id="@+id/textView2"**

**android:layout\_width="139dp"**

**android:layout\_height="27dp"**

**android:text="Enter Marks:"**

**android:textColor="#928787"**

**android:textSize="20sp"**

**android:textStyle="bold"**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.054"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.298" />**

**<TextView**

**android:id="@+id/textView3"**

**android:layout\_width="136dp"**

**android:layout\_height="25dp"**

**android:text="Enter Name:"**

**android:textColor="#928787"**

**android:textSize="20sp"**

**android:textStyle="bold"**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.052"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.207" />**

**<TextView**

**android:id="@+id/textView4"**

**android:layout\_width="134dp"**

**android:layout\_height="33dp"**

**android:text="Enter Rollno:"**

**android:textColor="#928787"**

**android:textSize="20sp"**

**android:textStyle="bold"**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.051"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.12" />**

**<EditText**

**android:id="@+id/Rollnumber"**

**android:layout\_width="188dp"**

**android:layout\_height="40dp"**

**android:ems="10"**

**android:hint="Rollno"**

**android:inputType="number|numberSigned"**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.735"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.121" />**

**<EditText**

**android:id="@+id/name"**

**android:layout\_width="188dp"**

**android:layout\_height="40dp"**

**android:ems="10"**

**android:hint="Name"**

**android:inputType="text"**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.735"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.211" />**

**<EditText**

**android:id="@+id/marks"**

**android:layout\_width="188dp"**

**android:layout\_height="40dp"**

**android:ems="10"**

**android:hint="Marks"**

**android:inputType="number|numberSigned"**

**app:layout\_constraintBottom\_toBottomOf="parent"**

**app:layout\_constraintEnd\_toEndOf="parent"**

**app:layout\_constraintHorizontal\_bias="0.735"**

**app:layout\_constraintStart\_toStartOf="parent"**

**app:layout\_constraintTop\_toTopOf="parent"**

**app:layout\_constraintVertical\_bias="0.285" />**

**</androidx.constraintlayout.widget.ConstraintLayout>**

MainActivity.kt

package com.example.ca\_2\_ps\_11

import android.app.AlertDialog

import android.os.Bundle

import android.widget.Button

import android.widget.EditText

import android.widget.Toast

import androidx.activity.enableEdgeToEdge

import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

*enableEdgeToEdge*()

setContentView(R.layout.*activity\_main*)

val dbHelper = DBHelper(this)

val rollNumber = findViewById<EditText>(R.id.*Rollnumber*)

val name = findViewById<EditText>(R.id.*name*)

val marks = findViewById<EditText>(R.id.*marks*)

val insertButton = findViewById<Button>(R.id.*buttoninsert*)

val updateButton = findViewById<Button>(R.id.*buttonUpdate*)

val deleteButton = findViewById<Button>(R.id.*buttonDelete*)

val viewButton = findViewById<Button>(R.id.*buttonView*)

val viewAllButton = findViewById<Button>(R.id.*buttonViewAll*)

insertButton.setOnClickListener **{**

val roll = rollNumber.*text*.toString().*trim*()

val studentName = name.*text*.toString().*trim*()

val studentMarks = marks.*text*.toString().*trim*()

if (roll.*isEmpty*() || studentName.*isEmpty*() || studentMarks.*isEmpty*()) {

showMessage("Error", "Please enter all values")

return@setOnClickListener

}

if (!roll.*matches*("\\d+".*toRegex*()) || !studentMarks.*matches*("\\d+".*toRegex*())) {

showMessage("Error", "Roll number and Marks must be numeric")

return@setOnClickListener

}

val inserted = dbHelper.insertData(roll.*toInt*(), studentName, studentMarks.*toInt*())

showMessage("Success", if (inserted) "Record Inserted" else "Insertion Failed")

**}**

updateButton.setOnClickListener **{**

val roll = rollNumber.*text*.toString().*trim*()

val studentName = name.*text*.toString().*trim*()

val studentMarks = marks.*text*.toString().*trim*()

if (roll.*isEmpty*() || studentName.*isEmpty*() || studentMarks.*isEmpty*()) {

showMessage("Error", "Please enter all values")

return@setOnClickListener

}

val updated = dbHelper.updateData(roll.*toInt*(), studentName, studentMarks.*toInt*())

showMessage("Success", if (updated) "Record Updated" else "Update Failed")

**}**

deleteButton.setOnClickListener **{**

val roll = rollNumber.*text*.toString().*trim*()

if (roll.*isEmpty*() || !roll.*matches*("\\d+".*toRegex*())) {

showMessage("Error", "Enter a valid Roll Number")

return@setOnClickListener

}

val deleted = dbHelper.deleteData(roll.*toInt*())

showMessage("Success", if (deleted) "Record Deleted" else "Deletion Failed")

**}**

viewButton.setOnClickListener **{**

val roll = rollNumber.*text*.toString().*trim*()

if (roll.*isEmpty*() || !roll.*matches*("\\d+".*toRegex*())) {

showMessage("Error", "Enter a valid Roll Number")

return@setOnClickListener

}

val student = dbHelper.getData(roll.*toInt*())

if (student != null) {

showMessage("Student Details", "Roll No: ${student.first}\nName: ${student.second}\nMarks: ${student.third}")

} else {

showMessage("Error", "No Data Found")

}

**}**

viewAllButton.setOnClickListener **{**

val allData = dbHelper.getAllData()

if (allData.isEmpty()) {

showMessage("Error", "No Data Found")

return@setOnClickListener

}

val buffer = StringBuilder()

for (student in allData) {

buffer.append("Roll No: ${student.first}\n")

buffer.append("Name: ${student.second}\n")

buffer.append("Marks: ${student.third}\n\n")

}

showMessage("Student Details", buffer.toString())

**}**

}

private fun showMessage(title: String, message: String) {

AlertDialog.Builder(this)

.setTitle(title)

.setMessage(message)

.setPositiveButton("OK") **{** dialog, \_ **->** dialog.dismiss() **}**

.show()

}

}

DBHelper.kt

package com.example.ca\_2\_ps\_11

import android.content.ContentValues

import android.content.Context

import android.database.sqlite.SQLiteDatabase

import android.database.sqlite.SQLiteOpenHelper

class DBHelper(context: Context) : SQLiteOpenHelper(context, "StudentDB.db", null, 1) {

override fun onCreate(db: SQLiteDatabase) {

db.execSQL("CREATE TABLE students (roll\_number INTEGER PRIMARY KEY, name TEXT, marks INTEGER)")

}

override fun onUpgrade(db: SQLiteDatabase, oldVersion: Int, newVersion: Int) {

db.execSQL("DROP TABLE IF EXISTS students")

onCreate(db)

}

fun insertData(rollNumber: Int, name: String, marks: Int): Boolean {

val db = *writableDatabase*

val values = ContentValues().*apply* **{**

put("roll\_number", rollNumber)

put("name", name)

put("marks", marks)

**}**

return db.insert("students", null, values) != -1L

}

fun updateData(rollNumber: Int, name: String, marks: Int): Boolean {

val db = *writableDatabase*

val values = ContentValues().*apply* **{**

put("name", name)

put("marks", marks)

**}**

return db.update("students", values, "roll\_number=?", *arrayOf*(rollNumber.toString())) > 0

}

fun deleteData(rollNumber: Int): Boolean {

val db = *writableDatabase*

return db.delete("students", "roll\_number=?", *arrayOf*(rollNumber.toString())) > 0

}

fun getData(rollNumber: Int): Triple<Int, String, Int>? {

val db = *readableDatabase*

val cursor = db.rawQuery("SELECT \* FROM students WHERE roll\_number=?", *arrayOf*(rollNumber.toString()))

return if (cursor.moveToFirst()) {

Triple(cursor.getInt(0), cursor.getString(1), cursor.getInt(2)).*also* **{** cursor.close() **}**

} else {

cursor.close()

null

}

}

fun getAllData(): List<Triple<Int, String, Int>> {

val db = *readableDatabase*

val cursor = db.rawQuery("SELECT \* FROM students", null)

return *mutableListOf*<Triple<Int, String, Int>>().*apply* **{**

while (cursor.moveToNext()) add(Triple(cursor.getInt(0), cursor.getString(1), cursor.getInt(2)))

cursor.close()

**}**

}

}