

­

19011519-015

Mobile Application Development

Assignment-2

Zunair Ahmad

CS-407

University of Gujrat

Faculty of Computing and information technology

Submitted to:

Dr. Muhammad Usman

Contents

[**1.** **Main Activity:** 3](#_Toc111382846)

[**2.** **Main Activity Xml:** 4](#_Toc111382847)

[**3.** **Login Activity:** 4](#_Toc111382848)

[**4.** **Login Activity Xml:** 5](#_Toc111382849)

[**5.** **DB Registration:** 6](#_Toc111382850)

[**6.** **Home Activity:** 7](#_Toc111382851)

[**7.** **Home Activity .xml:** 10](#_Toc111382852)

[**8.** **DB Form:** 11](#_Toc111382853)

[**9.** **Output of Login/Registration:** 14](#_Toc111382854)

[**10.** **Application Form Output:** 15](#_Toc111382855)

**ABSTRACT**

In this assignment task was assigned individually. As far my case I have to design an android application of **student account activation.** I used Android studio and implement the basic attribute of application form as input such as Name , Roll no. Semester, Degree and Fee details Etc. I designed the layout and implement the activity with database i have used **SQLite** . I performed the basic **CURD Operation** which is create update read and delete in the application form.It is easy to use and the interface is quite simple.

**Github Link:**

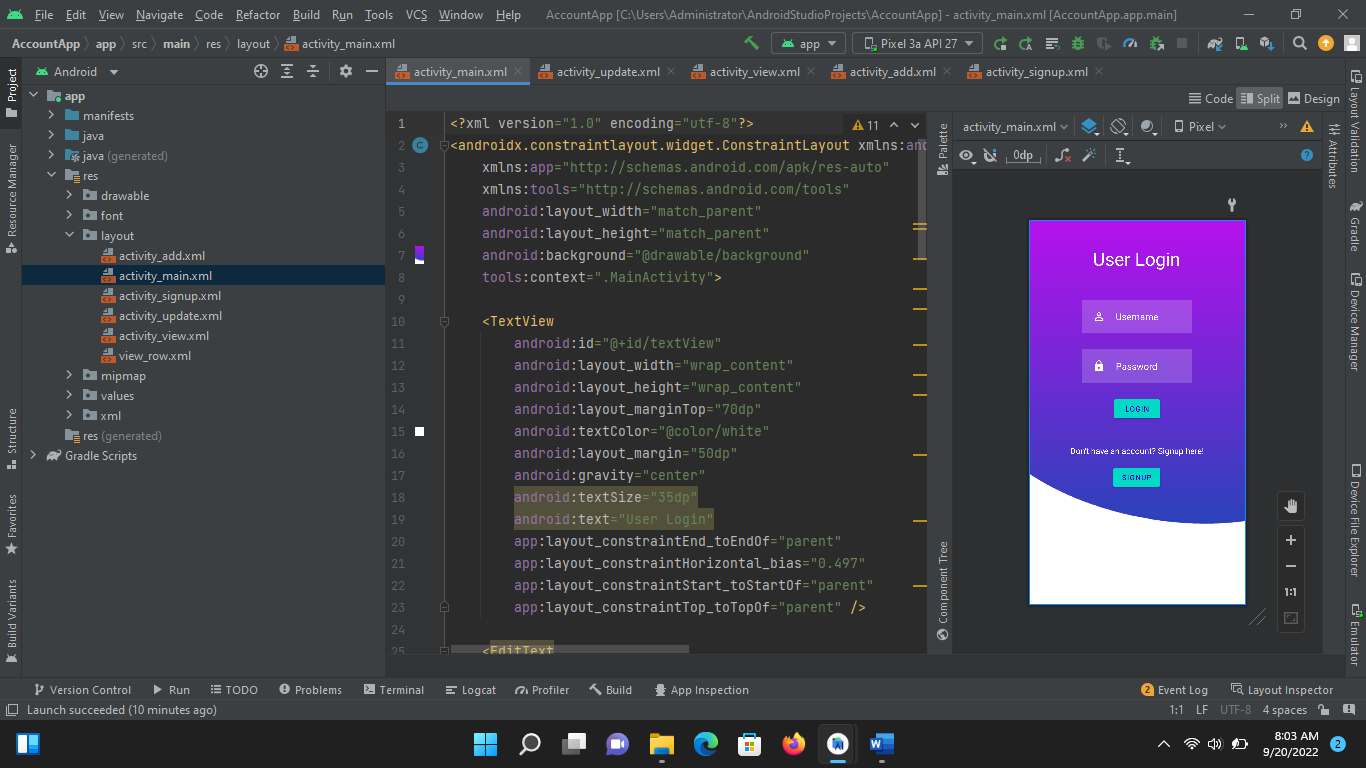
**https://github.com/ZunairAhmad/Student-Account-Activation-App**

**LMS Account Form App With SQLite Database**

# **Main Activity:**

public class MainActivity extends AppCompatActivity {  
  
 EditText username , password, repassword;  
 Button signup, signin;  
 DBClass DB;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 username = (EditText) findViewById(R.id.*username*);  
 password = (EditText) findViewById(R.id.*password*);  
 repassword = (EditText) findViewById(R.id.*repassword*);  
 signin = (Button) findViewById(R.id.*btnsigin*);  
 signup = (Button) findViewById(R.id.*btnsignup*);  
 DB = new DBClass(this);  
 signup.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String user = username.getText().toString();  
 String pass = password.getText().toString();  
 String repass = repassword.getText().toString();  
 if(user.equals("") || pass.equals("") || repass.equals("") ){  
 Toast.*makeText*(MainActivity.this,"Please enter all Fields",Toast.*LENGTH\_SHORT*).show();  
 }else{  
 if(pass.equals(repass)){  
 Boolean checkuser = DB.checkusername(user);  
 if (checkuser == false){  
 Boolean insert = DB.insertData(user,pass);  
 if (insert == true){  
 Toast.*makeText*(MainActivity.this,"Registered Successfully",Toast.*LENGTH\_SHORT*).show();  
 Intent intent = new Intent(getApplicationContext(),HomeActivity.class);  
 startActivity(intent);  
 }else{  
 Toast.*makeText*(MainActivity.this,"Registration Failed",Toast.*LENGTH\_SHORT*).show();  
 }  
  
 }else{  
 Toast.*makeText*(MainActivity.this,"User Already Exist Sign-in",Toast.*LENGTH\_SHORT*).show();  
 }else{  
 Toast.*makeText*(MainActivity.this,"Password Not Matched",Toast.*LENGTH\_SHORT*).show();  
 } }});  
 signin.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent intent = new Intent(getApplicationContext(),LoginActivity.class);  
 startActivity(intent);  
 }  
 });}}

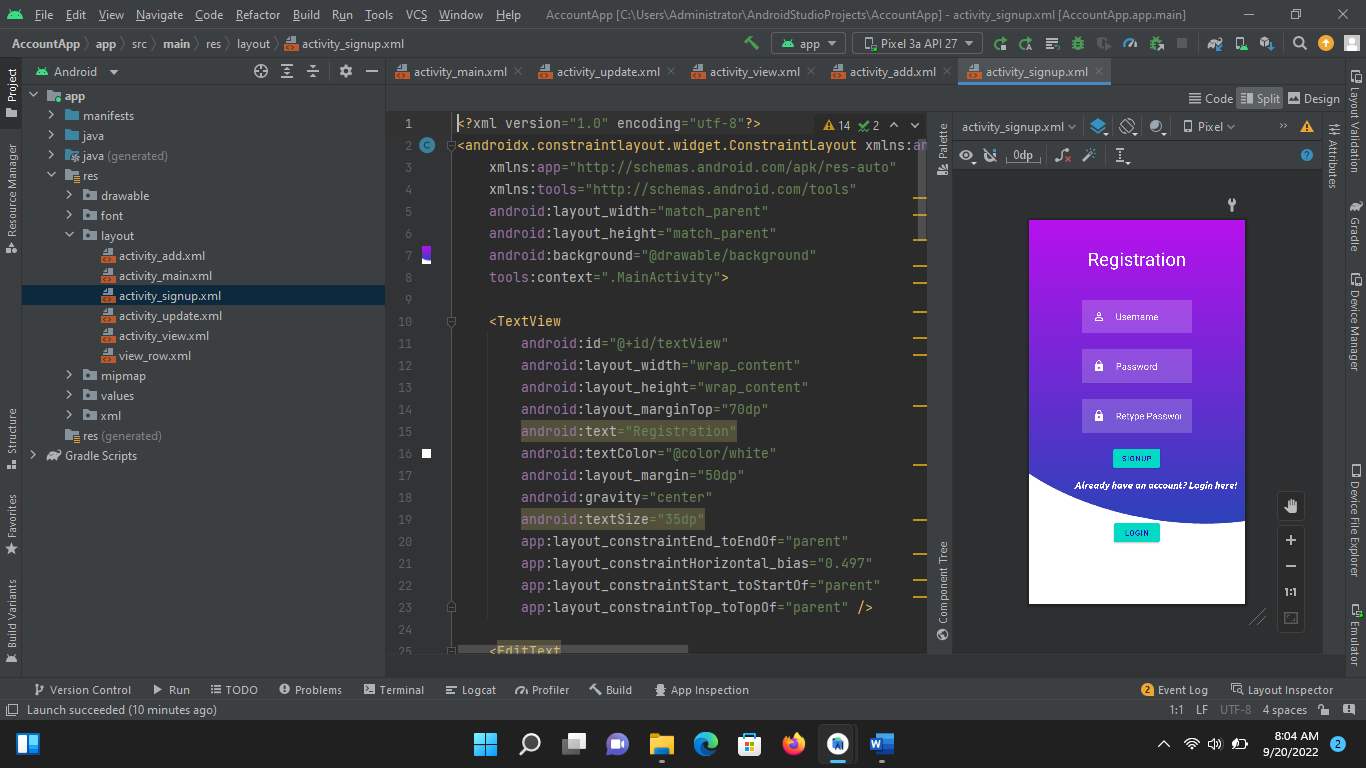
# **Main Activity Xml:**



# **Login Activity:**

public class LoginActivity extends AppCompatActivity {  
  
 EditText username, password;  
 Button btnlogin;  
 DBClass DB;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_login*);  
 username = (EditText) findViewById(R.id.*username1*);  
 password = (EditText) findViewById(R.id.*password1*);  
 btnlogin = (Button) findViewById(R.id.*btnsigin1*);  
 DB =new DBClass(this);  
  
 btnlogin.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String user = username.getText().toString();  
 String pass = password.getText().toString();  
 if (user.equals("") || pass.equals("")){  
 Toast.*makeText*(LoginActivity.this,"Please Enter all Fields",Toast.*LENGTH\_SHORT*).show();  
  
 }else{  
 Boolean checkuserpass = DB.checkusernamepassword(user,pass);  
 if (checkuserpass == true){  
 Toast.*makeText*(LoginActivity.this,"Sign in Successfully",Toast.*LENGTH\_SHORT*).show();  
 Intent intent = new Intent(getApplicationContext(),HomeActivity.class);  
 startActivity(intent);  
 }else{  
 Toast.*makeText*(LoginActivity.this,"Invalid Input",Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 }  
 });  
  
 }  
}

# **Login Activity Xml:**



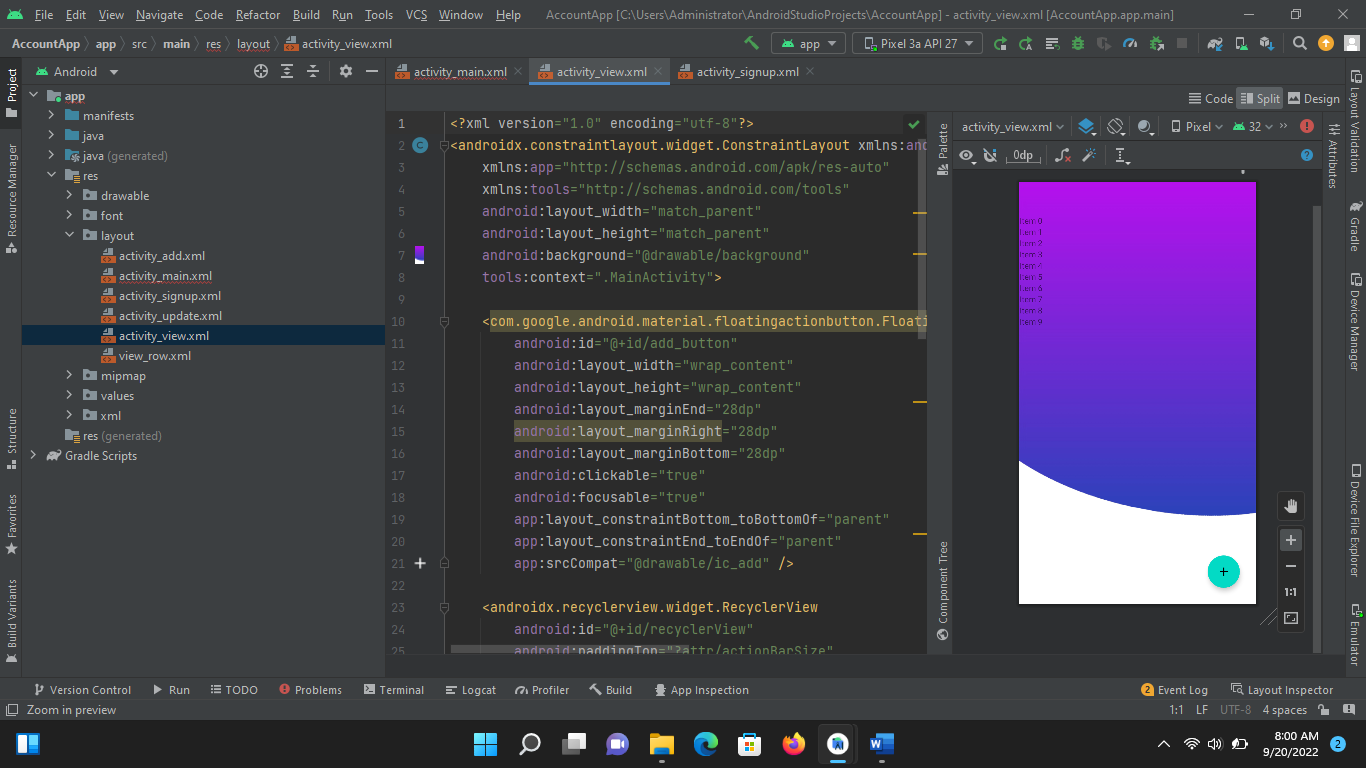
# **DB Registration:**

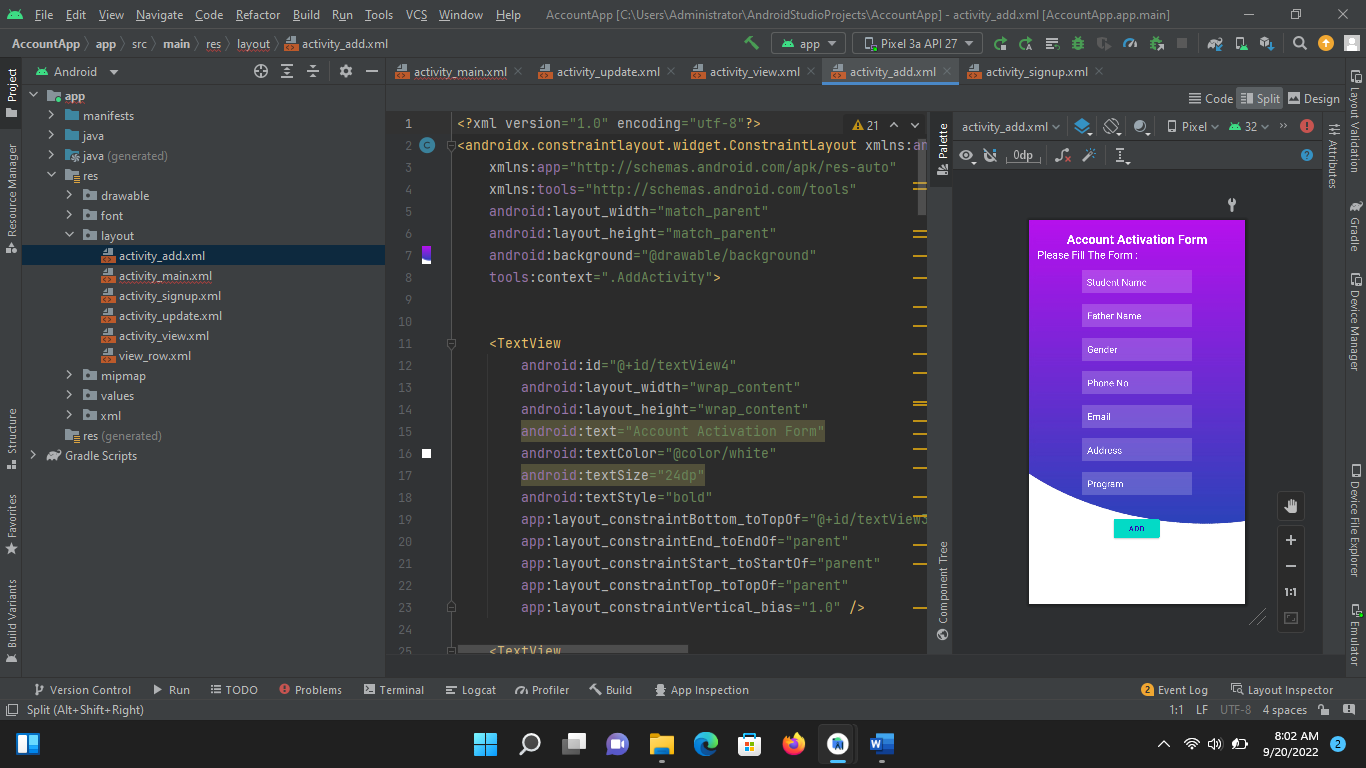
public class DBClass extends SQLiteOpenHelper {  
 public static final String *DBNAME* = "login.db";  
  
 public DBClass( Context context ) {  
 super(context, "login.db", null, 1);  
 }  
  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
 db.execSQL("create Table users(username TEXT primary key,password TEXT)");  
 //db.execSQL("create Table userdetails(name TEXT primary key,contact TEXT,dob TEXT)");  
  
 }  
  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
 db.execSQL("drop Table if exists users");  
 //db.execSQL("drop table if exists userdetails");  
 }  
 public Boolean insertData(String username, String password){  
 SQLiteDatabase db = this.getWritableDatabase();  
 ContentValues contentValues = new ContentValues();  
 contentValues.put("username",username);  
 contentValues.put("password",password);  
 long result = db.insert("users",null,contentValues);  
 if (result == -1) {  
 return false;  
 }  
 else {  
 return true;  
 }  
 }  
 public Boolean checkusername(String username){  
 SQLiteDatabase db = this.getWritableDatabase();  
 Cursor cursor = db.rawQuery("select \* from users where username = ?",new String[] {username});  
 if(cursor.getCount()>0){  
 return true;  
 }  
 else {  
 return false;  
 }  
  
  
 }  
 public Boolean checkusernamepassword(String username, String password){  
 SQLiteDatabase db = this.getWritableDatabase();  
 Cursor cursor = db.rawQuery("select \* from users where username = ? and password = ?",new String[] {username,password});  
 if (cursor.getCount()>0){  
 return true;  
 }  
 else{  
 return false;  
 }  
  
 }  
  
  
  
}

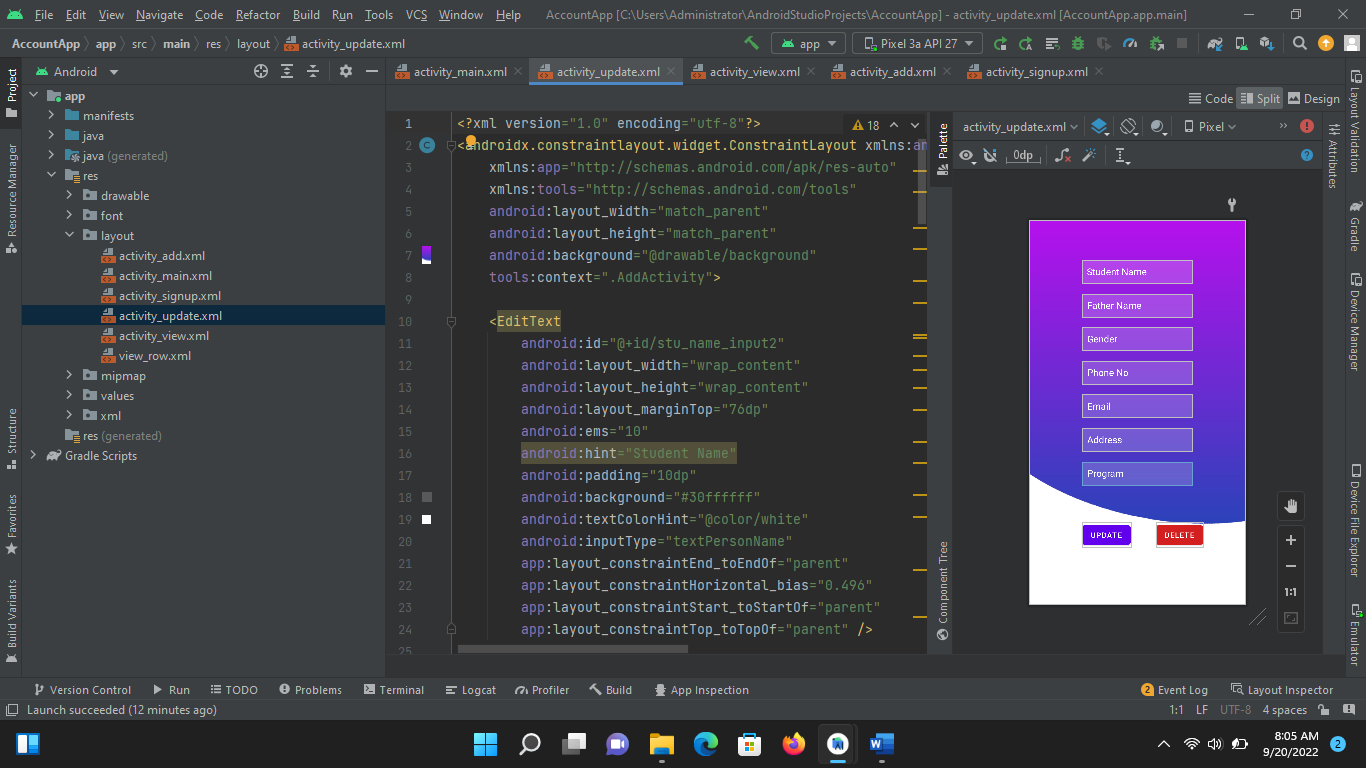
# **Home Activity:**

public class HomeActivity extends AppCompatActivity {  
  
 EditText name , roll, regNo,sem,program,degree,fee,depNo,subject;  
 Button btninsert, btnupdate,btndelete,btnview;  
 DBForm dbf;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_home*);  
  
 name = findViewById(R.id.*name*);  
 roll = findViewById(R.id.*roll*);  
 regNo = findViewById(R.id.*reg*);  
 sem = findViewById(R.id.*sem*);  
 program= findViewById(R.id.*program*);  
 degree = findViewById(R.id.*degree*);  
 fee = findViewById(R.id.*fee*);  
 depNo= findViewById(R.id.*depositNo*);  
 subject = findViewById(R.id.*subj*);  
  
 btninsert= findViewById(R.id.*btninsert*);  
 btndelete = findViewById(R.id.*btnDel*);  
 btnupdate = findViewById(R.id.*btnUpdate*);  
 btnview=findViewById(R.id.*btnView*);  
 dbf = new DBForm(this);  
  
 btninsert.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String nameTxt = name.getText().toString();  
 String rollTxt = roll.getText().toString();  
 String regNoTxt = regNo.getText().toString();  
 String semTxt = sem.getText().toString();  
 String prgmTxt = program.getText().toString();  
 String degTxt = degree.getText().toString();  
 String feeTxt= fee.getText().toString();  
 String depTxt = depNo.getText().toString();  
 String subjTxt= subject.getText().toString();  
  
 Boolean checkinsertdata = dbf.insertuserdata(nameTxt,rollTxt,regNoTxt,semTxt,prgmTxt,degTxt,feeTxt,depTxt,subjTxt) ;  
 if (checkinsertdata == true){  
 Toast.*makeText*(HomeActivity.this,"New Entry Inserted",Toast.*LENGTH\_SHORT*).show();  
 } else{  
 Toast.*makeText*(HomeActivity.this,"Entry Not Inserted",Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 });  
  
 btnupdate.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String nameTxt = name.getText().toString();  
 String rollTxt = roll.getText().toString();  
 String regNoTxt = regNo.getText().toString();  
 String semTxt = sem.getText().toString();  
 String prgmTxt = program.getText().toString();  
 String degTxt = degree.getText().toString();  
 String feeTxt= fee.getText().toString();  
 String depTxt = depNo.getText().toString();  
 String subjTxt= subject.getText().toString();  
 Boolean checkupdatedata = dbf.updateuserdata(nameTxt, rollTxt, regNoTxt,semTxt,prgmTxt,degTxt,feeTxt,depTxt,subjTxt);  
 if (checkupdatedata == true) {  
 Toast.*makeText*(HomeActivity.this, "Data Updated", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 Toast.*makeText*(HomeActivity.this, "Data not Updated", Toast.*LENGTH\_SHORT*).show();  
 }  
  
 }  
  
 });  
  
 btndelete.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String nameTxt = name.getText().toString();  
  
  
 Boolean checkdeletedata = dbf.deleteuserdata(nameTxt);  
 if (checkdeletedata == true) {  
 Toast.*makeText*(HomeActivity.this, "Data Deleted", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 Toast.*makeText*(HomeActivity.this, "Data not Deleted", Toast.*LENGTH\_SHORT*).show();  
 }  
  
  
  
 }  
 });  
  
 btnview.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Cursor res = dbf.getdata();  
 if (res.getCount() == 0) {  
 Toast.*makeText*(HomeActivity.this,"No Data Exists",Toast.*LENGTH\_SHORT*).show();  
 return;  
 }  
 StringBuffer buffer = new StringBuffer();  
 while(res.moveToNext()){  
 buffer.append("Name :"+res.getString(0)+"\n");  
 buffer.append("Roll No :"+res.getString(1)+"\n");  
 buffer.append("Subject :"+res.getString(8)+"\n");  
 /\* buffer.append("Sem :"+res.getString(3)+"\n");  
 buffer.append("Program :"+res.getString(4));  
 buffer.append("Degree :"+res.getString(5)+"\n");  
 buffer.append("Fee :"+res.getString(6));  
 buffer.append("Deposit No :"+res.getString(7)+"\n");  
 buffer.append("Subject:"+res.getString(8)+"\n");\*/  
 }  
 AlertDialog.Builder builder = new AlertDialog.Builder(HomeActivity.this);  
 builder.setCancelable(true);  
 builder.setTitle("User Entries:") ;  
 builder.setMessage(buffer.toString());  
 builder.show();  
 }  
 });  
 }  
}

# **Home Activity .xml:**



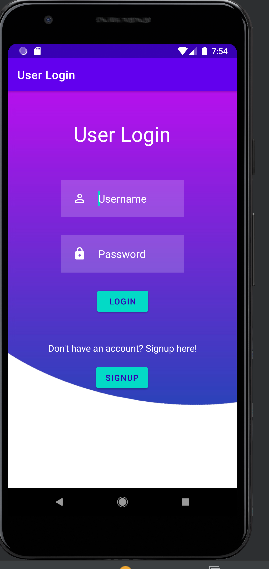
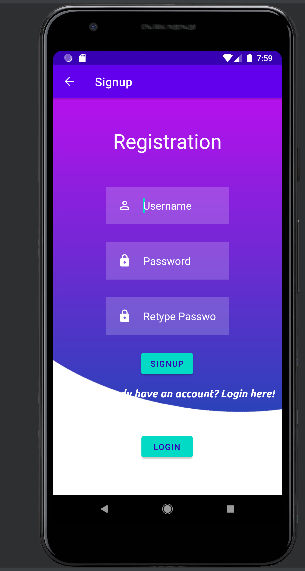


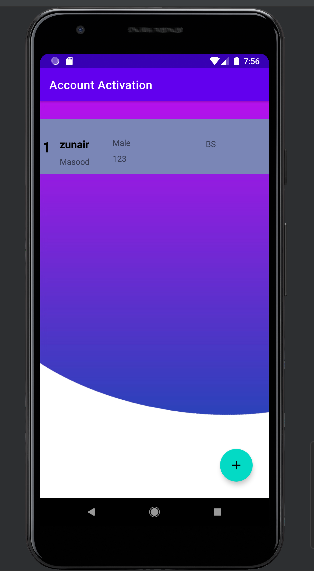
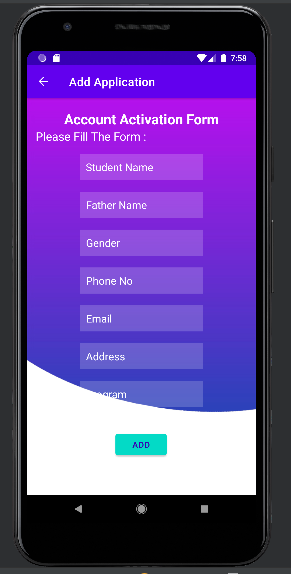


# **DB Form:**

public class DBForm extends SQLiteOpenHelper {  
 public DBForm(Context context) {  
 super(context, "data.db", null, 1);  
 }  
  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
 db.execSQL("create Table userDetails(name TEXT primary key,rollNo TEXT,regNo TEXT,sem TEXT,program TEXT,degree TEXT,fee TEXT,depNo TEXT,subject TEXT)");  
  
 }  
  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
 db.execSQL("drop table if exists userDetails");  
  
 }  
 public Boolean insertuserdata(String name,String rollNo,String regNo,String sem,String program,  
 String degree,String fee,String depNo,String subject){  
 SQLiteDatabase db = this.getWritableDatabase();  
 ContentValues contentValues = new ContentValues();  
 contentValues.put("name",name);  
 contentValues.put("rollNo",rollNo);  
 contentValues.put("regNo",regNo);  
 contentValues.put("sem",sem);  
 contentValues.put("program",program);  
 contentValues.put("degree",degree);  
 contentValues.put("fee",fee);  
 contentValues.put("depNo",depNo);  
 contentValues.put("subject",subject);  
   
  
 long result = db.insert("userDetails",null,contentValues);  
 if(result == -1){  
 return false;  
 }else{  
 return true;  
 }  
  
 }  
 public Boolean updateuserdata(String name,String rollNo,String regNo,String sem,String program,  
 String degree,String fee,String depNo,String subject){  
 SQLiteDatabase db = this.getWritableDatabase();  
 ContentValues contentValues = new ContentValues();  
 contentValues.put("rollNo",rollNo);  
 contentValues.put("regNo",regNo);  
 contentValues.put("sem",sem);  
 contentValues.put("program",program);  
 contentValues.put("degree",degree);  
 contentValues.put("fee",fee);  
 contentValues.put("depNo",depNo);  
 contentValues.put("subject",subject);  
  
 Cursor cursor = db.rawQuery("select \* from userDetails where name =?",new String[] {name} );  
 if(cursor.getCount()>0){  
 long result = db.update("userDetails",contentValues,"name=?",new String[] {name});  
 if(result == -1){  
 return false;  
 }else{  
 return true;  
 }  
 }else{  
 return false;  
 }  
  
 }  
 public Boolean deleteuserdata(String name){  
 SQLiteDatabase db = this.getWritableDatabase();  
  
 Cursor cursor = db.rawQuery("select \* from userDetails where name =?",new String[] {name} );  
 if(cursor.getCount()>0){  
 long result = db.delete("userDetails","name=?",new String[] {name});  
 if(result == -1){  
 return false;  
 }else{  
 return true;  
 }  
 }else{  
 return false;  
 }  
 }  
  
 public Cursor getdata(){  
 SQLiteDatabase db = this.getWritableDatabase();  
  
 Cursor cursor = db.rawQuery("select \* from userDetails",null );  
 return cursor;  
 }  
  
}

1. **Output of Login/Registration:**

****

1. **Application Form Output:**