



**Department of Computer
Engineering/Information
Technology**

- Enhance an application which is already created in practical-10 by implementing following features:**
1. Copy practical-10 into different folder with name of practical-11.
 2. Create NodeData Class in which store Table Name, Column name, Table Query by using String Constant & in companion object.
 3. Create DatabaseHelper class by implementing these methods:
getAllNotes(), getNotesCount(), insertNote(), deleteNote(), UpdateNote()
 4. Modify ListView & its adapter to display all notes.
 5. Modify Common Custom Dialog Box for add, edit note, delete note.
 6. Alarm, Reminder, notification functionality should be same as built in practical-10.
 7. Store Login details like Username & password by using Shared Preferences.

Activity_splash.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:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/splash_layout"
    android:background="@drawable/gradient_background"
    tools:context=".SplashActivity">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="150dp"
        android:id="@+id/logo"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
    />

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

```
package com.example.practical11_20072532001
```

```
import android.content.Context  
import android.content.Intent  
import android.graphics.Color  
import android.graphics.drawable.AnimationDrawable  
import android.os.Build  
import android.os.Bundle  
import android.view.Handler  
import android.view.View  
import android.view.WindowManager  
import android.view.animation.Animation  
import android.view.animation.AnimationUtils  
import android.widget.ImageView  
import androidx.appcompat.app.AppCompatActivity  
import androidx.constraintlayout.widget.ConstraintLayout
```

```
class SplashActivity : AppCompatActivity()
```

```
    lateinit var imageView : ImageView  
    lateinit var mainLayout : ConstraintLayout  
    lateinit var animation : AnimationDrawable  
    lateinit var animatedLogo : Animation
```

```
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.activity_splash)
```

```
        setStatusBarTransparent()
```

```
        imageView = findViewById(R.id.logo)  
        mainLayout = findViewById(R.id.splash_layout)
```

```
        imageView.setBackgroundResource(R.drawable.splash_logo_animation)  
        animation = imageView.background as AnimationDrawable  
        animation.start()
```

```
        animatedLogo = AnimationUtils.loadAnimation(this, R.anim.logo_animation)  
        imageView.startAnimation(animatedLogo)
```

```
        animatedLogo.setAnimationListener(object : Animation.AnimationListener {  
            override fun onAnimationStart(animation: Animation?) {}  
            override fun onAnimationEnd(animation: Animation?) {  
                imageView.animate().scaleX(3.0f).scaleY(3.0f).alpha(0.1f).duration = 1500  
            }  
        })
```

```
        val sharedPrefs = getSharedPreferences("user_data", Context.MODE_PRIVATE)
```

```
if(sharedPrefs.getInt("user_id", -1)==-1){
    Handler().postDelayed({
        val intent = Intent(this@SplashActivity, MainActivity::class.java)
        startActivity(intent)
        finish()
    }, 500)
}
else{
    Handler().postDelayed({
        val intent = Intent(this@SplashActivity, DashBoardActivity::class.java)
        startActivity(intent)
        finish()
    }, 500)
}
}
override fun onAnimationRepeat(animation: Animation?) {
}

private fun setWindowFlag(flagTranslucentStatus: Int, b: Boolean) {
    val winParameters = window.attributes
    if(b){
        winParameters.flags = winParameters.flags or flagTranslucentStatus
    }
    else{
        winParameters.flags = winParameters.flags and flagTranslucentStatus.inv()
    }
    window.attributes = winParameters
}

private fun setStatusBarTransparent(){
    if (Build.VERSION.SDK_INT <= 19..20) {
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT) {
            setWindowFlag(WindowManager.LayoutParams.FLAG_TRANSLUCENT_STATUS, true)
        }
    }
    if (Build.VERSION.SDK_INT >= 19) {
        window.decorView.systemUiVisibility = View.SYSTEM_UI_FLAG_LAYOUT_STABLE or
        View.SYSTEM_UI_FLAG_LAYOUT_FULLSCREEN
    }
    if(Build.VERSION.SDK_INT >= 21){
        setWindowFlag(WindowManager.LayoutParams.FLAG_TRANSLUCENT_STATUS, false)
        window.statusBarColor = Color.TRANSPARENT
    }
}
```

activity_main.xml(login):

```
<?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:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_gravity="center"
    style="@style/Theme.AppCompat.Light.NoActionBar"
    tools:context=".MainActivity">
```

<ImageView

```
    android:id="@+id/imageView"
    android:layout_width="175dp"
    android:layout_height="67dp"
    app:layout_constraintBottom_toTopOf="@+id/view2"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.537"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="1.0"
    app:srcCompat="@drawable/ganpat_logo_foreground" />
```

<View

```
    android:id="@+id/view2"
    android:layout_width="220dp"
    android:layout_height="570dp"
    android:background="@drawable/gradient_view2"
    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" />
```

<TextView

```
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Login"
    android:textColor="@color/white"
    android:textSize="25sp"
    app:layout_constraintBottom_toTopOf="@+id/cardViewForm"
```

```
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.2"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/imageView"
    app:layout_constraintVertical_bias="0.224">></TextView>
```

<TextView

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="96dp"
    android:layout_marginLeft="96dp"
    android:text="Signup"
    android:textSize="25sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.792"
    app:layout_constraintStart_toEndOf="@+id/textView3"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.144">></TextView>
```

<androidx.cardview.widget.CardView

```
    android:id="@+id/cardViewForm"
    android:layout_width="319dp"
    android:layout_height="321dp"
    android:padding="20dp"
    app:cardCornerRadius="20dp"
    app:cardElevation="20dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent">>
```

<LinearLayout

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">
```

<LinearLayout

```
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="20dp"
    android:paddingBottom="0dp">>
```

```
<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
        app:startIconDrawable="@drawable/ic_baseline_email_24">

    <com.google.android.material.textfield.TextInputEditText
        android:layout_width="match_parent"
        android:id="@+id/email"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:hint="Enter Email" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="99dp"
    app:startIconDrawable="@drawable/ic_baseline_vpn_key_24">

    <com.google.android.material.textfield.TextInputEditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/password"
        android:layout_marginTop="20dp"
        android:hint="Enter Password" />

</com.google.android.material.textfield.TextInputLayout>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Forgot Password?"
    android:layout_gravity="right"
    />

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#00000000"
    android:layout_marginTop="-33dp"
    >

    <ImageView
        android:layout_width="match_parent"
```

```
        android:layout_height="150dp"
        android:src="@drawable/ic_wave_svg" />
    </LinearLayout>
</LinearLayout>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:id="@+id/cardView"
    android:layout_width="200dp"
    android:layout_height="wrap_content"
    android:layout_marginLeft="70dp"
    android:backgroundTint="#00000000"
    app:cardElevation="30dp"
    app:layout_constraintBottom_toBottomOf="@+id/cardViewForm"
    app:layout_constraintEnd_toEndOf="@+id/cardViewForm"
    app:layout_constraintHorizontal_bias="0.113"
    app:layout_constraintStart_toStartOf="@+id/cardViewForm"
    app:layout_constraintTop_toBottomOf="@+id/cardViewForm">

    <androidx.appcompat.widget.AppCompatButton
        android:id="@+id/login"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="@drawable/button_gradient"
        android:text="Login"
        android:textColor="@color/white"
        android:textSize="20sp" />
</androidx.cardview.widget.CardView>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:layout_marginLeft="8dp"
    android:text="Don't Have an Account?"
    android:textSize="20sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toStartOf="@+id/account"
    app:layout_constraintHorizontal_bias="1.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/view2"
    app:layout_constraintVertical_bias="0.509" />

<TextView
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="right"
        android:layout_marginEnd="52dp"
        android:layout_marginRight="52dp"
        android:text="Signup"
        android:textAllCaps="true"
        android:textColor="#FF00FF"
        android:textSize="20sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/cardView"
        app:layout_constraintVertical_bias="0.831" />

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

MainActivity.kt(login):

```
package com.example.practical_11_20012532001

import android.content.Context
import android.content.Intent
import android.graphics.Color
import android.os.Build
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.os.Handler
import android.view.View
import android.view.WindowManager
import android.view.animation.Animation
import android.widget.TextView
import android.widget.Toast
import androidx.appcompat.widget.AppCompatButton
import com.google.android.material.textfield.TextInputEditText

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        setStatusBarTransparent()

        val loginBtn = findViewById<AppCompatButton>(R.id.login)
        val email = findViewById<TextInputEditText>(R.id.email)
        val password = findViewById<TextInputEditText>(R.id.password)
        val account = findViewById<TextView>(R.id.account)
```

```
val sharedPrefs = getSharedPreferences("user_data", Context.MODE_PRIVATE)
val dbUserController = DBUserController(this)

account.setOnClickListener{
    Intent(this, SignupActivity::class.java).apply {
        startActivity(this)
    }
}

loginBtn.setOnClickListener{
    when {
        email.text!! .toString() == "" -> {
            Toast.makeText(this, "Email Is Required!", Toast.LENGTH_SHORT).show()
        }
        password.text!! .toString() == "" -> {
            Toast.makeText(this, "Password Is Required!", Toast.LENGTH_SHORT).show()
        }
        else -> {
            val user = dbUserController.verifyUser(email.text.toString(), password.text.toString())
            if(user!=null){
                sharedPrefs.edit().putInt("user_id",user.getId()).commit()

                Handler().postDelayed({
                    val intent = Intent(this, DashBoardActivity::class.java)
                    startActivity(intent)
                    finish()
                }, 0)
            }
            else{
                Toast.makeText(this,"Fault",Toast.LENGTH_SHORT).show()
                email.setText("", TextView.BufferType.EDITABLE)
                password.setText("", TextView.BufferType.EDITABLE)
                Toast.makeText(this,"Email and Password doesn't matched!",Toast.LENGTH_SHORT).show()
            }
        }
    }
}
```

```
}

private fun setWindowFlag(flagTranslucentStatus: Int, b: Boolean) {
    val winParameters = window.attributes
    if(b){
        winParameters.flags = winParameters.flags or flagTranslucentStatus
    }
    else{
        winParameters.flags = winParameters.flags and flagTranslucentStatus.inv()
    }
    window.attributes = winParameters
}

private fun statusBarTransparent(){
    if (Build.VERSION.SDK_INT in 19..20) {
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT) {
            setWindowFlag(WindowManager.LayoutParams.FLAG_TRANSLUCENT_STATUS, true)
        }
    }
    if (Build.VERSION.SDK_INT >= 19) {
        window.decorView.systemUiVisibility = View.SYSTEM_UI_FLAG_LAYOUT_STABLE or
View.SYSTEM_UI_FLAG_LAYOUT_FULLSCREEN
    }

    if(Build.VERSION.SDK_INT >= 21){
        setWindowFlag(WindowManager.LayoutParams.FLAG_TRANSLUCENT_STATUS, false)
        window.statusBarColor = Color.TRANSPARENT
    }
}
}
```

activity_signup.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:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_gravity="center"
    style="@style/Theme.AppCompat.Light.NoActionBar"
    tools:context=".MainActivity">

    <View
        android:id="@+id/view2"
        android:layout_width="195dp"
```

```
        android:layout_height="662dp"
        android:background="@drawable/gradient_view1"
        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="1.0" />

<ImageView
    android:id="@+id/imageView"
    android:layout_width="175dp"
    android:layout_height="67dp"
    app:layout_constraintBottom_toTopOf="@+id/cardViewForm"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.022"
    app:srcCompat="@drawable/ganpat_logo_foreground" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Login"
    android:textSize="25sp"
    app:layout_constraintBottom_toTopOf="@+id/cardViewForm"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.192"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/imageView"
    app:layout_constraintVertical_bias="0.395"></TextView>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="96dp"
    android:layout_marginLeft="96dp"
    android:text="Signup"
    android:textColor="@color/white"
    android:textSize="25sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.682"
    app:layout_constraintStart_toEndOf="@+id/textView3"
    app:layout_constraintTop_toTopOf="parent"
```

```
    app:layout_constraintVertical_bias="0.127"></TextView>

<androidx.cardview.widget.CardView
    android:id="@+id/cardViewForm"
    android:layout_width="342dp"
    android:layout_height="537dp"
    android:padding="20dp"
    app:cardCornerRadius="20dp"
    app:cardElevation="20dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.492"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.809">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical"
            android:padding="20dp"
            android:paddingBottom="0dp">

            <com.google.android.material.textfield.TextInputLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                app:startIconDrawable="@drawable/ic_baseline_account_circle_24">

                <com.google.android.material.textfield.TextInputEditText
                    android:layout_width="match_parent"
                    android:id="@+id/username"
                    android:layout_height="wrap_content"
                    android:layout_marginTop="0dp"
                    android:hint="User Full Name" />

            </com.google.android.material.textfield.TextInputLayout>

            <com.google.android.material.textfield.TextInputLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                app:startIconDrawable="@drawable/ic_baseline_call_24">
```

```
<com.google.android.material.textfield.TextInputEditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:id="@+id/phoneno"
    android:hint="Phone Number" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    app:startIconDrawable="@drawable/ic_baseline_location_city_24">

    <com.google.android.material.textfield.TextInputEditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:id="@+id/city"
        android:hint="City" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    app:startIconDrawable="@drawable/ic_baseline_email_24">

    <com.google.android.material.textfield.TextInputEditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:id="@+id/email"
        android:hint="Email" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    app:startIconDrawable="@drawable/ic_baseline_vpn_key_24">
```

```
<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:id="@+id/password"
    android:hint="Password"/>

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    app:startIconDrawable="@drawable/ic_baseline_vpn_key_24">

    <com.google.android.material.textfield.TextInputEditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:id="@+id/cpassword"
        android:hint="Confirm Password"/>

</com.google.android.material.textfield.TextInputLayout>

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#00000000">

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="150dp"
        android:src="@drawable/ic_wave_svg"/>

</LinearLayout>

</LinearLayout>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:layout_width="200dp"
```

```
        android:layout_height="wrap_content"
        android:backgroundTint="#00000000"
        android:foregroundGravity="center"
        app:cardElevation="30dp"
        app:layout_constraintBottom_toBottomOf="@+id/cardViewForm"
        app:layout_constraintEnd_toEndOf="@+id/cardViewForm"
        app:layout_constraintStart_toStartOf="@+id/cardViewForm"
        app:layout_constraintTop_toBottomOf="@+id/cardViewForm">

    <androidx.appcompat.widget.AppCompatButton
        android:id="@+id/signup"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="@drawable/button_gradient"
        android:text="Signup"
        android:textColor="@color/white"
        android:textSize="20sp" />
</androidx.cardview.widget.CardView>

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

SignupActivity.kt:

```
package com.example.practical_11_20012532001

import android.content.Context
import android.content.Intent
import android.content.SharedPreferences
import android.graphics.Color
import android.os.Build
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.View
import android.view.WindowManager
import android.widget.TextView
import android.widget.Toast
import androidx.appcompat.widget.AppCompatButton
import com.google.android.material.textfield.TextInputEditText

class SignupActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_signup)

        setStatusBarTransparent()

        val signupBtn = findViewById<AppCompatButton>(R.id.signup)
```

```
val username = findViewById<TextInputEditText>(R.id.username)
val phoneno = findViewById<TextInputEditText>(R.id.phoneno)
val city = findViewById<TextInputEditText>(R.id.city)
val email = findViewById<TextInputEditText>(R.id.email)
val password = findViewById<TextInputEditText>(R.id.password)
val cpassword = findViewById<TextInputEditText>(R.id.cpassword)

val dbUserController : DBUserController = DBUserController(this)

signupBtn.setOnClickListener {

    when {
        username.text.toString() == "" -> {
            Toast.makeText(this, "Username is required!", Toast.LENGTH_SHORT).show()
        }
        phoneno.text.toString() == "" -> {
            Toast.makeText(this, "Phone No. is required!", Toast.LENGTH_SHORT).show()
        }
        city.text.toString() == "" -> {
            Toast.makeText(this, "City is required!", Toast.LENGTH_SHORT).show()
        }
        email.text.toString() == "" -> {
            Toast.makeText(this, "Email is required!", Toast.LENGTH_SHORT).show()
        }
        password.text.toString() == "" -> {
            Toast.makeText(this, "Password is required!", Toast.LENGTH_SHORT).show()
        }
        password.text.toString() != cpassword.text.toString() -> {
            Toast.makeText(this, "Password Doesn't Match!", Toast.LENGTH_SHORT).show()
        }
        else -> {
            val userData = LoginInfo(
                username.text.toString(),
                phoneno.text.toString().toInt(),
                city.text.toString(),
                email.text.toString(),
                password.text.toString()
            )

            dbUserController.addUser(userData)

            Intent(this, MainActivity::class.java).apply{
                startActivity(this)
            }
        }
    }
}
```

```
}

private fun setWindowFlag(flagTranslucentStatus: Int, b: Boolean) {
    val winParameters = window.attributes
    if(b){
        winParameters.flags = winParameters.flags or flagTranslucentStatus
    }
    else{
        winParameters.flags = winParameters.flags and flagTranslucentStatus.inv()
    }
    window.attributes = winParameters
}

private fun setStatusBarTransparent(){
    if (Build.VERSION.SDK_INT in 19..20) {
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT) {
            setWindowFlag(WindowManager.LayoutParams.FLAG_TRANSLUCENT_STATUS, true)
        }
    }
    if (Build.VERSION.SDK_INT >= 19) {
        window.decorView.systemUiVisibility = View.SYSTEM_UI_FLAG_LAYOUT_STABLE or
        View.SYSTEM_UI_FLAG_LAYOUT_FULLSCREEN
    }

    if(Build.VERSION.SDK_INT >= 21){
        setWindowFlag(WindowManager.LayoutParams.FLAG_TRANSLUCENT_STATUS, false)
        window.statusBarColor = Color.TRANSPARENT
    }
}

override fun onDestroy(){
    super.onDestroy()
    Intent(this,MainActivity::class.java).apply {
        startActivity(this)
    }
}
}
```

activity_dashboard.xml:

```
<?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"
```

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

<ScrollView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/scrollView"
    >

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        >

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical"
            >

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="300dp"
                android:padding="20dp"
                android:background="@drawable/gradient_background"
                >

                <LinearLayout
                    android:layout_width="260dp"
                    android:layout_height="wrap_content"
                    android:layout_gravity="center"
                    android:gravity="center"
                    android:orientation="vertical"
                    android:padding="10dp"
                    >

                    <TextView
                        android:id="@+id/name"
                        android:layout_width="wrap_content"
                        android:layout_height="wrap_content"
                        android:baselineAligned="false"
                        android:text="Rajesh Shah"
                        android:textColor="@color/white"
                        android:textSize="30sp" />

                    <TextView
                        android:layout_width="wrap_content"
                        android:layout_height="wrap_content"
                        android:text="rajesh@123@gmail.com"
                        android:id="@+id/email"
                        >
                
```

```
        android:textColor="#FFFFFF00"
        android:textSize="15sp">></TextView>

    <RatingBar
        android:layout_width="match_parent"
        android:scaleX="0.8"
        android:scaleY="0.8"
        android:layout_height="wrap_content" />

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:layout_gravity="center"
    >

    <ImageView
        android:id="@+id/imageView2"
        android:layout_width="60dp"
        android:layout_height="60dp"
        android:layout_gravity="center"
        app:tint="#FFFFFF"
        android:src="@drawable/user_account"
        />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Logout"
        android:textSize="15sp"
        android:id="@+id/logout"
        android:textColor="#FF0000"
        android:layout_gravity="center"
        />

    </LinearLayout>
</LinearLayout>

<androidx.cardview.widget.CardView
    android:layout_width="300dp"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="-50dp"
    android:background="@color/white"
```

```
    android:padding="20dp"
    app:cardCornerRadius="10dp"
    app:cardElevation="20dp">

    <LinearLayout
        android:layout_width="300dp"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:padding="20dp">

        <RelativeLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:orientation="horizontal">

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginEnd="80dp"
                android:layout_marginRight="80dp"
                android:text="Personal Details"
                android:textColor="@color/black"
                android:textSize="20sp"
                android:textStyle="bold"></TextView>

            <ImageView
                android:layout_width="35dp"
                android:layout_height="35dp"
                android:layout_alignParentEnd="true"
                android:layout_alignParentRight="true"
                android:layout_marginEnd="6dp"
                android:layout_marginRight="6dp"
                android:src="@drawable/ic_baseline_account_circle_24" />
        </RelativeLayout>

        <View
            android:layout_width="match_parent"
            android:layout_height="1dp"
            android:layout_marginTop="10dp"
            android:layout_marginBottom="10dp"
            android:background="@android:color/darker_gray" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
```

```
    android:layout_marginBottom="10dp"
    android:text="Full Name"
    android:textSize="18sp">></TextView>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Rajesh Mohanlal Shah"
    android:textColor="@color/black"
    android:id="@+id/username"
    android:textSize="18sp">></TextView>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="20dp"
    android:orientation="horizontal">

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_marginTop="20dp"
    android:layout_marginRight="20dp"
    android:orientation="vertical">

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Phone Number"
    android:textSize="18sp">></TextView>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="+91 9999999999"
    android:textColor="@color/black"
    android:id="@+id/number"
    android:textSize="18sp">></TextView>

</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_marginTop="20dp"
    android:orientation="vertical">
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="City"
    android:textSize="18sp"></TextView>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Mehsana,Gujarat"
    android:id="@+id/city"
    android:textColor="@color/black"
    android:textSize="18sp"></TextView>

</LinearLayout>

</LinearLayout>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="10dp"
    android:text="Email Id"
    android:textSize="18sp"></TextView>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="rajesh123@gmail.com"
    android:id="@+id/email1"
    android:textColor="@color/black"
    android:textSize="18sp"></TextView>

</LinearLayout>
</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:layout_width="300dp"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="30dp"
    android:background="@color/white"
    app:cardCornerRadius="10dp"
    app:cardElevation="20dp">
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    >

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:paddingLeft="20dp"
        android:paddingTop="20dp"
        android:paddingRight="15dp"
        android:orientation="horizontal">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginEnd="80dp"
            android:layout_marginRight="80dp"
            android:text="Photos"
            android:textColor="@color/black"
            android:textSize="20sp"
            android:textStyle="bold"></TextView>

        <ImageView
            android:id="@+id/heart"
            android:layout_width="40dp"
            android:layout_height="40dp"
            android:layout_alignParentEnd="true"
            android:layout_alignParentRight="true"
            android:layout_marginEnd="6dp"
            android:layout_marginRight="6dp"/>
    </RelativeLayout>

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:scaleType="fitXY"
        android:id="@+id/slider_image"
        />

    </LinearLayout>
</androidx.cardview.widget.CardView>
```

```
<androidx.cardview.widget.CardView
    android:layout_width="300dp"
    android:layout_height="250dp"
    android:layout_gravity="center"
    android:layout_marginTop="30dp"
    android:background="@color/white"
    app:cardElevation="20dp"
    android:padding="20dp"
    app:cardCornerRadius="10dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:padding="20dp">

        <RelativeLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:orientation="horizontal">

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Set Alarm Time"
                android:textColor="@color/black"
                android:textSize="20sp"
                android:textStyle="bold"></TextView>

            <ImageView
                android:layout_width="35dp"
                android:layout_height="35dp"
                android:layout_alignParentEnd="true"
                android:layout_alignParentRight="true"
                android:layout_gravity="end"
                android:layout_marginEnd="6dp"
                android:layout_marginRight="6dp"
                android:src="@drawable/ic_baseline_description_24" />
        </RelativeLayout>

        <View
            android:layout_width="match_parent"
            android:layout_height="1dp"
            android:layout_marginTop="10dp"
            android:layout_marginBottom="10dp"
            android:background="@android:color/darker_gray" />
    
```

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Current Date Time"  
    android:textSize="25sp"  
/>  
  
<TextClock  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:format12Hour="hh:mm:ss a MMM,dd yyyy"  
/>  
  
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Alarm"  
    android:textSize="25sp"  
/>  
<TextView  
    android:layout_width="wrap_content"  
    android:text="Set Alarm"  
    android:textSize="25sp"  
    android:id="@+id/set_alarm_text"  
    android:layout_height="wrap_content"  
/>  
  
</LinearLayout>  
</androidx.cardview.widget.CardView>  
  
<androidx.cardview.widget.CardView  
    android:layout_width="300dp"  
    android:layout_height="200dp"  
    android:layout_gravity="center"  
    android:layout_marginTop="30dp"  
    android:background="@color/white"  
    app:cardElevation="20dp"  
    android:padding="20dp"  
    app:cardCornerRadius="10dp">  
  
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"
```

```
    android:orientation="vertical"
    android:padding="20dp">>

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:orientation="horizontal">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginEnd="122dp"
            android:layout_marginRight="122dp"
            android:text="Description"
            android:textColor="@color/black"
            android:textSize="20sp"
            android:textStyle="bold"></TextView>

        <ImageView
            android:layout_width="35dp"
            android:layout_height="35dp"
            android:layout_alignParentEnd="true"
            android:layout_alignParentRight="true"
            android:layout_gravity="end"
            android:layout_marginEnd="6dp"
            android:layout_marginRight="6dp"
            android:src="@drawable/ic_baseline_description_24" />
    </RelativeLayout>

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:layout_marginTop="10dp"
        android:layout_marginBottom="10dp"
        android:background="@android:color/darker_gray" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="15sp"
        android:text="Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris
nisi ut aliquip ex ea commodo consequat.">
    </TextView>
```

```
</LinearLayout>
</androidx.cardview.widget.CardView>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="bottom"
    android:layout_marginTop="-150dp"
    android:layout_marginBottom="-100dp"
    >
    <ImageView
        android:layout_width="match_parent"
        android:layout_height="400dp"
        android:src="@drawable/ic_wave_svg"
        android:layout_gravity="bottom"
    />
</LinearLayout>
</LinearLayout>

</ScrollView>

<com.google.android.material.bottomnavigation.BottomNavigationView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/bottom_navigation"
    android:layout_alignBottom="@+id/scrollView"
    android:layout_alignParentBottom="true"
    app:menu="@menu/bottom_menu"
    />

</RelativeLayout>
```

DashBoard.kt:

```
package com.example.practical_11_20012532001

import android.app.AlarmManager
import android.app.PendingIntent
import android.app.TimePickerDialog
import android.content.Context
import android.content.Intent
```

```
import android.graphics.Color
import android.graphics.drawable.AnimationDrawable
import android.media.Image
import android.os.Build
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.os.Handler
import android.view.View
import android.view.WindowManager
import android.widget.ImageView
import android.widget.TextView
import android.widget.Toast
import com.google.android.material.bottomnavigation.BottomNavigationView
import java.text.SimpleDateFormat
import java.util.*

class DashBoardActivity : AppCompatActivity() {
    private lateinit var setAlarmText: TextView
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_dash_board)

        setStatusBarTransparent()

        val name = findViewById<TextView>(R.id.name)
        val email = findViewById<TextView>(R.id.email)
        val email1 = findViewById<TextView>(R.id.email1)
        val username = findViewById<TextView>(R.id.username)
        val number = findViewById<TextView>(R.id.number)
        val city = findViewById<TextView>(R.id.city)
        val logout = findViewById<TextView>(R.id.logout)
        setAlarmText = findViewById(R.id.set_alarm_text)
        val bottomMenu = findViewById<BottomNavigationView>(R.id.bottom_navigation)
        val slider = findViewById<ImageView>(R.id.slider_image)
        val heart_slider = findViewById<ImageView>(R.id.heart)

        slider.setBackgroundResource(R.drawable.slider_animation)
        heart_slider.setBackgroundResource(R.drawable.heart_animation)

        val animation: AnimationDrawable = slider.background as AnimationDrawable
        animation.start()

        val heartAnim: AnimationDrawable = heart_slider.background as AnimationDrawable
        heartAnim.start()
    }
}
```

```
val sharedPrefs = getSharedPreferences("user_data", Context.MODE_PRIVATE)

bottomMenu.setOnItemSelectedListener{
    when(it.itemId){
        R.id.dashboard -> {
            Intent(this, DashBoardActivity::class.java).apply {
                startActivity(this)
            }
            return@setOnItemSelectedListener true
        }
        R.id.notes -> {
            Intent(this, NotesListActivity::class.java).apply {
                startActivity(this)
            }
            return@setOnItemSelectedListener true
        }
        else -> {
            return@setOnItemSelectedListener true
        }
    }
}

logout.setOnClickListener{

    sharedPrefs.edit().remove("user_id").commit()
    Handler().postDelayed({
        val intent = Intent(this, MainActivity::class.java)
        startActivity(intent)
        finish()
    }, 0)
}

val dbUserController = DBUserController(this)
val userId = sharedPrefs.getInt("user_id", -1)

if(userId != -1){
    val getUser = dbUserController.getUser(userId)
    name.text = getUser.getUsername()
    username.text = getUser.getUsername()
    email.text = getUser.getEmail()
    number.text = getUser.getPhone().toString()
    city.text = getUser.getCity()
    email1.text = getUser.getEmail()
}
else{
    Handler().postDelayed({

```

```
    val Intent = Intent(this, MainActivity::class.java)
    startActivity(Intent)
    finish()
}, 0)
}

setAlarmText.setOnClickListener {
    showTimerDialog()
}
}

private fun setAlarm(millisTime: Long, str: String)
{
    val intent = Intent(this, MyReceiver::class.java)
    intent.putExtra("Service1", str)
    val pendingIntent = PendingIntent.getBroadcast(applicationContext, 234324243, intent, 0)
    val alarmManager = getSystemService(ALARM_SERVICE) as AlarmManager
    if(str == "Start") {
        alarmManager.setExact(
            AlarmManager.RTC_WAKEUP,
            millisTime,
            pendingIntent
        )
    }else if(str == "Stop")
    {
        alarmManager.cancel(pendingIntent)
    }
}

private fun showTimerDialog()
{
    val cldr: Calendar = Calendar.getInstance()
    val hour: Int = cldr.get(Calendar.HOUR_OF_DAY)
    val minutes: Int = cldr.get(Calendar.MINUTE)
    //time picker dialog
    val picker = TimePickerDialog(
        this,
        { _, sHour, sMinute -> sendDialogDataToActivity(sHour, sMinute)},
        hour,
        minutes,
        false
    )
    picker.show()
}
```

```
private fun sendDialogDataToActivity(hour: Int, minute: Int) {
    val alarmCalendar = Calendar.getInstance()
    val year: Int = alarmCalendar.get(Calendar.YEAR)
    val month: Int = alarmCalendar.get(Calendar.MONTH)
    val day: Int = alarmCalendar.get(Calendar.DATE)
    alarmCalendar.set(year, month, day, hour, minute, 0)
    setAlarmText.text = SimpleDateFormat("hh:mm:ss a").format(alarmCalendar.time)
    setAlarm(alarmCalendar.timeInMillis, "Start")
    Toast.makeText(
        this,
        "Time: hours:${hour}, minutes:${minute}, millis:${alarmCalendar.timeInMillis}",
        Toast.LENGTH_SHORT
    ).show()
}

private fun setWindowFlag(flagTranslucentStatus: Int, b: Boolean) {
    val winParameters = window.attributes
    if(b){
        winParameters.flags = winParameters.flags or flagTranslucentStatus
    }
    else{
        winParameters.flags = winParameters.flags and flagTranslucentStatus.inv()
    }
    window.attributes = winParameters
}

private fun setStatusBarTransparent(){
    if (Build.VERSION.SDK_INT in 19..20) {
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT) {
            setWindowFlag(WindowManager.LayoutParams.FLAG_TRANSLUCENT_STATUS, true)
        }
    }
    if (Build.VERSION.SDK_INT >= 19) {
        window.decorView.systemUiVisibility = View.SYSTEM_UI_FLAG_LAYOUT_STABLE or
        View.SYSTEM_UI_FLAG_LAYOUT_FULLSCREEN
    }
    if(Build.VERSION.SDK_INT >= 21){
        setWindowFlag(WindowManager.LayoutParams.FLAG_TRANSLUCENT_STATUS, false)
        window.statusBarColor = Color.TRANSPARENT
    }
}
```

```
<?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"
    android:orientation="vertical"
    android:id="@+id/notes_list"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".NotesListActivity">

    <View
        android:layout_width="wrap_content"
        android:layout_height="160dp"
        android:background="@drawable/c_wave__2_"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintVertical_bias="0"
        android:elevation="-1dp"
        app:layout_constraintBottom_toBottomOf="@+id/listView"
        android:id="@+id/wave_upper"
    />

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="150dp"
        android:id="@+id/image_slider"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.1"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintBottom_toTopOf="@+id/wave_down"
    />

    <ListView
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        app:layout_constraintTop_toBottomOf="@+id/image_slider"
        app:layout_constraintBottom_toTopOf="@+id/bottom_navigation"
    />
```

```
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    android:divider="#00FFFFFF"
    android:background="#00FFFFFF"
    android:gravity="center"
    android:paddingRight="10dp"
    android:paddingLeft="10dp"
    android:dividerHeight="15dp"
    android:paddingTop="200dp"
    android:paddingBottom="65dp" >
```

```
</ListView>
```

```
<View
```

```
    android:layout_width="wrap_content"
    android:layout_height="140dp"
    android:background="@drawable/ic_wave_svg"
    app:layout_constraintBottom_toBottomOf="@+id/bottom_navigation"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="1"
    android:elevation="-1dp"
    android:layout_marginBottom="40dp"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    android:id="@+id/wave_down"
/>
```

```
<com.google.android.material.bottomnavigation.BottomNavigationView
    android:id="@+id/bottom_navigation"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:elevation="70dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:menu="@menu/bottom_menu" />
```

```
<com.google.android.material.floatingactionbutton.FloatingActionButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/floating_action_btn"
    android:tint="@color/white"
    android:src="@drawable/ic_baseline_add_24"
    android:layout_marginBottom="70dp"
    android:layout_marginEnd="2dp"
    android:backgroundTint="#DF62A4"
```

```
        android:enabled="true"
        android:clickable="true"
        android:elevation="60dp"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintBottom_toBottomOf="parent"
        android:focusable="true" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

NotesListActivity.kt:

```
package com.example.practical_11_20012532001

import android.app.AlarmManager
import android.app.PendingIntent
import android.content.Context
import android.content.DialogInterface
import android.content.Intent
import android.graphics.Color
import android.graphics.drawable.AnimationDrawable
import android.os.Build
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.LayoutInflater
import android.view.View
import android.view.WindowManager
import android.widget.*
import androidx.annotation.RequiresApi
import androidx.appcompat.app.AlertDialog
import com.google.android.material.bottomnavigation.BottomNavigationView
import com.google.android.material.floatingactionbutton.FloatingActionButton
import java.text.SimpleDateFormat
import java.util.*
import kotlin.collections.ArrayList

class NotesListActivity : AppCompatActivity() {
    companion object{
        lateinit var listView: ListView
        lateinit var arrayList: ArrayList<Notes>
        lateinit var adapter: NotesAdapter
    }

    fun updateStateOfListView(context: Context, dbNoteController: DBNoteController){
        arrayList = dbNoteController.getAllNotes()!!

        adapter = NotesAdapter(context, arrayList)

        listView.adapter = adapter
    }
}
```

```
@RequiresApi(Build.VERSION_CODES.M)
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_notes_list)

    setStatusBarTransparent()

    val slider = findViewById<ImageView>(R.id.image_slider)
    slider.setBackgroundResource(R.drawable.slider_animation)

    val animation: AnimationDrawable = slider.background as AnimationDrawable
    animation.start()

    val bottomMenu = findViewById<BottomNavigationView>(R.id.bottom_navigation)
    listView = findViewById(R.id.listView)
    val fButton = findViewById<FloatingActionButton>(R.id.floating_action_btn)

    bottomMenu.selectedItemId = R.id.notes

    val dbNoteController = DBNoteController(this)

    arrayList = dbNoteController.getAllNotes()!!

    adapter = NotesAdapter(this, arrayList)

    listView.adapter = adapter

    fButton.setOnClickListener {
        val boxView = LayoutInflater.from(this).inflate(R.layout.form_view, null)
        val alertDialog = AlertDialog.Builder(this).setView(boxView)

        val titleText = boxView.findViewById<TextView>(R.id.form_title)
        val subTitleText = boxView.findViewById<TextView>(R.id.form_sub_title)
        val descriptionText = boxView.findViewById<TextView>(R.id.form_description)
        val timePicker = boxView.findViewById<TimePicker>(R.id.form_time_picker)
        val switch = boxView.findViewById<Switch>(R.id.form_switch)

        alertDialog.apply {
            setTitle("Add Note here..")
            var isReminder = 0

            switch.setOnClickListener {
                isReminder = if(isReminder==0)1 else 0
            }
        }
    }
}
```

```
setPositiveButton("Add") { dialogInterface: DialogInterface, i: Int ->
    val alarmCalendar = Calendar.getInstance()
    val year: Int = alarmCalendar.get(Calendar.YEAR)
    val month: Int = alarmCalendar.get(Calendar.MONTH)
    val day: Int = alarmCalendar.get(Calendar.DATE)

    val time = SimpleDateFormat("MMM, dd yyyy hh:mm:ss a").format(alarmCalendar.time)

    var reminderTime = ""
    if(isReminder==1){
        alarmCalendar.set(year, month, day, timePicker.hour, timePicker.minute, 0)

        reminderTime = SimpleDateFormat("MMM, dd yyyy hh:mm:ss a").format(alarmCalendar.time)
    }

    val newItem =
        Notes(titleText.text.toString(), subTitleText.text.toString(), descriptionText.text.toString(), time, isReminder, reminderTime)
}

newItem.setId(Notes.randomID())
dbNoteController.addNote(newItem)

updateStateOfListView(this@NotesListActivity, dbNoteController)

if(isReminder==1) {
    setAlarm(alarmCalendar.timeInMillis, "Start", newItem.getId())
}
}

setNegativeButton("Cancel") { dialogInterface: DialogInterface, i: Int ->
    Toast.makeText(applicationContext, "Canceled", Toast.LENGTH_SHORT).show()
}
}.show()
```

```
bottomMenu.setOnItemSelectedListener{
    when(itemId){
        R.id.dashboard -> {
            Intent(this, DashBoardActivity::class.java).apply{
                startActivity(this)
            }
        }
        return@setOnItemSelectedListener true
    }
    R.id.notes -> {
        Intent(this, NotesListActivity::class.java).apply{
```

```
        startActivity(this)
    }
    return@setOnItemSelectedListener true
}
else -> {
    return@setOnItemSelectedListener true
}
}

private fun setAlarm(millisTime: Long, str: String, id: String) {
    val intent = Intent(this, MyReceiver::class.java)
    intent.putExtra("Service1", str)
    val intentService = Intent(this, AlarmService::class.java)
    intentService.putExtra("id", id)

    val pendingIntent = PendingIntent.getBroadcast(this@NotesListActivity, 234324243, intent, 0)
    val pendingIntentService = PendingIntent.getService(this@NotesListActivity, 234324244, intentService, 0)

    val alarmManager = getSystemService(ALARM_SERVICE) as AlarmManager

    if(str == "Start") {
        alarmManager.setExact(
            AlarmManager.RTC_WAKEUP,
            millisTime,
            pendingIntent
        )
        alarmManager.setExact(
            AlarmManager.RTC_WAKEUP,
            millisTime,
            pendingIntentService
        )
    } else if(str == "Stop") {
        alarmManager.cancel(pendingIntent)
    }
}

private fun setWindowFlag(flagTranslucentStatus: Int, b: Boolean) {
    val winParameters = window.attributes
    if(b){
        winParameters.flags = winParameters.flags or flagTranslucentStatus
    }
    else{

```

```
    winParameters.flags = winParameters.flags and flagTranslucentStatus.inv()
}
window.attributes = winParameters
}

private fun setStatusBarTransparent(){
    if (Build.VERSION.SDK_INT in 19..20) {
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT) {
            setWindowFlag(WindowManager.LayoutParams.FLAG_TRANSLUCENT_STATUS, true)
        }
    }
    if (Build.VERSION.SDK_INT >= 19) {
        window.decorView.systemUiVisibility = View.SYSTEM_UI_FLAG_LAYOUT_STABLE or
        View.SYSTEM_UI_FLAG_LAYOUT_FULLSCREEN
    }

    if (Build.VERSION.SDK_INT >= 21) {
        setWindowFlag(WindowManager.LayoutParams.FLAG_TRANSLUCENT_STATUS, false)
        window.statusBarColor = Color.TRANSPARENT
    }
}
```

activity_notification.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:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp"
    android:id="@+id/notification_activity"
    tools:context=".NotificationActivity">

    <androidx.cardview.widget.CardView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:backgroundTint="#1B1A1A"
        app:cardCornerRadius="10dp"
        >

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            >
```

```
    android:padding="15dp"
    >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/notificationtextViewTitle"
        android:textSize="20sp"
        android:text="title"
        android:textColor="@color/white"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="subtitle"
        android:textColor="#4A4444"
        android:id="@+id/notificationtextViewSubTitle"
        android:textSize="16sp"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/notificationtextViewDescription"
        android:textSize="10sp"
        android:text="description"
        android:textColor="#5C5353"/>
    />

    <TextView
        android:id="@+id/notificationtime"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Time"
        android:textColor="#5C5353"
        android:textSize="10sp"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textColor="#5C5353"
        android:text="Reminder Time"
        android:textSize="10sp"
        android:id="@+id/notificationreminder_time"/>
    />
</LinearLayout>

</androidx.cardview.widget.CardView>
```

```
package com.example.practical_11_20012532001

import android.os.Bundle
import android.view.MenuItem
import android.widget.ImageButton
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity

class NotificationActivity : AppCompatActivity() {
    private lateinit var title : TextView
    private lateinit var subTitle : TextView
    private lateinit var description : TextView
    private lateinit var reminderTime : TextView
    private lateinit var modifiedTime : TextView
    private var isReminder : Int = 0

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_notification)

        val note = (intent.extras?.getSerializable("note") as? Notes)!!

        this.title = findViewById(R.id.notificationTextViewTitle)
        this.subTitle = findViewById(R.id.notificationTextViewSubTitle)
        this.description = findViewById(R.id.notificationTextViewDescription)
        this.reminderTime = findViewById(R.id.notificationReminder_time)
        this.modifiedTime = findViewById(R.id.notificationTime)

        supportActionBar?.setDisplayHomeAsUpEnabled(true)
        supportActionBar?.title = "Reminder Of ${note.getTitle()}"
        this.title.text = note.getTitle()
        this.subTitle.text = note.getSubTitle()
        this.description.text = note.getDescription()
        this.reminderTime.text = note.getReminderTime()
        this.modifiedTime.text = note.getModifiedTime()
        this.isReminder = note.getIsReminder()
    }
}
```

```
override fun onContextItemSelected(item: MenuItem): Boolean {
    when (item.itemId) {
        R.id.notes_list-> {
            finish()
            return true
        }
    }
    return super.onContextItemSelected(item)
}
```

form_view.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:padding="20dp"
        >

        <com.google.android.material.textfield.TextInputLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginBottom="20dp"
            app:startIconDrawable="@drawable/ic_baseline_event_note_24">

            <com.google.android.material.textfield.TextInputEditText
                android:id="@+id/form_title"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginTop="20dp"
                android:hint="Note Title"
                />

        </com.google.android.material.textfield.TextInputLayout>

        <com.google.android.material.textfield.TextInputLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
```

```
        android:layout_marginBottom="20dp"
        app:startIconDrawable="@drawable/ic_baseline_event_note_24">

    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/form_sub_title"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:hint="Note Subtitle" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout>
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="20dp"
    app:startIconDrawable="@drawable/ic_baseline_event_note_24">

    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/form_description"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:hint="Note Description" />

</com.google.android.material.textfield.TextInputLayout>

<Switch
    android:layout_gravity="right"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Set Reminder"
    android:id="@+id/form_switch"
/>

<TimePicker
    android:id="@+id/form_time_picker"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"/>

</LinearLayout>
</ScrollView>
```

```
<?xml version='1.0' encoding='utf-8'?>
<LinearLayout
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_marginBottom="20dp"
    android:layout_height="wrap_content">

    <androidx.cardview.widget.CardView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:backgroundTint="#1B1A1A"
        app:cardCornerRadius="10dp"
        >

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:padding="15dp"
            >

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:id="@+id/textViewTitle"
                android:textSize="20sp"
                android:text="title"
                android:textColor="@color/white"/>
            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="subtitle"
                android:textColor="#4A4444"
                android:id="@+id/textViewSubTitle"
                android:textSize="16sp"/>
            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:id="@+id/textViewDescription"
                android:textSize="10sp"
                android:text="description"
                android:textColor="#5C5353"
                />
        <androidx.constraintlayout.widget.ConstraintLayout
            android:layout_marginTop="20dp"
```

```
        android:layout_marginBottom="10dp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
    >

    <TextView
        android:id="@+id/time"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Time"
        android:textColor="#5C5353"
        android:textSize="10sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toStartOf="@+id/delete_btn"
        app:layout_constraintHorizontal_bias="0.006"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"/>/>

    <ImageButton
        android:id="@+id/edit_btn"
        android:layout_width="40dp"
        android:layout_height="40dp"
        android:backgroundTint="#00FFFFFF"
        app:tint="#00FF00"
        android:clickable="true"
        android:src="@drawable/ic_baseline_edit_24"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toStartOf="@+id/delete_btn"
        app:layout_constraintHorizontal_bias="0.85"
        app:layout_constraintStart_toEndOf="@+id/time"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.0"/>/>

    <ImageButton
        android:id="@+id/delete_btn"
        android:layout_width="40dp"
        android:layout_height="40dp"
        android:backgroundTint="#00FFFFFF"
        android:layout_marginEnd="16dp"
        android:clickable="true"
        app:tint="#FF0000"
        android:src="@drawable/ic_baseline_delete_24"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.0"/>/>
```

```
    this.hours = hours
}
fun getMinutes():String{
    return this.minutes
}
fun setMinutes(minutes: String){
    this.minutes = minutes
}
fun getAMPM():String{
    return this.ampm
}
fun setAMPM(ampm:String){
    this.ampm = ampm
}
```

NotesAdapter.kt:

```
package com.example.practical_11_20012532001

import android.content.Context
import android.content.DialogInterface
import android.os.Build
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.*
import androidx.annotation.RequiresApi
import androidx.appcompat.app.AlertDialog
import java.text.SimpleDateFormat
import java.util.*

class NotesAdapter(private val context: Context, private val arrayList: ArrayList<Notes>): BaseAdapter() {

    private lateinit var title : TextView
    private lateinit var subTitle : TextView
    private lateinit var description : TextView
    private lateinit var reminderTime : TextView
    private lateinit var modifiedTime : TextView
    private var isReminder : Int = 0
    private lateinit var editBtn : ImageButton
    private lateinit var deleteBtn : ImageButton
    private val dbNoteController = DBNoteController(context)

    override fun getCount(): Int{
        return arrayList.size
    }
```

```
    override fun getItem(position: Int): Any {
        return position
    }

    override fun getItemId(position: Int): Long {
        return position.toLong()
    }

    @RequiresApi(Build.VERSION_CODES.M)
    override fun getView(position: Int, convertView: View?, parent: ViewGroup?): View {
        val convertView: View? = LayoutInflater.from(context).inflate(R.layout.item_box, parent, false)
        this.title = convertView.findViewById(R.id.textViewTitle)
        this subTitle = convertView.findViewById(R.id.textViewSubTitle)
        this.description = convertView.findViewById(R.id.textViewDescription)
        this.reminderTime = convertView.findViewById(R.id.reminder_time)
        this.modifiedTime = convertView.findViewById(R.id.time)
        this.editBtn = convertView.findViewById(R.id.edit_btn)
        this.deleteBtn = convertView.findViewById(R.id.delete_btn)

        val notId = arrayList[position].getId()
        val note = dbNoteController.getNote(notId)

        this.title.text = note.getTitle()
        this.subTitle.text = note.getSubTitle()
        this.description.text = note.getDescription()
        this.reminderTime.text = note.getReminderTime()
        this.modifiedTime.text = note.getModifiedTime()
        this.isReminder = note.getIsReminder()

        this.editBtn.setOnClickListener {

            val boxView = LayoutInflater.from(context).inflate(R.layout.form_view, null)
            val alertDialog = AlertDialog.Builder(context).setView(boxView)

            val titleText = boxView.findViewById<TextView>(R.id.form_title)
            val subTitleText = boxView.findViewById<TextView>(R.id.form_sub_title)
            val descriptionText = boxView.findViewById<TextView>(R.id.form_description)
            val timePicker = boxView.findViewById<TimePicker>(R.id.form_time_picker)
            var switch = boxView.findViewById<Switch>(R.id.form_switch)

            alertDialog.apply {
                setTitle("Edit Note here..")

                titleText.text = arrayList[position].getTitle()
                subTitleText.text = arrayList[position].getSubTitle()
                descriptionText.text = arrayList[position].getDescription()
                timePicker.setInitialHourOfDay(arrayList[position].getReminderTime().hour)
                timePicker.setInitialMinuteOfHour(arrayList[position].getReminderTime().minute)
                switch.isChecked = arrayList[position].getIsReminder()
            }
        }
    }
}
```

```
descriptionText.text = arrayList[position].getDescription()
timePicker.minute = arrayList[position].getMinutes().toInt()

switch.isChecked = isReminder==1

switch.setOnClickListener {
    isReminder = if(isReminder==0) 1 else 0
}

setPositiveButton("Update") { dialogInterface: DialogInterface, i: Int->
    val alarmCalendar = Calendar.getInstance()

    val year: Int = alarmCalendar.get(Calendar.YEAR)
    val month: Int = alarmCalendar.get(Calendar.MONTH)
    val day: Int = alarmCalendar.get(Calendar.DATE)
    val time = SimpleDateFormat("MMM, dd yyyy hh:mm:ss a").format(alarmCalendar.time)

    alarmCalendar.set(year,month,day,timePicker.hour,timePicker.minute)

    val note = Notes(
        titleText.text.toString(),
        subTitleText.text.toString(),
        descriptionText.text.toString(),
        time.toString(),
        isReminder,
        SimpleDateFormat("MMM, dd yyyy hh:mm:ss a").format(alarmCalendar.time),
    )
    note.setId(arrayList[position].getId())
    dbNoteController.updateNote(note)

    NotesListActivity.updateStateOfListView(context,dbNoteController)

    Toast.makeText(context,"Note Updated!",Toast.LENGTH_SHORT).show()
}

setNegativeButton("Cancel"){ dialogInterface: DialogInterface, i: Int->
    Toast.makeText(context,"Canceled",Toast.LENGTH_SHORT).show()
}
}.show()

this.deleteBtn.setOnClickListener {
    DBNoteController(context).deleteNote(arrayList[position].getId())
    NotesListActivity.updateStateOfListView(context, dbNoteController)
```

```
}

return convertView
}
```

MyReceiver.kt:

```
package com.example.practical_11_20012532001

import android.content.BroadcastReceiver
import android.content.Context
import android.content.Intent
import android.media.MediaPlayer

class MyReceiver : BroadcastReceiver() {
    var mp: MediaPlayer? = null
    override fun onReceive(context: Context?, intent: Intent?) {
        mp = MediaPlayer.create(context, R.raw.alarm)
        mp?.start()
    }
}
```

AlarmService.kt:

```
package com.example.practical_11_20012532001

import android.app.Service
import android.content.Intent
import android.media.MediaPlayer
import android.os.IBinder
import android.widget.Toast

class AlarmService : Service() {

    val PERSONAL_CHANNEL_ID = "PersonalChannel"

    override fun onStartCommand(intent: Intent?, flags: Int, startId: Int): Int {
        if (intent != null) {
            val id: String? = intent.getStringExtra("id")
            val dbNoteController = DBNoteController(applicationContext)
            Toast.makeText(applicationContext, id.toString(), Toast.LENGTH_SHORT).show()
        }
        return START_STICKY
    }
}
```

```
val data: Notes = dbNoteController.getNote(id!!)

    val notificationUtils = NotificationUtils(applicationContext)
    notificationUtils.createChannel(PERSONAL_CHANNEL_ID, "PERSONAL")
    notificationUtils.sendNotificationInDefaultChannel(
        data,
        101
    )
}

return START_STICKY
}

override fun onDestroy() {
    super.onDestroy()
}

override fun onBind(intent: Intent?): IBinder? {
    TODO("Not yet implemented")
}
```

LoginInfo.kt:

```
package com.example.practical.11_20012532001

import java.io.Serializable

class LoginInfo : Serializable {
    private var username: String = ""
    private var phoneno: Int = 0
    private var city: String = ""
    private var email: String = ""
    private var password: String = ""
    private var id: Int = -1

    constructor(username: String, phoneno: Int, city: String, email: String, password: String) {
        this.username = username
        this.phoneno = phoneno
        this.city = city
        this.email = email
        this.password = password
    }

    fun setId(id: Int) {
        this.id = id
    }

    fun getId(): Int {
        return id
    }
}
```

```
    return this.id
}

fun getUsername():String {
    return this.username
}

fun getPassword():String {
    return this.password
}

fun getCity():String{
    return this.city
}

fun getEmail():String {
    return this.email
}

fun getPhone():Int{
    return this.phoneno
}
}
```

NotificationUtils.kt:

```
package com.example.practical_11_20012532001
```

```
import android.R
import android.app.Notification
import android.app.NotificationChannel
import android.app.NotificationManager
import android.app.PendingIntent
import android.content.Context
import android.content.ContextWrapper
import android.content.Intent
import android.graphics.Color
import android.media.RingtoneManager
import android.os.Build
import androidx.core.app.NotificationCompat
```

```
class NotificationUtils(context: Context) : ContextWrapper(context) {
    private var notificationManager: NotificationManager? = null
    private get() {
        if (field == null) {
            field = getSystemService(NOTIFICATION_SERVICE) as NotificationManager
        }
        return field
    }
}
```

```
val COL_NOTE_ID = "note_id"
val COL_NOTE_ISREMINDER = "isreminder"

val note_columns = arrayOf(
    COL_NOTE_ID,
    COL_NOTE_TITLE,
    COL_NOTE_SUBTITLE,
    COL_NOTE_DESCRIPTION,
    COL_NOTE_TIME,
    COL_NOTE_ISREMINDER,
    COL_NOTE_Reminder_Time
)

val NOTE_TABLE = "note"

private val NOTE_TABLE_CREATE = (
    "CREATE TABLE " + NOTE_TABLE +
    "(" +
    COL_NOTE_ID + " TEXT PRIMARY KEY, " +
    COL_NOTE_TITLE + " TEXT NOT NULL, " +
    COL_NOTE_SUBTITLE + " TEXT NOT NULL, " +
    COL_NOTE_DESCRIPTION + " TEXT NOT NULL, " +
    COL_NOTE_TIME + " TEXT NOT NULL, " +
    COL_NOTE_ISREMINDER + " INTEGER, " +
    COL_NOTE_Reminder_Time + " TEXT" +
    ")"
)

private val NOTE_TABLE_DROP = "DROP TABLE $NOTE_TABLE"

/*Note Table*/
}

override fun onCreate(db: SQLiteDatabase?) {
    db?.execSQL(USER_TABLE_CREATE)
    db?.execSQL(NOTE_TABLE_CREATE)
}

override fun onUpgrade(db: SQLiteDatabase?, oldVersion: Int, newVersion: Int) {
    db?.execSQL(USER_TABLE_DROP)
    db?.execSQL(NOTE_TABLE_DROP)
    onCreate(db)
}
```

```
package com.example.practical_11_20012532001
```

```
import android.content.Context  
import android.database.sqlite.SQLiteDatabase  
import android.content.ContentValues  
import android.database.Cursor  
import android.widget.Toast
```

```
class DBNoteController(context: Context) {
```

```
    private var dbHelper: DBHelper  
    private lateinit var database: SQLiteDatabase  
    private var contextprivate = context
```

```
    init {  
        dbHelper = DBHelper(context)  
    }
```

```
    fun close() {  
        dbHelper.close()  
    }
```

```
    fun addNote(note: Notes){  
        database = dbHelper.writableDatabase  
  
        val values = ContentValues()  
  
        values.put(DBHelper.COL_NOTE_ID, note.getId())  
        values.put(DBHelper.COL_NOTE_TITLE, note.getTitle())  
        values.put(DBHelper.COL_NOTE_SUBTITLE, note.getSubTitle())  
        values.put(DBHelper.COL_NOTE_DESCRIPTION, note.getDescription())  
        values.put(DBHelper.COL_NOTE_TIME, note.getModifiedTime())  
        values.put(DBHelper.COL_NOTE_Reminder_TIME, note.getReminderTime())  
        values.put(DBHelper.COL_NOTE_ISREMINDER, note.getIsReminder())  
  
        database.insert(DBHelper.NOTE_TABLE, null, values)
```

```
        Toast.makeText(contextprivate, "Note Added!", Toast.LENGTH_SHORT).show()  
        database.close()  
    }
```

```
    fun getNote(id: String): Notes{  
        database = dbHelper.readableDatabase
```

```
// Adding Note to list
notesList.add(note)
} while (cursor.moveToNext())
}

// return Notes list
return notesList
}

// Updating single Note
fun updateNote(note: Notes): Int {
    val db = dbHelper.writableDatabase
    val values = ContentValues()
    values.put(DBHelper.COL_NOTE_TITLE, note.getTitle())
    values.put(DBHelper.COL_NOTE_SUBTITLE, note.getSubTitle())
    values.put(DBHelper.COL_NOTE_DESCRIPTION, note.getDescription())
    values.put(DBHelper.COL_NOTE_TIME, note.getModifiedTime())
    values.put(DBHelper.COL_NOTE_ISREMINDER, note.getIsReminder())
    values.put(DBHelper.COL_NOTE_Reminder_TIME, note.getReminderTime())

    // updating row
    return db.update(
        DBHelper.NOTE_TABLE, values,
        DBHelper.COL_NOTE_ID + " = ?",
        arrayOf(
            note.getId()
        )
    )
}

// Deleting single Note
fun deleteNote(note : Notes) {
    val db = dbHelper.writableDatabase
    db.delete(
        DBHelper.NOTE_TABLE,
        DBHelper.COL_NOTE_ID + " = ?",
        arrayOf(note.getId())
    )
    db.close()
}

// Deleting single Note by ID
fun deleteNote(_id: String) {
    val db = dbHelper.writableDatabase
    db.delete(
        DBHelper.NOTE_TABLE,
```

```
    android:pivotY="50%"  
    android:toXScale="2.0"  
    android:toYScale="2.0" />  
  
<scale  
    android:startOffset="4700"  
    android:fromXScale="1.0"  
    android:toXScale="0.5"  
    android:fromYScale="1.0"  
    android:toYScale="0.5"  
    android:duration="1500"  
    android:pivotX="50%"  
    android:pivotY="50%"  
/>  
</set>
```

Heart_animation.xml:

```
<?xml version="1.0" encoding="utf- 8"?>  
<animation-list  
    xmlns:android="http://schemas.android.com/apk/res/android">  
  
<item android:drawable="@drawable/heart1" android:duration="400" />  
<item android:drawable="@drawable/heart2" android:duration="400" />  
<item android:drawable="@drawable/heart3" android:duration="400" />  
<item android:drawable="@drawable/heart4" android:duration="400" />  
<item android:drawable="@drawable/heart5" android:duration="400" />  
  
</animation-list>
```

Slider_animation.xml:

```
<?xml version="1.0" encoding="utf- 8"?>  
<animation-list  
    xmlns:android="http://schemas.android.com/apk/res/android">  
    <item android:drawable="@drawable/slider4" android:duration="3000" />  
    <item android:drawable="@drawable/slider1" android:duration="3000" />  
    <item android:drawable="@drawable/slider2" android:duration="3000" />  
    <item android:drawable="@drawable/slider3" android:duration="3000" />  
    <item android:drawable="@drawable/slider5" android:duration="3000" />  
</animation-list>
```

Splash_logo_animation.xml:

```
<?xml version="1.0" encoding="utf- 8"?>  
<animation-list android:oneshot="true"  
    xmlns:android="http://schemas.android.com/apk/res/android">  
    <item android:drawable="@drawable/uvpce_logo_1" android:duration="400" />  
    <item android:drawable="@drawable/uvpce_logo_2" android:duration="400" />  
    <item android:drawable="@drawable/uvpce_logo_3" android:duration="400" />
```

```
<item name="colorOnSecondary">@color/black</item>
<!-- Status bar color.-->
<item name="android:statusBarColor" tools:targetApi="l">?attr/colorPrimaryVariant</item>
<!-- Customize your theme here.-->
</style>

<style name="Theme.Practical_11_20012022003.ActionBar"
parent="Theme.MaterialComponents.DayNight.DarkActionBar">
<!-- Primary brand color.-->
<item name="colorPrimary">@color/purple_500</item>
<item name="colorPrimaryVariant">@color/purple_700</item>
<item name="colorOnPrimary">@color/white</item>
<!-- Secondary brand color.-->
<item name="colorSecondary">@color/teal_200</item>
<item name="colorSecondaryVariant">@color/teal_700</item>
<item name="colorOnSecondary">@color/black</item>
<!-- Status bar color.-->
<item name="android:statusBarColor" tools:targetApi="l">?attr/colorPrimaryVariant</item>
<!-- Customize your theme here.-->
</style>

</resources>
```

AndroidManifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.practical_11_20012532001">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        >
        <activity
            android:name=".SplashActivity"
            android:exported="true"
            android:theme="@style/Theme.Practical_11_20012532001"
            >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```



