B

1. **Create an android application to implement the concept of Notification builder class**

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Notification builder"
        android:layout_marginTop="20dp"
        android:gravity="center"
        android:textSize="25dp"

        />

    <Button
        android:id="@+id/btnShowNotification"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="60dp"
        android:text="Show Notification" />


</LinearLayout>
```

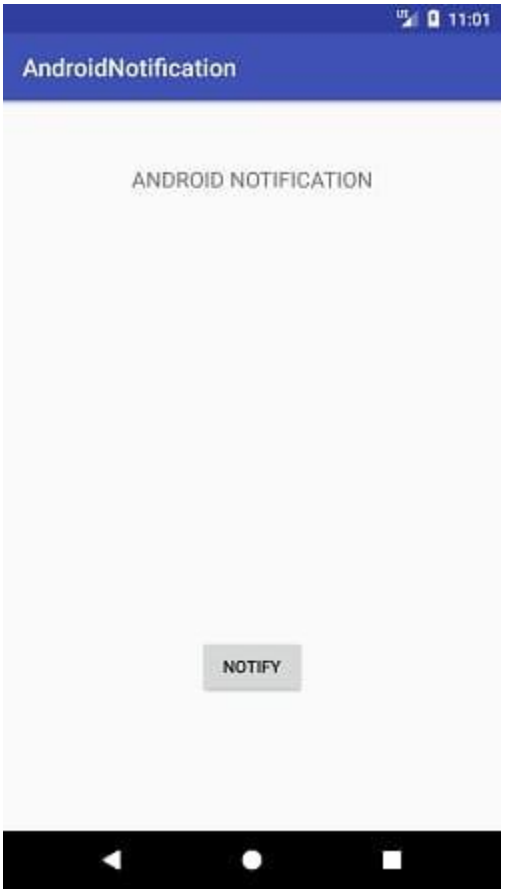
Java file—

```
package com.example.notification;

import android.app.Notification;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.content.Context;
import android.os.Build;
import android.os.Bundle;
import android.view.View;

import androidx.annotation.RequiresApi;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.NotificationCompat;
import androidx.core.app.NotificationManagerCompat;
```

B

```
public class MainActivity extends AppCompatActivity {

   private static final String CHANNEL_ID = "MyChannelID";

   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);

      createNotificationChannel();

      findViewById(R.id.btnShowNotification).setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View v) {
            showNotification();
         }
      });
   }

   private void createNotificationChannel() {
      if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
         CharSequence name = "My Channel";
         String description = "Channel description";
         int importance = NotificationManager.IMPORTANCE_DEFAULT;
         NotificationChannel channel = new NotificationChannel(CHANNEL_ID, name,
importance);
         channel.setDescription(description);

         NotificationManager notificationManager =
getSystemService(NotificationManager.class);
         notificationManager.createNotificationChannel(channel);
      }
   }

   private void showNotification() {
      NotificationCompat.Builder builder = new NotificationCompat.Builder(this,
CHANNEL_ID)
            .setSmallIcon(R.drawable.ic_notification)
            .setContentTitle("My Notification")
            .setContentText("This is a notification.")
            .setPriority(NotificationCompat.PRIORITY_DEFAULT);

      NotificationManagerCompat notificationManager =
NotificationManagerCompat.from(this);
      notificationManager.notify(1, builder.build());
   }
}
```

B



2. Create an android application to implement the concept of Web view withvarious functionality

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity">

  <WebView
     android:id="@+id/webView"
     android:layout_width="match_parent"
     android:layout_height="match_parent"/>

   <Button
     android:id="@+id/btnBack"
     android:layout_width="wrap_content"
     android:layout_height="wrap_content"
     android:text="Back"
     android:layout_margin="16dp"/>

   <Button
```

B

```
        android:id="@+id/btnForward"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Forward"
        android:layout_margin="16dp"
        android:layout_toRightOf="@id/btnBack"/>

    <Button
        android:id="@+id/btnRefresh"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Refresh"
        android:layout_margin="16dp"
        android:layout_toRightOf="@id/btnForward"/>
</RelativeLayout>
```

Java file—

```
import android.annotation.SuppressLint;
import android.os.Bundle;
import android.view.View;
import android.webkit.WebChromeClient;
import android.webkit.WebSettings;
import android.webkit.WebView;
import android.webkit.WebViewClient;
import android.widget.Button;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private WebView webView;

    @SuppressLint("SetJavaScriptEnabled")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        webView = findViewById(R.id.webView);
        WebSettings webSettings = webView.getSettings();
        webSettings.setJavaScriptEnabled(true);

        // Load a webpage
        webView.loadUrl("https://www.example.com");

        // Enable navigation buttons
        Button backButton = findViewById(R.id.btnBack);
```

B

```
        backButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (webView.canGoBack()) {
                    webView.goBack();
                }
            }
        });

        Button forwardButton = findViewById(R.id.btnForward);
        forwardButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (webView.canGoForward()) {
                    webView.goForward();
                }
            }
        });

        // Enable refresh button
        Button refreshButton = findViewById(R.id.btnRefresh);
        refreshButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                webView.reload();
            }
        });

        // Set WebView clients
        webView.setWebViewClient(new WebViewClient());
        webView.setWebChromeClient(new WebChromeClient());
    }
}
```

B

3. **Create an android application to perform database connectivity using SQLite database andSQLite Helper class**

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout  xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="@drawable/signup">

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Register here!"
        android:layout_marginTop="60dp"
        android:gravity="center"
```

B

```
    android:textColor="@color/black"

    android:textSize="30dp"
    tools:ignore="MissingConstraints" />

  <EditText
    android:id="@+id/Name"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="textPersonName"
    android:hint="Enter your name :-"
    android:textColorHint="@color/black"
    android:textAlignment="center"
    android:layout_marginTop="50dp"

    tools:ignore="MissingConstraints"
   />

  <EditText
    android:id="@+id/Email"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="textEmailAddress"
    android:hint="Enter Email :-"
    android:textColorHint="@color/black"
    android:textAlignment="center"
    tools:ignore="MissingConstraints"
    android:layout_marginTop="30dp"
     />
```

B

```
  <EditText
     android:id="@+id/Password"
     android:layout_width="match_parent"
     android:layout_height="wrap_content"
     android:backgroundTint="@color/black"
     android:ems="10"
     android:hint="Enter Your password :-"
     android:textColorHint="@color/black"
     android:textAlignment="center"
     android:layout_marginTop="30dp"
     android:inputType="textPassword"
     tools:ignore="MissingConstraints"
      />

  <EditText
     android:id="@+id/Phone"
     android:layout_width="match_parent"
     android:layout_height="wrap_content"
     android:ems="10"
     android:inputType="phone"
     android:hint="Enter your Mobile Number :-"
     android:textColorHint="@color/black"
     android:textAlignment="center"
     android:layout_marginTop="30dp"
     tools:ignore="MissingConstraints"
      />
 <Button
     android:id="@+id/Submit"
     android:layout_width="wrap_content"
     android:layout_height="wrap_content"
```

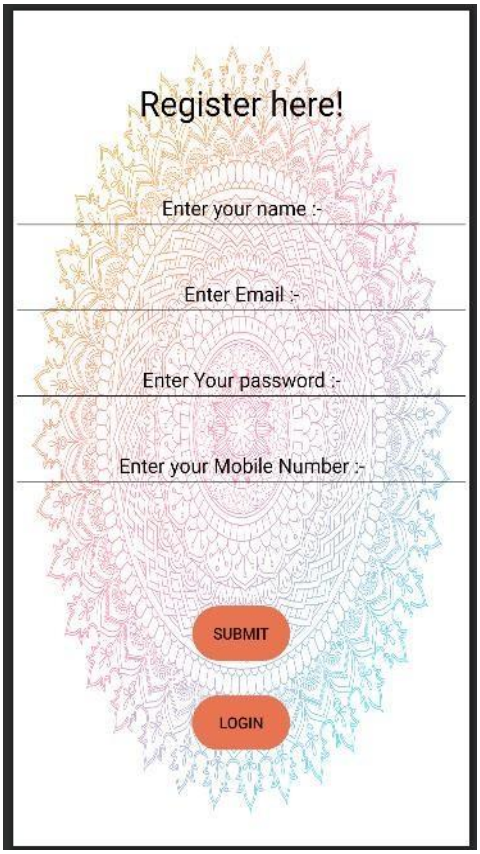
```
B

    android:text="Submit"
    android:layout_gravity="center"
    android:layout_marginTop="100dp"
    tools:ignore="MissingConstraints"
    android:background="@drawable/style"/>

  <Button
    android:id="@+id/Login"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Login"
    android:layout_gravity="center"
    android:gravity="center"
    android:layout_marginTop="30dp"
    tools:ignore="MissingConstraints"
    android:background="@drawable/style"/>
</LinearLayout>
```

Sign -Up page java: -

```
package com.example.dlogin;
import android.content.ContentValues;
import android.content.Intent;
import  android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;
public class signup extends AppCompatActivity {
private EditText editTextUsername, editTextPassword, Email, Phone;
```

```
  private Button buttonRegister, login;
 private DatabaseHelper databaseHelper;
@Override
  protected void onCreate(Bundle savedInstanceState) {
     super.onCreate(savedInstanceState);
     setContentView(R.layout.sign_up);
     editTextUsername = findViewById(R.id.Name);
     editTextPassword = findViewById(R.id.Password);
     Email = findViewById(R.id.Email);
      Phone = findViewById(R.id.Phone);
     buttonRegister = findViewById(R.id.Submit);
     login = findViewById(R.id.Login);
     databaseHelper = new DatabaseHelper(this);
     buttonRegister.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
           String username = editTextUsername.getText().toString().trim();
           String password = editTextPassword.getText().toString().trim();
           String email = Email.getText().toString().trim();
           String phoneno = Phone.getText().toString().trim();
         if (!username.isEmpty() && !password.isEmpty()) {
              SQLiteDatabase db = databaseHelper.getWritableDatabase();
              ContentValues values = new ContentValues();
              values.put(DatabaseHelper.COLUMN_USERNAME, username);
              values.put(DatabaseHelper.COLUMN_PASSWORD,  password);
              values.put(DatabaseHelper.COLUMN_Email, email);
              values.put(DatabaseHelper.COLUMN_PHONE, phoneno);
              long newRowId = db.insert(DatabaseHelper.TABLE_USERS, null, values);
              db.close();
            if (newRowId != -1) {
                 Toast.makeText(signup.this, "Registration successful", Toast.LENGTH_SHORT).show();
```

```
            } else {
                Toast.makeText(signup.this, "Registration failed", Toast.LENGTH_SHORT).show();
            }
        } else {
            Toast.makeText(signup.this, "Please fill in all fields", Toast.LENGTH_SHORT).show();
        }
    }
});
login.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent i = new Intent(getApplicationContext(), loginActivity.class);
        startActivity(i);
    } }); }}
```



4. Create an android application to perform database connectivity using SQLite database andSQLite Helper class and perform insert, update and delete operations

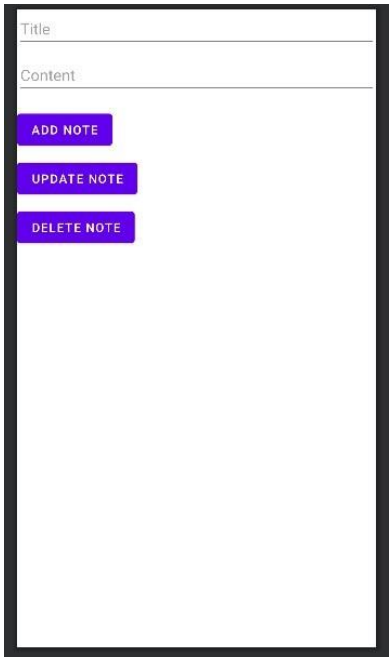
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  tools:context=".MainActivity">

  <EditText
    android:id="@+id/editTextTitle"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Title" />

  <EditText
    android:id="@+id/editTextContent"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/editTextTitle"
    android:layout_marginTop="8dp"
    android:hint="Content" />

  <Button
    android:id="@+id/addButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/editTextContent"
    android:layout_marginTop="16dp"
    android:text="Add Note" />

  <Button
    android:id="@+id/updateButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/addButton"
    android:layout_marginTop="8dp"
    android:text="Update Note" />

  <Button
    android:id="@+id/deleteButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/updateButton"
    android:layout_marginTop="8dp"
    android:text="Delete Note" />
```

```
</RelativeLayout>
```

Java file----

```
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class DatabaseHelper extends SQLiteOpenHelper {

   // Database Information
   private static final String DATABASE_NAME = "mydatabase";
   private static final int DATABASE_VERSION = 1;

   // Table Information
   private static final String TABLE_NAME = "notes";
   private static final String COLUMN_ID = "id";
   private static final String COLUMN_TITLE = "title";
   private static final String COLUMN_CONTENT = "content";

   // Constructor
   public DatabaseHelper(Context context) {
      super(context, DATABASE_NAME, null, DATABASE_VERSION);
   }

   // Creating the table
   @Override
   public void onCreate(SQLiteDatabase db) {
      String createTableQuery = "CREATE TABLE " + TABLE_NAME + " (" +
            COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
            COLUMN_TITLE + " TEXT, " +
            COLUMN_CONTENT + " TEXT)";
      db.execSQL(createTableQuery);
   }

   // Upgrading the table if needed
   @Override
   public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
      db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
      onCreate(db);
   }

   // Inserting a new note
   public long insertNote(Note note) {
      SQLiteDatabase db = this.getWritableDatabase();
      ContentValues values = new ContentValues();
```

```
        values.put(COLUMN_TITLE, note.getTitle());
        values.put(COLUMN_CONTENT, note.getContent());
        long insertedId = db.insert(TABLE_NAME, null, values);
        db.close();
        return insertedId;
    }

    // Updating a note
    public int updateNote(Note note) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(COLUMN_TITLE, note.getTitle());
        values.put(COLUMN_CONTENT, note.getContent());
        int rowsAffected = db.update(TABLE_NAME, values, COLUMN_ID + " = ?",
                new String[]{String.valueOf(note.getId())});
        db.close();
        return rowsAffected;
    }

    // Deleting a note
    public int deleteNote(int noteId) {
        SQLiteDatabase db = this.getWritableDatabase();
        int deletedRows = db.delete(TABLE_NAME, COLUMN_ID + " = ?",
                new String[]{String.valueOf(noteId)});
        db.close();
        return deletedRows;
    }

    // Getting all notes
    public Cursor getAllNotes() {
        SQLiteDatabase db = this.getReadableDatabase();
        return db.query(TABLE_NAME, null, null, null, null, null, null);
    }
}

Db file---
public class Note {
    private int id;
    private String title;
    private String content;

    public Note(String title, String content) {
        this.title = title;
        this.content = content;
    }

    // Getters and setters for id, title, and content
}
```



5. Create the steps of flutter installation.

1. *System Requirements:*
   - Flutter supports Windows, macOS, and Linux. Ensure your system meets the [system requirements](https://flutter.dev/docs/get-started/install) for your operating system.

2. *Download Flutter:*
   - Download the latest stable version of Flutter from the [official Flutter website](https://flutter.dev/docs/get-started/install).
3. *Extract the ZIP file:*
   - Extract the downloaded ZIP file to a location on your computer.
4. *Update System Path:*
   - Add the flutter/bin directory to your system PATH. This step allows you to run the flutter command from any terminal window.
     - *On macOS and Linux:*
       bash
       export PATH="$PATH:`<path_to_flutter_directory>`/flutter/bin"
     - *On Windows:*
       - Open the System Properties.
       - Click on "Advanced system settings" -> "Environment Variables..."
       - In the "System variables" section, select the "Path" variable and click "Edit..."

- Add a new entry with the path to the flutter\bin directory.

5. **Run flutter doctor

- Open a terminal window and run the following command to check if there are any dependencies you still need to install:

bash

flutter doctor

6. *Install Flutter dependencies:*

- The flutter doctor command will guide you on installing any missing dependencies. Follow the instructions provided.

7. *Android Studio/VS Code Setup (Optional but recommended):*

- For a better development experience, it's recommended to use Android Studio or Visual Studio Code with the Flutter and Dart plugins.

- If you choose Android Studio, make sure to install the Flutter and Dart plugins from the marketplace.

8. **Run flutter pub get:**

- In your Flutter project directory, run the following command to get the dependencies specified in your pubspec.yaml file:

bash

flutter pub get

9. *Verify Installation:*

- Run the following command to verify that Flutter is installed correctly:

bash

flutter --version

- Run the following command to verify that all dependencies are satisfied:

bash

flutter doctor

10. *Create a Flutter project:*

- Create a new Flutter project by running the following command in your terminal:

bash

flutter create my_flutter_project

11. *Run your Flutter project:*

- Change into your project directory and run your app using the following commands:

bash

cd my_flutter_projectflutter run