

MOBILE APPLICATION DEVELOPMENT LAB
CYCLE - 5

1. Create a registration and login page using SQLite database

PROGRAM

(REGISTRATION.xml)

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:minWidth="25px"
    android:minHeight="25px">

    <EditText
        android:hint="Username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:id="@+id/edtusername"/>

    <EditText
        android:hint="Email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:id="@+id/edtEmail"
    />

    <EditText
        android:hint="Password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:id="@+id/edtpassword"
        android:inputType="textPassword" />

    <EditText
        android:hint="First Name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:id="@+id/edtfname"/>

    <EditText
        android:hint="Last Name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
```

```

        android:id="@+id/edtlname"/>
    <EditText
        android:hint="Mobile"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:id="@+id/edtMobile"
        android:inputType="phone" />
    <EditText
        android:hint="Address"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:id="@+id/edtAddress"/>
    <EditText
        android:hint="Country"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:id="@+id/edtCountry"/>
    <Button
        android:text="Create"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:id="@+id/btnCreate" />
    <Button
        android:text="Back to Home"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:id="@+id/btnBack" />
</LinearLayout>

```

Login (Main.xml)

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:minWidth="25px"

```

```

        android:minHeight="25px">
        <EditText
            android:hint="Username"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="20dp"
            android:layout_marginRight="20dp"
            android:id="@+id/txtusername"/>
        <EditText
            android:hint="Password"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="20dp"
            android:layout_marginRight="20dp"
            android:id="@+id/txtpassword"
            android:inputType="textPassword" />
        <Button
            android:text="Sign In"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="20dp"
            android:layout_marginRight="20dp"
            android:id="@+id/btnSign" />
        <TextView
            android:text="Or"
            android:textAppearance="?android:attr/textAppearanceMedium"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textColor="#000"
            android:gravity="center"
            android:id="@+id/txtOr" />
        <Button
            android:text="Register"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="20dp"
            android:layout_marginRight="20dp"
            android:id="@+id/btnSignUp" />
    </LinearLayout>

```

WELCOME.XML

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:minWidth="25px"
    android:minHeight="25px">

```

```

<TextView
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textColor="#000"
    android:gravity="left"
    android:id="@+id/txtWelcome" />
<Button
    android:text="Logout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:id="@+id/btnLogout" />
</LinearLayout>

```

UserDetails.cs (DTO)

```

public class UserDetails
{
    public string ID { get; set; }
    public string Username { get; set; }
    public string FirstName { get; set; }
    public string LastName { get; set; }
    public string Address { get; set; }
    public string Country { get; set; }
    public string Email { get; set; }
    public string Password { get; set; }
    public string Mobile { get; set; }
    public UserDetails() { }
    public UserDetails(string Id, string username, string fname, string lname, string address, string country, string email, string password, string mobile) //Constructor with all parameters
    {
        ID = Id;
        Username = username;
        FirstName = fname;
        LastName = lname;
        Address = address;
        Country = country;
        Email = email;
        Password = password;
        Mobile = mobile;
    }
    public UserDetails(string Password) //Constructor with one parameter
    {
        this.Password = Password;
    }
}

```

Helper.cs (Database Helper - SQLite Operations)

```
public class Helper : SQLiteOpenHelper
{
    private static string _DatabaseName = "clientDatabase";

    public Helper(Context context) : base(context, _DatabaseName, null, 1) { }

    public override void OnCreate(SQLiteDatabase db)
    {
        db.ExecSQL(Helper.CreateQuery);
    }

    public override void OnUpgrade(SQLiteDatabase db, int oldVersion, int newVersion)
    {
        db.ExecSQL(Helper.DeleteQuery);
        OnCreate(db);
    }

    private const string TableName = "adminTable";
    private const string ColumnID = "id";
    private const string ColumnUsername = "username";
    private const string ColumnFirstName = "firstname";
    private const string ColumnLastName = "lastname";
    private const string ColumnAddress = "address";
    private const string ColumnCountry = "country";
    private const string ColumnPassword = "password";
    private const string ColumnEmail = "email";
    private const string ColumnMobile = "mobile";

    public const string CreateQuery = "CREATE TABLE " + TableName +
        " ( "
        + ColumnID + " INTEGER PRIMARY KEY,"
        + ColumnUsername + " TEXT,"
        + ColumnFirstName + " TEXT,"
        + ColumnLastName + " TEXT,"
        + ColumnAddress + " TEXT,"
        + ColumnCountry + " TEXT,"
        + ColumnPassword + " TEXT,"
        + ColumnEmail + " TEXT,"
        + ColumnMobile + " TEXT)";

    public const string DeleteQuery = "DROP TABLE IF EXISTS " + TableName;

    public void Register(Context context, UserDetails user)
    {
        SQLiteDatabase db = new Helper(context).WritableDatabase;
        ContentValues Values = new ContentValues();
```

```

        Values.Put(ColumnUsername, user.Username);
        Values.Put(ColumnFirstName, user.FirstName);
        Values.Put(ColumnLastName, user.LastName);
        Values.Put(ColumnCountry, user.Country);
        Values.Put(ColumnAddress, user.Address);
        Values.Put(ColumnPassword, user.Password);
        Values.Put(ColumnEmail, user.Email);
        Values.Put(ColumnMobile, user.Mobile);
        db.Insert(TableName, null, Values);
        db.Close();
    }

    public UserDetails Authenticate(Context context, UserDetails user)
    {
        SQLiteDatabase db = new Helper(context).ReadableDatabase;
        ICursor cursor = db.Query(TableName, new string[]
        { ColumnID, ColumnFirstName, ColumnLastName, ColumnAddress, ColumnCountry, ColumnUsername, ColumnPassword, ColumnEmail, ColumnMobile },
        ColumnUsername + "=?", new string[] { user.Username }, null, null, null)
        ;
        if(cursor != null && cursor.moveToFirst() && cursor.Count > 0)
        {
            UserDetails user1 = new UserDetails(cursor.GetString(6));
            if (user.Password.Equals(user1.Password))
                return user1;
        }
        return null;
    }

    public List<UserDetails> GetUser(Context context)
    {
        List<UserDetails> users = new List<UserDetails>();
        SQLiteDatabase db = new Helper(context).ReadableDatabase;
        string[] columns = new string[] { ColumnID, ColumnUsername, ColumnFirstName, ColumnLastName, ColumnAddress, ColumnCountry, ColumnPassword, ColumnEmail, ColumnMobile };
        using(ICursor cursor = db.Query(TableName, columns, null, null, null, null, null))
        {
            while (cursor.MoveNext())
            {
                users.Add(new UserDetails
                {
                    ID = cursor.GetString(cursor.GetColumnIndexOrThrow(ColumnID)),
                    Username = cursor.GetString(cursor.GetColumnIndexOrThrow(ColumnUsername)),
                    FirstName = cursor.GetString(cursor.GetColumnIndexOrThrow(ColumnFirstName)),

```

```

        Country = cursor.GetString(cursor.GetColumnIndexOrThrow(Col
mnCountry)),
        Address = cursor.GetString(cursor.GetColumnIndexOrThrow(Colu
mnAddress)),
        LastName = cursor.GetString(cursor.GetColumnIndexOrThrow(Col
umnLastName)),
        Password = cursor.GetString(cursor.GetColumnIndexOrThrow(Col
umnPassword)),
        Email = cursor.GetString(cursor.GetColumnIndexOrThrow(Column
Email)),
        Mobile = cursor.GetString(cursor.GetColumnIndexOrThrow(Colum
nMobile))

    });
}
db.Close();
return users;
}
}

```

ACTIVITY

MainActivity.cs

```

[Activity(MainLauncher = true)]
public class MainActivity : Activity
{
    private EditText txtUsername, txtPassword;
    private Button btnSignIn, btnCreate;
    Helper helper;
    protected override void OnCreate(Bundle savedInstanceState)
    {
        base.OnCreate(savedInstanceState);

        // Set our view from the "main" layout resource
        SetContentView(Resource.Layout.Main);

        txtUsername = FindViewById<EditText>(Resource.Id.txtusername);
        txtPassword = FindViewById<EditText>(Resource.Id.txtpassword);
        btnCreate = FindViewById<Button>(Resource.Id.btnSignUp);
        btnSignIn = FindViewById<Button>(Resource.Id.btnSignIn);
        helper = new Helper(this);

        btnCreate.Click += delegate { StartActivity(typeof(SignUp)); };

        btnSignIn.Click += delegate
        {
            try
            {

```



```

        string Username = txtUsername.Text.ToString();
        string Password = txtPassword.Text.ToString();
        var user = helper.Authenticate(this,new UserDetails(null,Username,null,null,null,null,null,null,null,null,null,null,null),Password,null));
        if (user != null)
        {
            Toast.MakeText(this, "Login Successful", ToastLength.Short).Show();

            Intent intent = new Intent(this, typeof(Welcome));
            intent.PutExtra("UserName", Username);
            StartActivity(typeof(Welcome));
        }
        else
        {
            Toast.MakeText(this, "Login Unsuccessful! Please verify your Username and Password", ToastLength.Short).Show();
        }
    }
    catch (SQLiteException ex)
    {
        Toast.MakeText(this, ""+ex, ToastLength.Short).Show();
    }
};
}
}

```

REGISTRATION.CS

```

public class SignUp : Activity
{
    private EditText edtfname,edtlname,edtCountry,edtAddress, edtUsername, edtEmail, edtPassword, edtMobile;
    private Button btnCreate, btnBack;
    Helper helper;
    protected override void OnCreate(Bundle savedInstanceState)
    {
        base.OnCreate(savedInstanceState);

        // Create your application here
        SetContentView(Resource.Layout.SignUp);

        edtfname = FindViewById<EditText>(Resource.Id.edtfname);
        edtlname= FindViewById<EditText>(Resource.Id.edtlname);
        edtCountry= FindViewById<EditText>(Resource.Id.edtCountry);
        edtAddress= FindViewById<EditText>(Resource.Id.edtAddress);
        edtUsername = FindViewById<EditText>(Resource.Id.edtusername);
    }
}

```

```

edtPassword = FindViewById<EditText>(Resource.Id.edtpassword);
edtEmail = FindViewById<EditText>(Resource.Id.edtEmail);
edtMobile = FindViewById<EditText>(Resource.Id.edtMobile);
btnCreate = FindViewById<Button>(Resource.Id.btnCreate);
btnBack = FindViewById<Button>(Resource.Id.btnBack);
helper = new Helper(this);

btnBack.Click += delegate { StartActivity(typeof(MainActivity)); };

btnCreate.Click += delegate
{
    UserDetails user = new UserDetails()
    {
        FirstName = edtfname.Text,
        LastName=edtlname.Text,
        Address=edtAddress.Text,
        Country=edtCountry.Text,
        Username = edtUsername.Text,
        Password = edtPassword.Text,
        Email = edtEmail.Text,
        Mobile = edtMobile.Text
    };
    string username = edtUsername.Text;
    string password = edtPassword.Text;
    if(string.IsNullOrEmpty(username) || string.IsNullOrEmpty(password))

    {
        Toast.MakeText(this, "Username and Password should not be empty."
, ToastLength.Short).Show();
    }
    else
    {
        helper.Register(this, user);
        var data = helper.GetUser(this);
        user = data[data.Count - 1];
        Toast.MakeText(this, $"User {user.Username} registration successful
!", ToastLength.Short).Show();
        Clear();
        Toast.MakeText(this, $"Total {data.Count} Users.", ToastLength.Sho
rt).Show();
    }
};

}
void Clear()
{
    edtfname.Text = "";
    edtlname.Text = "";
    edtAddress.Text = "";
    edtCountry.Text = "";

```

```
        edtUsername.Text = "";
        edtPassword.Text = "";
        edtMobile.Text = "";
        edtEmail.Text = "";
    }
}
```

WELCOME.CS

```
[Activity(Label = "Welcome")]
public class Welcome : Activity
{
    string loggedInUser { get; set; }

    private TextView txtWelcome;
    private Button btnLogout;
    protected override void OnCreate(Bundle savedInstanceState)
    {
        base.OnCreate(savedInstanceState);

        // Create your application here
        SetContentView(Resource.Layout.Welcome);

        txtWelcome = FindViewById<TextView>(Resource.Id.txtWelcome);
        txtWelcome.Text = $"Welcome {Intent.GetStringExtra("UserName") ?? "User" }";
        btnLogout = FindViewById<Button>(Resource.Id.btnLogout);

        btnLogout.Click += delegate { StartActivity(typeof(MainActivity)); };
    }
}
```