

ANDROID PROGRAMMING

PRACTICAL FILE

**Submitted By:**

Name: Madhubanti Mitra

College Roll No: 19/78015

Course: BSc(H) Computer Science

Sem: IV

Submitted To:

Mr. Lokesh Kumar Shrivastav

(Department of Computer Science)

INDEX

Serial No.	Practical
7	Create a radio button group with radio button of all courses in your college and on selecting a particular course, teacher-in- charge of that course should appear at the bottom of the screen.
8	Create a list of all courses in your college and on selecting a particular course parent department and the teacher-in-charge of that department should appear at the bottom of the screen.
9	a)Create an application with three buttons (with different color names) vertically aligned, on selecting a button color of the screen will change. b)Create an application with three buttons horizontally aligned, on selecting a button color of the screen will change.
10	Create a Login application (check username and password). On successful login, pop up the message. ("Welcome username")
11	Create a login application as above, on successful login redirect to another activity with logout button. On click of logout button a dialog appears with OK and CANCEL button. On OK button click go to login activity and on CANCEL stay at same activity.
12	Create an application to Create, Insert, update and Delete operation on the database.

Objective of Practical 7:

Create a radio button group with radio button of all courses in your college and on selecting a particular course, teacher-in-charge of that course should appear at the bottom of the screen.

Code:

activity_main.xml

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

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="75dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" >

        <RadioGroup
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/rgCourses"
            android:layout_centerInParent="true">
        </RadioGroup>
    </RelativeLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

```
package com.example.courses;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.provider.MediaStore;
import android.widget.RadioGroup;
import android.widget.RadioButton;
import com.google.android.material.snackbar.Snackbar;
import java.util.HashMap;

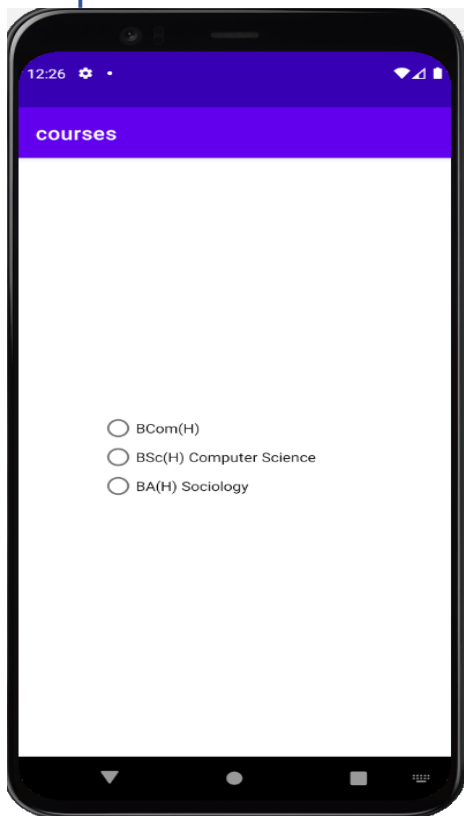
public class MainActivity extends AppCompatActivity {
    private RadioGroup rgCourses;
    private HashMap<String, String> coursesMap;
    @Override
```

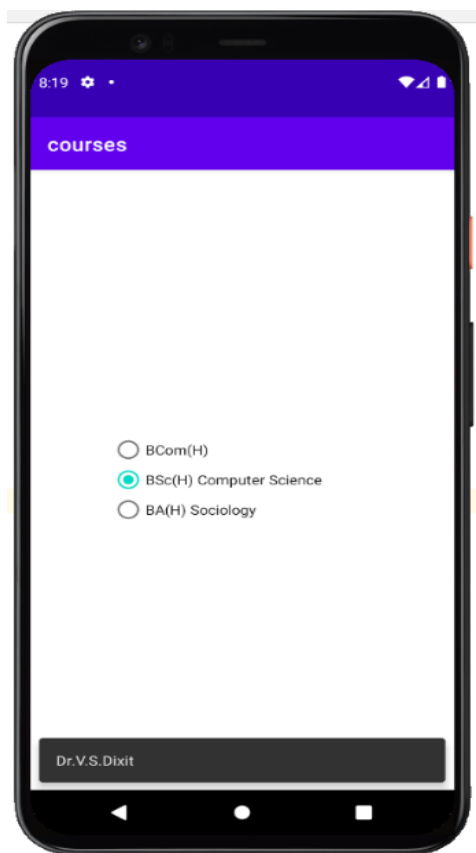
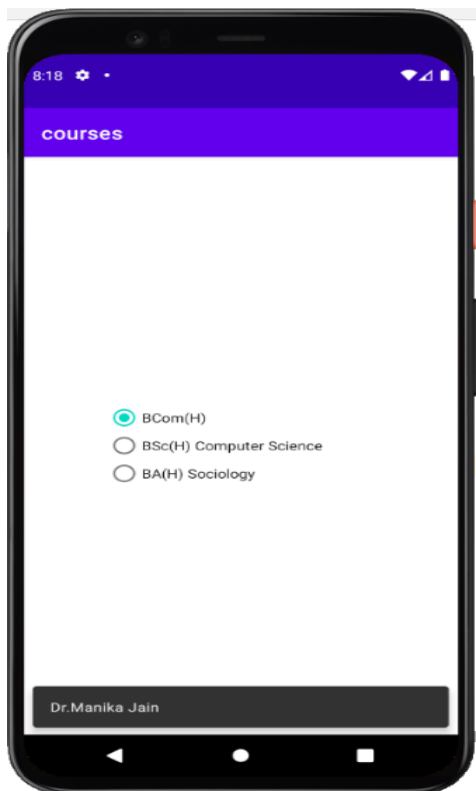
```

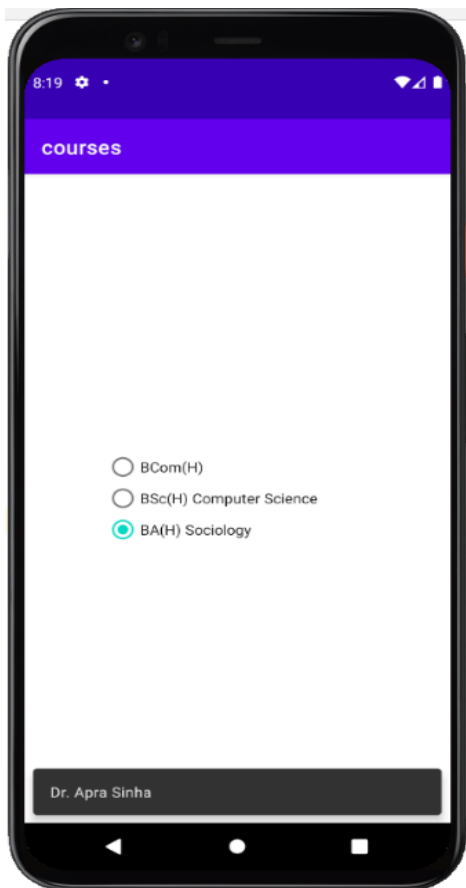
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    initCourses();
}
private void initCourses(){
    rgCourses=(RadioGroup)findViewById(R.id.rgCourses);
    if (coursesMap==null){
        coursesMap=new HashMap<>();
        coursesMap.put("BSc(H) Computer Science", " Dr.V.S.Dixit");
        coursesMap.put("BCom(H)", " Dr.Manika Jain");
        coursesMap.put("BA(H) Sociology", " Dr. Apra Sinha");
        for (String course : coursesMap.keySet()){
            RadioButton rButton=new RadioButton(MainActivity.this);
            rButton.setText(course);
            rgCourses.addView(rButton);
        }
        rgCourses.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener() {
            @Override
            public void onCheckedChanged(RadioGroup group, int checkedId) {
                int checkedRadioButtonId=rgCourses.getCheckedRadioButtonId();
                RadioButton rButton=(RadioButton) findViewById(checkedRadioButtonId);
                Snackbar.make(rgCourses,coursesMap.get(rButton.getText()),Snackbar.LENGTH_SHORT)
                    .show();
            }
        });
    }
}

```

Output:







Objective of Practical 8:

Create a list of all courses in your college and on selecting a particular course parent department and the teacher-in-charge of that department should appear at the bottom of the screen.

Code:

activity_main.xml

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

```
<RelativeLayout
```

```

        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="75dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" >

        <RadioGroup
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/rgCourses"
            android:layout_centerInParent="true">
        </RadioGroup>
    </RelativeLayout>

</androidx.constraintlayout.widget.ConstraintLayout>

```

MainActivity.java

```

package com.example.incharge;

import androidx.appcompat.app.AppCompatActivity;

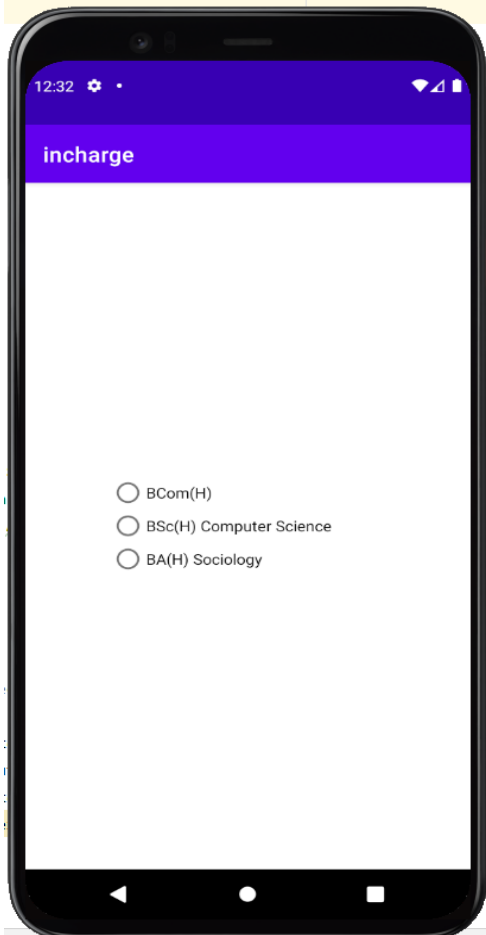
import android.os.Bundle;
import android.provider.MediaStore;
import android.widget.RadioGroup;
import android.widget.RadioButton;
import com.google.android.material.snackbar.Snackbar;
import java.util.HashMap;

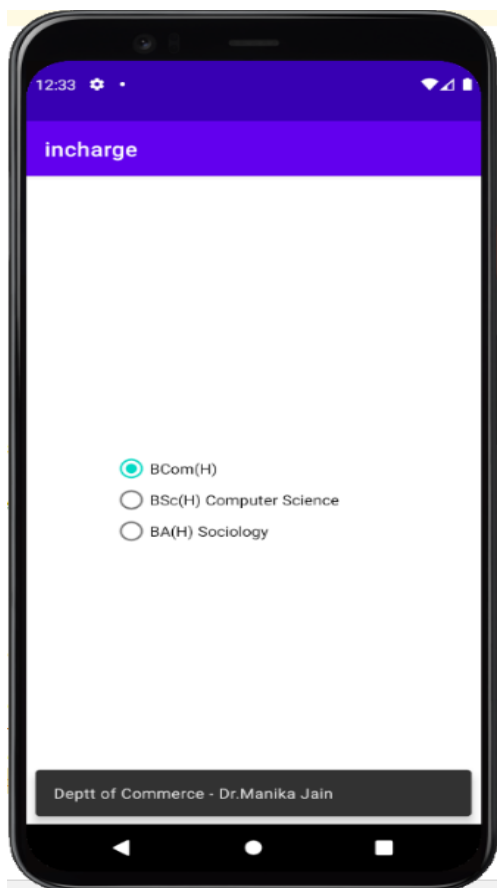
public class MainActivity extends AppCompatActivity {
    private RadioGroup rgCourses;
    private HashMap<String, String> coursesMap;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        initCourses();
    }
    private void initCourses(){
        rgCourses=(RadioGroup)findViewById(R.id.rgCourses);
        if (coursesMap==null){
            coursesMap=new HashMap<>();
            coursesMap.put("BSc(H) Computer Science","Deptt of Computer Sc - Dr.V.S.Dixit");
            coursesMap.put("BCom(H)","Deptt of Commerce - Dr.Manika Jain");
            coursesMap.put("BA(H) Sociology"," Deptt of Economics - Dr. Apra Sinha");
            for (String course : coursesMap.keySet()){
                RadioButton rButton=new RadioButton(MainActivity.this);
                rButton.setText(course);
                rgCourses.addView(rButton);
            }
            rgCourses.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener() {
                @Override
                public void onCheckedChanged(RadioGroup group, int checkedId) {
                    int checkedRadioButtonId=rgCourses.getCheckedRadioButtonId();
                    RadioButton rButton=(RadioButton) findViewById(checkedRadioButtonId);

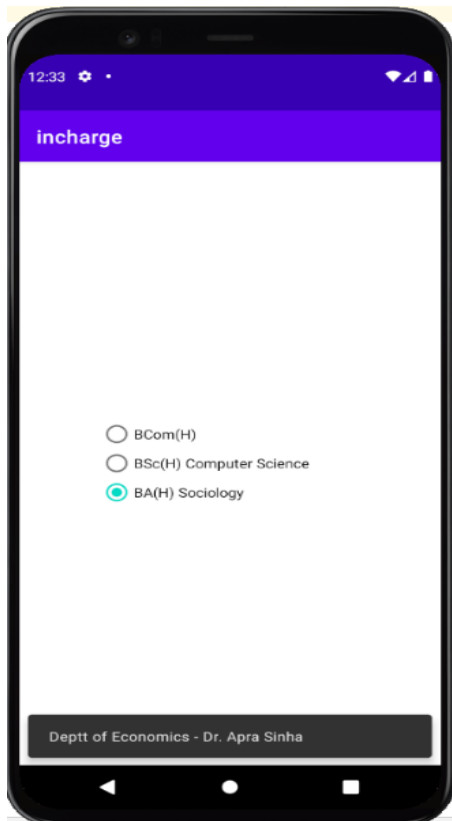
```

```
Snackbar.make(rgCourses,coursesMap.get(rButton.getText()),Snackbar.LENGTH_SHORT)  
        .show();  
    }  
});  
}  
}
```

Output:







Objective of Practical 9(a):

a) Create an application with three buttons (with different color names) vertically aligned, on selecting a button color of the screen will change.

Code:

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/layoutSuperParent"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent">
```

```

<Button
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:id="@+id/btnRed"
    android:layout_gravity="center"
    android:backgroundTint="@android:color/holo_red_dark"
    android:text="RED"/>
<Button
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:id="@+id/btnBlue"
    android:layout_gravity="center"
    android:backgroundTint="@android:color/holo_blue_light"
    android:text="BLUE"/>
<Button
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:id="@+id/btnGreen"
    android:gravity="center"
    android:backgroundTint="@android:color/holo_green_dark"
    android:text="GREEN"/>
<Button
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:id="@+id/btnReset"
    android:gravity="center"
    android:text="Reset"/>
</LinearLayout>

</androidx.constraintlayout.widget.ConstraintLayout>

```

MainActivity.java

```

package com.example.colorchangevertical;

import androidx.appcompat.app.AppCompatActivity;
import android.graphics.Color;
import android.view.View;
import android.widget.Button;
import androidx.constraintlayout.widget.ConstraintLayout;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity implements View.OnClickListener {
    private Button btnRed;
    private Button btnBlue;
    private Button btnGreen;
    private Button btnReset;
    private ConstraintLayout layoutSuperParent;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        layoutSuperParent=(ConstraintLayout)findViewById(R.id.layoutSuperParent);
        btnRed=(Button)findViewById(R.id.btnRed);
        btnBlue=(Button)findViewById(R.id.btnBlue);
        btnGreen=(Button)findViewById(R.id.btnGreen);
        btnReset=(Button)findViewById(R.id.btnReset);
        btnRed.setOnClickListener(this);
    }
}

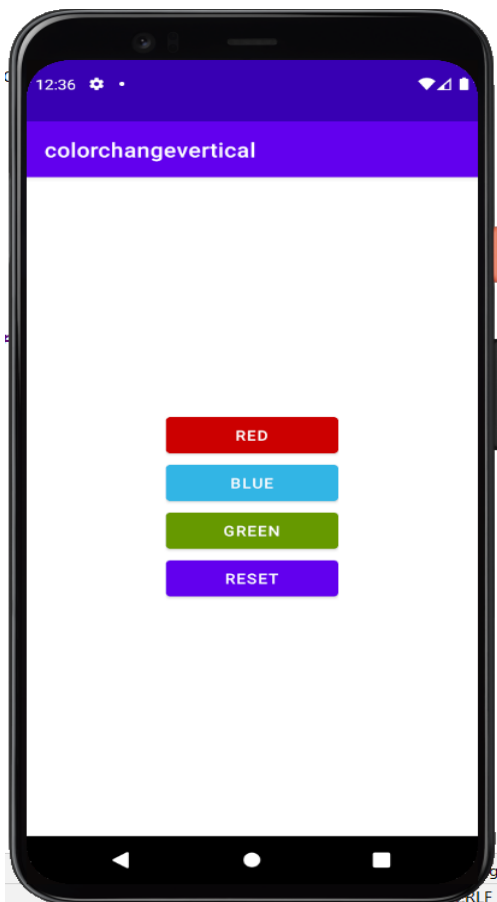
```

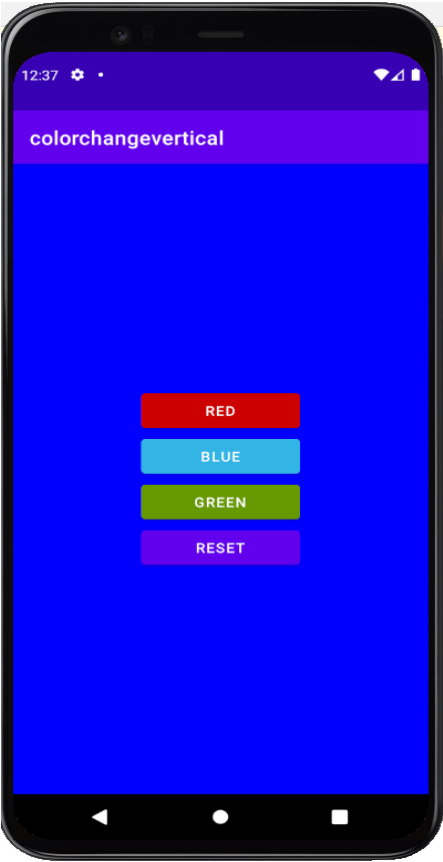
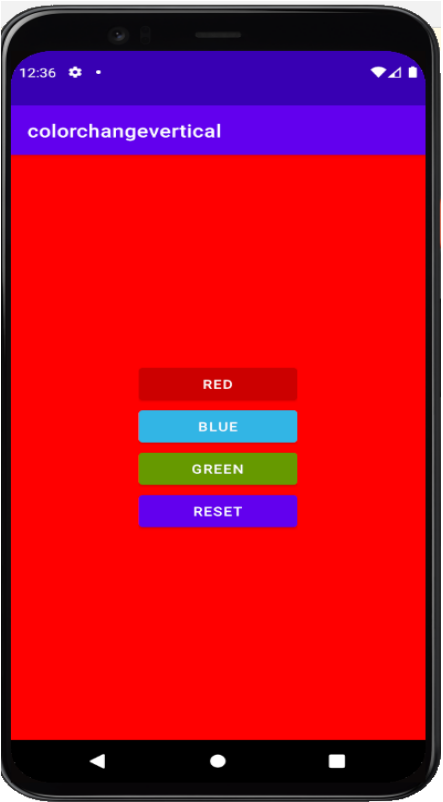
```

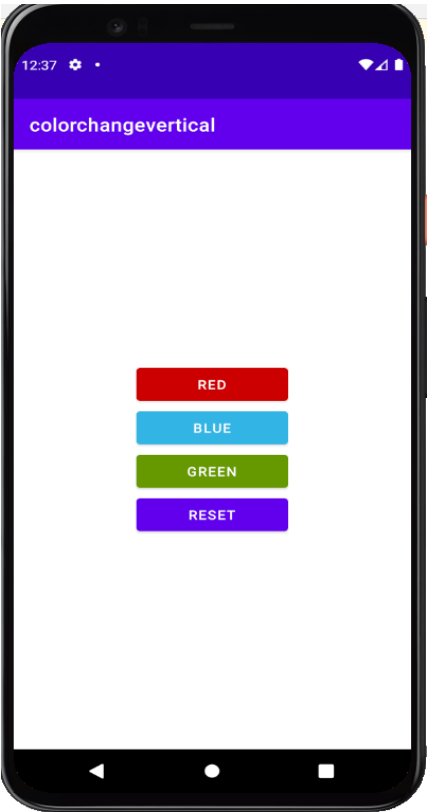
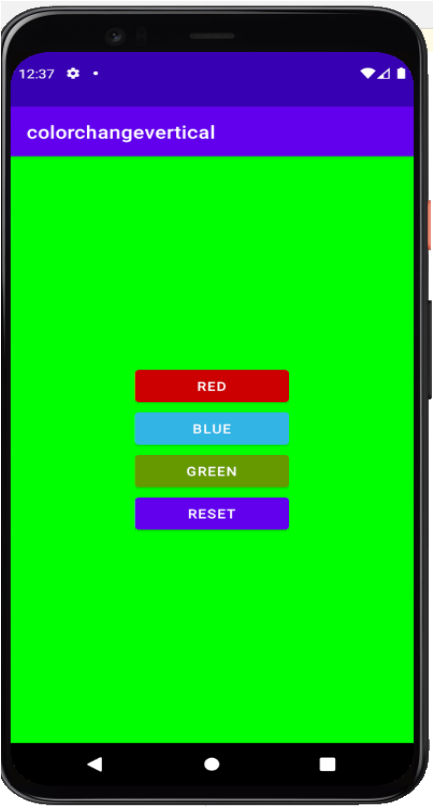
        btnBlue.setOnClickListener(this);
        btnGreen.setOnClickListener(this);
        btnReset.setOnClickListener(this);
    }
    @Override
    public void onClick(View v){
        switch (v.getId()){
            case R.id.btnRed:
                layoutSuperParent.setBackgroundColor(Color.RED);
                break;
            case R.id.btnBlue:
                layoutSuperParent.setBackgroundColor(Color.BLUE);
                break;
            case R.id.btnGreen:
                layoutSuperParent.setBackgroundColor(Color.GREEN);
                break;
            case R.id.btnReset:
                layoutSuperParent.setBackgroundColor(Color.TRANSPARENT);
                break;
            default:
                break;
        }
    }
}

```

Output:







Objective of Practical 9(b):

b) Create an application with three buttons horizontally aligned, on selecting a button color of the screen will change.

Code:

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/layoutSuperParent"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent">
    <Button
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:id="@+id/btnRed"
        android:layout_gravity="center"
        android:backgroundTint="@android:color/holo_red_dark"
        android:text="RED"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:id="@+id/btnBlue"
        android:layout_gravity="center"
        android:backgroundTint="@android:color/holo_blue_light"
        android:text="BLUE"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:id="@+id/btnGreen"
        android:gravity="center"
        android:backgroundTint="@android:color/holo_green_dark"
        android:text="GREEN"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:id="@+id/btnReset"
        android:gravity="center"
        android:text="Reset"/>
</LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

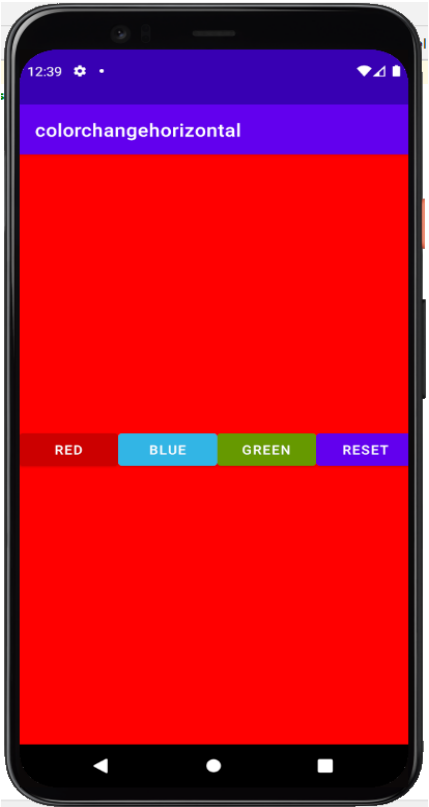
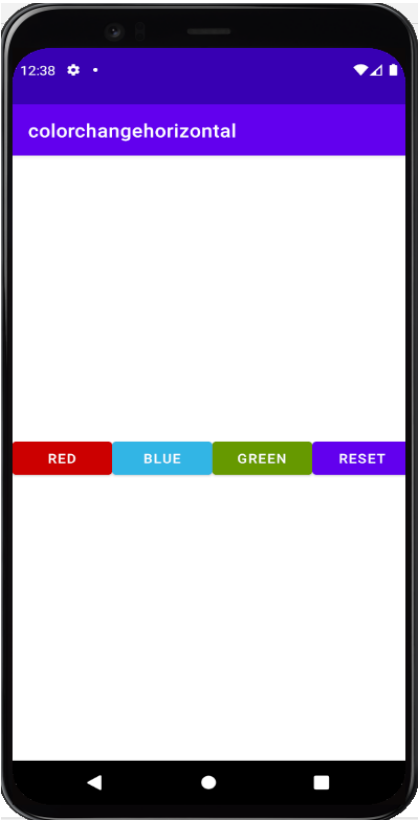
MainActivity.java

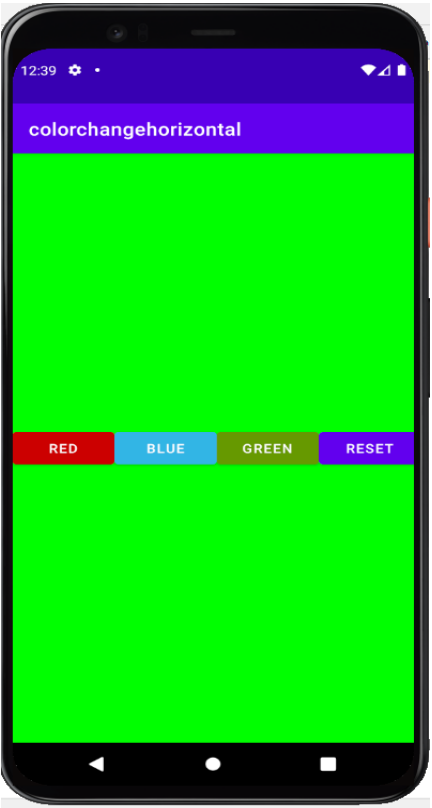
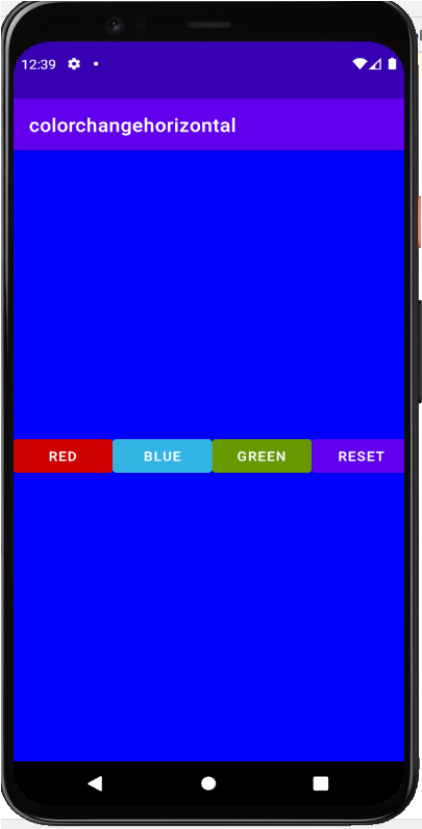
```
package com.example.colorchangehorizontal;

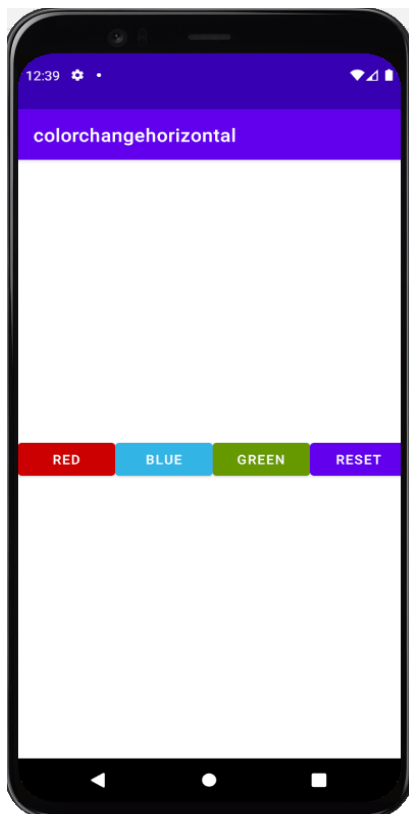
import androidx.appcompat.app.AppCompatActivity;
import android.graphics.Color;
import android.view.View;
import android.widget.Button;
import androidx.constraintlayout.widget.ConstraintLayout;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity implements View.OnClickListener {
    private Button btnRed;
    private Button btnBlue;
    private Button btnGreen;
    private Button btnReset;
    private ConstraintLayout layoutSuperParent;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        layoutSuperParent=(ConstraintLayout)findViewById(R.id.layoutSuperParent);
        btnRed=(Button)findViewById(R.id.btnRed);
        btnBlue=(Button)findViewById(R.id.btnBlue);
        btnGreen=(Button)findViewById(R.id.btnGreen);
        btnReset=(Button)findViewById(R.id.btnReset);
        btnRed.setOnClickListener(this);
        btnBlue.setOnClickListener(this);
        btnGreen.setOnClickListener(this);
        btnReset.setOnClickListener(this);
    }
    @Override
    public void onClick(View v){
        switch (v.getId()){
            case R.id.btnRed:
                layoutSuperParent.setBackgroundColor(Color.RED);
                break;
            case R.id.btnBlue:
                layoutSuperParent.setBackgroundColor(Color.BLUE);
                break;
            case R.id.btnGreen:
                layoutSuperParent.setBackgroundColor(Color.GREEN);
                break;
            case R.id.btnReset:
                layoutSuperParent.setBackgroundColor(Color.TRANSPARENT);
                break;
            default:
                break;
        }
    }
}
```


Output:







Objective of Practical 10:

Create a Login application (check username and password). On successful login, pop up the message. ("Welcome username")

Code:

app/src/main/res/layout/activity_main.xml

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

    <EditText
        android:id="@+id/etEmail"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="240dp"
        android:autofillHints="false"
        android:ems="10"
        android:hint="@string/e_mail_id"
```

```

        android:id:inputType="textEmailAddress"
        app:layout_constraintBottom_toTopOf="@id/etPassword"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

```

<EditText

```

        android:id="@+id/etPassword"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:autoFillHints="false"
        android:ems="10"
        android:hint="@string/password"
        android:inputType="textPassword"
        app:layout_constraintBottom_toTopOf="@id/btnLogin"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/etEmail" />

```

<Button

```

        android:id="@+id/btnLogin"
        android:layout_width="wrap_content"

        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="@string/login"
        app:layout_constraintBottom_toTopOf="@id/btnRegister"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/etPassword" />

```

<Button

```

        android:id="@+id/btnRegister"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="250dp"
        android:text="@string/register"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/btnLogin" />

```

</androidx.constraintlayout.widget.ConstraintLayout>

app/src/main/res/values/strings.xml

<resources>

```

    <string name="app_name">LoginValidation</string>
    <string name="welcome">Welcome</string>
    <string name="log_out">Log Out</string>
    <string name="password">Password</string>
    <string name="email_id">E-mail ID</string>
    <string name="login">Login</string>
    <string name="register">Register</string>
    <string name="cancel">Cancel</string>
    <string name="ok">Ok</string>
</resources>

```

app/src/main/java/com/example/loginvalidation/MainActivity.java

```

package com.example.loginvalidation;

import android.database.sqlite.SQLiteConstraintException;
import android.os.Bundle;
import android.util.Log;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import com.example.loginvalidation.models.User;
import com.example.loginvalidation.services.UserService;

public class MainActivity extends AppCompatActivity {
    private EditText etEmail;
    private EditText etPassword;
    private Button btnLogin;
    private Button btnRegister;
    private UserService userService;

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

        userService = UserService.getInstance();
        UserService.initDatabase(this);

        etEmail = (EditText) findViewById(R.id.etEmail);
        etPassword = (EditText) findViewById(R.id.etPassword);
        btnLogin = (Button) findViewById(R.id.btnLogin);
        btnRegister = (Button) findViewById(R.id.btnRegister);

        btnLogin.setOnClickListener(v -> {
            User user = validateCredentials();

```

```

        if (user != null) {
            try {
                userService.login(user);
                Toast.makeText(this, "Welcome " + userService.getUser().email, Toast.LENGTH_SHORT).show();
            } catch (IllegalStateException illegalStateException) {
                Toast.makeText(this, "Invalid E-mail/Password", Toast.LENGTH_SHORT).show();
            } catch (Exception e) {
                Log.d("TAG", e.getMessage());
                Toast.makeText(this, "Error", Toast.LENGTH_SHORT).show();
            }
        }
    });

    btnRegister.setOnClickListener(v -> { User user
        = validateCredentials(); if (user != null) {
            try {
                userService.register(user);
                Toast.makeText(this, "Registered", Toast.LENGTH_SHORT).show();
            } catch (SQLiteConstraintException sqlLiteConstraintException)
            {
                Toast.makeText(this, "E-mail Already Used", Toast.LENGTH_SHORT).show();
            } catch (Exception e) {
                Toast.makeText(this, "Error", Toast.LENGTH_SHORT).show();
            }
        }
    });
}

private User validateCredentials() {
    String email = etEmail.getText().toString().trim();
    String password = etPassword.getText().toString().trim();

    if (email.length() == 0) {
        etEmail.setError("Enter your e-mail address");
        return null;
    }

    if (password.length() == 0) {
        etPassword.setError("Enter your password");
        return null;
    }

    return new User(email, password);
}
}

```

app/src/main/java/com/example/loginvalidation/models/User.java

```
package com.example.loginvalidation.models;
```

```
import androidx.annotation.NonNull;
```

```
import androidx.room.ColumnInfo;
```

```
import androidx.room.Entity;
```

```
import androidx.room.PrimaryKey;
```

```
@Entity
```

```
public class User {
```

```
    @PrimaryKey
```

```
    @NonNull
```

```
    @ColumnInfo(name = "email")
```

```
    public String email;
```

```
    @NonNull
```

```
    @ColumnInfo(name = "password")
```

```
    public String password;
```

```
    public User(@NonNull String email, @NonNull String password) {
```

```
        this.email = email;
```

```
        this.password = password;
```

```
    }
```

```
}
```

app/src/main/java/com/example/loginvalidation/data/UserDao.java

```
package com.example.loginvalidation.data;
```

```
import androidx.room.Dao;
```

```
import androidx.room.Delete;
```

```
import androidx.room.Insert;
```

```
import androidx.room.Query;
```

```
import java.util.List;
```

```
import
```

```
com.example.loginvalidation.models.User;
```

```
@Dao
```

```
public interface UserDao {
```

```
    @Query("SELECT * FROM user")
```

```
    List<User> getAll();
```

```
    @Query("SELECT * FROM user WHERE email LIKE :email LIMIT 1")
```

```
    User findByEmail(String email);
```

```

    @Insert
    void insert(User user);

    @Delete
    void delete(User user);
}

```

```

app/src/main/java/com/example/loginvalidation/data/UserDatabase.java
package com.example.loginvalidation.data;

```

```

import androidx.room.RoomDatabase;
import androidx.room.Database;

```

```

import com.example.loginvalidation.models.User;

```

```

@Database(entities = {User.class}, version = 1)
public abstract class UserDatabase extends RoomDatabase {
    public abstract UserDao userDao();
}

```

```

app/src/main/java/com/example/loginvalidation/services/UserService.java
package com.example.loginvalidation.services;
import android.content.Context;

```

```

import androidx.room.Room;

```

```

import com.example.loginvalidation.data.UserDatabase;
import com.example.loginvalidation.models.User;

```

```

public class UserService {
    private User user = null;
    private static UserService instance = null;
    private static UserDatabase database = null;

    public void login(User user) throws IllegalStateException {
        User u = database.userDao().findByEmail(user.email);
        if (!u.password.equals(user.password))
            throw new IllegalStateException("Password Mismatch");
        this.user = user;
    }

    public void register(User user) {
        database.userDao().insert(user);
        this.user = user;
    }

    public void logout() {
        this.user = null;
    }
}

```



```

    }

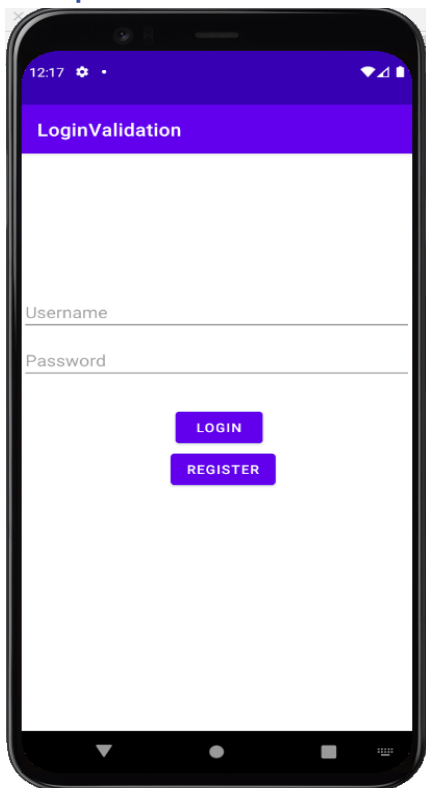
    public User getUser() {
        return user;
    }

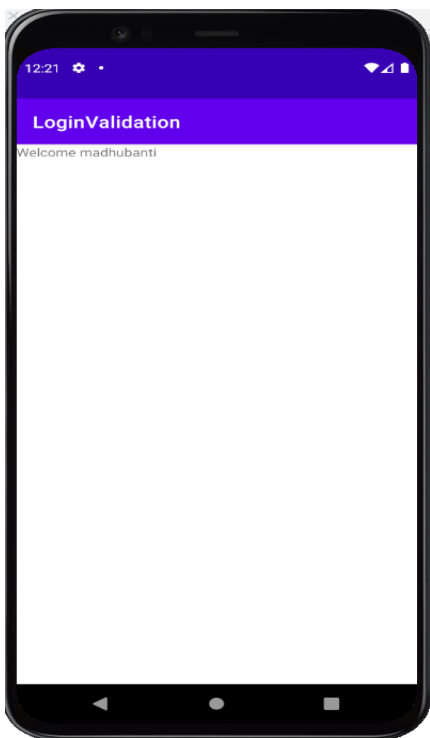
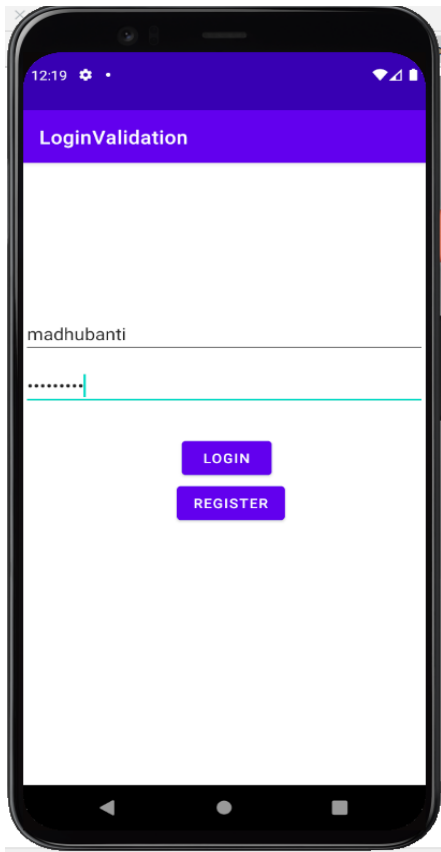
    public static UserService getInstance() {
        if (instance == null) {
            instance = new UserService();
        }
        return instance;
    }

    public static void initDatabase(Context ctx) {
        if (database == null) {
            database = Room
                .databaseBuilder(ctx, UserDatabase.class, "login_module")
                .allowMainThreadQueries()
                .build();
        }
    }
}

```

Output:





Objective of Practical 11:

Create a login application as above, on successful login redirect to another activity with logout button. On click of logout button a dialog appears with OK and CANCEL button. On OK button click go to login activity and on CANCEL stay at same activity.

Code:

app/src/main/AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.loginmodule">
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/Theme.LoginModule">
        <activity android:name=".MemberModule"></activity>
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

app/src/main/res/layout/activity_main.xml

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

    <EditText
        android:id="@+id/etEmail"
        android:layout_width="wrap_content"
```

```

        android:layout_height="wrap_content"
        android:layout_marginTop="240dp"
        android:autoFillHints="false"
        android:ems="10"
        android:hint="@string/e_mail_id"
        android:inputType="textEmailAddress"
        app:layout_constraintBottom_toTopOf="@id/etPassword"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

```

<EditText

```

        android:id="@+id/etPassword"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:autoFillHints="false"
        android:ems="10"
        android:hint="@string/password"
        android:inputType="textPassword"
        app:layout_constraintBottom_toTopOf="@id/btnLogin"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/etEmail" />

```

<Button

```

        android:id="@+id/btnLogin"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="@string/login"
        app:layout_constraintBottom_toTopOf="@id/btnRegister"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/etPassword" />

```

<Button

```

        android:id="@+id/btnRegister"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="250dp"
        android:text="@string/register"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/btnLogin" />

```

</androidx.constraintlayout.widget.ConstraintLayout>

app/src/main/res/layout/activity_member_module.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"
    tools:context=".MemberModule">

    <TextView
        android:id="@+id/tvWelcome"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/welcome"
        android:textAppearance="@style/TextAppearance.AppCompat.Display1"
        app:layout_constraintBottom_toTopOf="@id/tvEmail"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/tvEmail"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=""
        app:layout_constraintBottom_toTopOf="@id/btnLogout"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/tvWelcome" />

    <Button
        android:id="@+id/btnLogout"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/log_out"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/tvEmail" />

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

app/src/main/res/values/strings.xml

```
<resources>
    <string name="app_name">LoginModule</string>
    <string name="welcome">Welcome</string>
```

```

    <string name="log_out">Log Out</string>
    <string name="password">Password</string>
    <string name="email_id">E-mail ID</string>
    <string name="login">Login</string>
    <string name="register">Register</string>
    <string name="cancel">Cancel</string>
    <string name="ok">Ok</string>
</resources>

```

```

app/src/main/java/com/example/loginmodule/MainActivity.java
package com.example.loginmodule;

```

```

import android.content.Intent;
import android.database.sqlite.SQLiteConstraintException;
import android.os.Bundle;
import android.util.Log;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

```

```

import androidx.appcompat.app.AppCompatActivity;

```

```

import com.example.loginmodule.models.User;
import com.example.loginmodule.services.UserService;

```

```

public class MainActivity extends AppCompatActivity {
    private EditText etEmail;
    private EditText etPassword;
    private Button btnLogin;
    private Button btnRegister;
    private UserService userService;

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

        userService = UserService.getInstance();
        UserService.initDatabase(this);

        etEmail = (EditText) findViewById(R.id.etEmail);
        etPassword = (EditText) findViewById(R.id.etPassword);
        btnLogin = (Button) findViewById(R.id.btnLogin);
        btnRegister = (Button) findViewById(R.id.btnRegister);

        btnLogin.setOnClickListener(v -> {
            User user = validateCredentials();
            if (user != null){

```

```

        try {
            userService.login(user);
            Toast.makeText(this, "Logged In", Toast.LENGTH_SHORT).show();
            startActivity(new Intent(this, MemberModule.class));
        } catch (IllegalStateException illegalStateException) { Toast.makeText(this,
"Invalid E- mail/Password", Toast.LENGTH_SHORT).show();
        } catch (Exception e) {
            Log.d("TAG", e.getMessage());
            Toast.makeText(this, "Error", Toast.LENGTH_SHORT).show();}
    }
}); btnRegister.setOnClickListener(v -> { User
    user = validateCredentials(); if (user !=
    null) {
        try {
            userService.register(user);
            Toast.makeText(this, "Registered", Toast.LENGTH_SHORT).show();
            startActivity(new Intent(this, MemberModule.class));
        } catch (SQLiteConstraintException sqLiteConstraintException)
        {
            Toast.makeText(this, "E-mail Already Used",
            Toast.LENGTH_SHORT).show();
            } catch (Exception e) {
                Toast.makeText(this, "Error", Toast.LENGTH_SHORT).show();
            }
        }
    }
});
}
private User validateCredentials() {
    String email = etEmail.getText().toString().trim();
    String password = etPassword.getText().toString().trim();

    if (email.length() == 0) {
        etEmail.setError("Enter your e-mail address");
        return null;
    }

    if (password.length() == 0) {
        etPassword.setError("Enter your password");
        return null;
    }

    return new User(email, password);
}
}

```

app/src/main/java/com/example/loginmodule/MemberModule.java
package com.example.loginmodule;

import android.app.AlertDialog;

```

import android.os.Bundle;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import com.example.loginmodule.models.User;
import com.example.loginmodule.services.UserService;

public class MemberModule extends AppCompatActivity {
    private User user;
    private UserService userService;
    private TextView tvEmail;
    private TextView btnLogout;

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

        userService = UserService.getInstance();
        user = userService.getUser();

        tvEmail = (TextView) findViewById(R.id.tvEmail);
        btnLogout = (TextView) findViewById(R.id.btnLogout);

        if (user == null) {
            super.finish();
        } else {
            tvEmail.setText(user.email);
        }

        btnLogout.setOnClickListener(v -> {
            new AlertDialog.Builder(this)
                .setTitle("Logout")
                .setMessage("Are you sure you want to log out?")
                .setPositiveButton(android.R.string.ok, (dialog, which) -
> logoutUser())
                .setNegativeButton(android.R.string.cancel, null)
                .setIcon(android.R.drawable.ic_dialog_alert)
                .show();
        });
    }

    private void logoutUser() {
        userService.logout();
        Toast.makeText(this, "Logged Out", Toast.LENGTH_SHORT).show();
        super.finish();
    }
}

```



```

    }
}

```

app/src/main/java/com/example/loginmodule/models/User.java

```

package com.example.loginmodule.models;

import androidx.annotation.NonNull;
import androidx.room.ColumnInfo;
import androidx.room.Entity;
import androidx.room.PrimaryKey;

@Entity
public class User {
    @PrimaryKey
    @NonNull
    @ColumnInfo(name = "email")
    public String email;

    @NonNull
    @ColumnInfo(name = "password")
    public String password;

    public User(@NonNull String email, @NonNull String password) {
        this.email = email;
        this.password = password;
    }
}

```

app/src/main/java/com/example/loginmodule/data/UserDao.java

```

package com.example.loginmodule.data;

import androidx.room.Dao;
import androidx.room.Delete;
import androidx.room.Insert;
import androidx.room.Query;

import java.util.List;

import com.example.loginmodule.models.User;

@Dao
public interface UserDao {
    @Query("SELECT * FROM user")
    List<User> getAll();

    @Query("SELECT * FROM user WHERE email LIKE :email LIMIT 1")
    User findByEmail(String email);

    @Insert

```

```

        void insert(User user);

        @Delete
        void delete(User user);
    }

```

```

app/src/main/java/com/example/loginmodule/data/UserDatabase.java
package com.example.loginmodule.data;

```

```

import androidx.room.RoomDatabase;
import androidx.room.Database;

```

```

import com.example.loginmodule.models.User;

```

```

@Database(entities = {User.class}, version = 1)
public abstract class UserDatabase extends RoomDatabase {
    public abstract UserDao userDao();
}

```

```

app/src/main/java/com/example/loginmodule/services/UserService.java
package com.example.loginmodule.services;

```

```

import android.content.Context;

```

```

import androidx.room.Room;

```

```

import com.example.loginmodule.data.UserDatabase;
import com.example.loginmodule.models.User;

```

```

public class UserService {
    private User user = null;
    private static UserService instance = null;
    private static UserDatabase database = null;

    public void login(User user) throws IllegalStateException {
        User u = database.userDao().findByEmail(user.email);
        if (!u.password.equals(user.password))
            throw new IllegalStateException("Password Mismatch");
        this.user = user;
    }

    public void register(User user) {
        database.userDao().insert(user);
        this.user = user;
    }

    public void logout() {
        this.user = null;
    }
}

```

```

    }

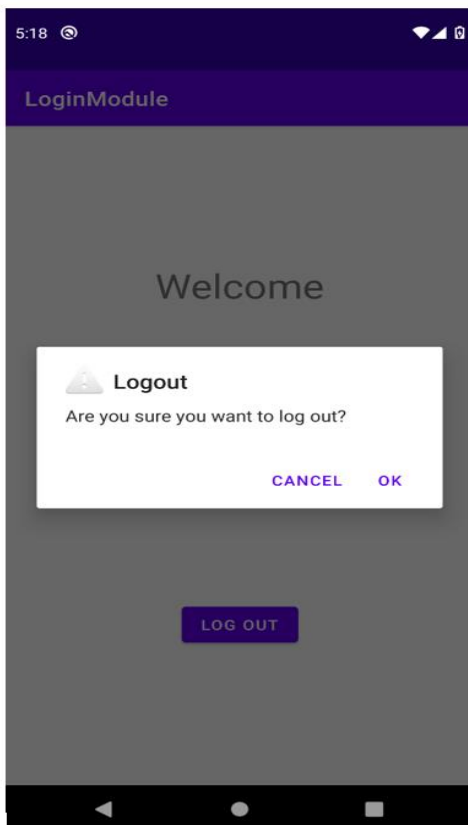
    public User getUser() {
        return user;
    }

    public static UserService getInstance() {
        if (instance == null) {
            instance = new UserService();
        }
        return instance;
    }

    public static void initDatabase(Context ctx) {
        if (database == null) {
            database = Room
                .databaseBuilder(ctx, UserDatabase.class, "login_module")
                .allowMainThreadQueries()
                .build();
        }
    }
}

```

Output:



Objective of Practical 12:

Create an application to Create, Insert, update and Delete operation on the database.

Code:

app/src/main/res/layout/activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/c1Root"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <androidx.swiperefreshlayout.widget.SwipeRefreshLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/swipeLayout"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent">

        <androidx.recyclerview.widget.RecyclerView
            android:id="@+id/rvTodos"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </androidx.swiperefreshlayout.widget.SwipeRefreshLayout>

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/fabAdd"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="end|bottom"
        android:layout_margin="16dp"
        android:src="@drawable/ic_add"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent" />

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

app/src/main/res/layout/todo_item.xml

```
<?xml version="1.0" encoding="utf-8"?>
<com.google.android.material.card.MaterialCardView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/cvParent"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginHorizontal="@dimen/card_margin"
    android:layout_marginTop="@dimen/card_margin"
    app:cardCornerRadius="@dimen/cardview_default_radius"
    app:cardElevation="@dimen/cardview_default_elevation">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="@dimen/default_card_content_margin">

        <TextView
            android:id="@+id/tvTodoId"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:visibility="invisible" />

        <TextView
            android:id="@+id/tvTodoContent"
            style="@style/TextAppearance.MaterialComponents.Headline6"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/default_todo_content" />

        <TextView
            android:id="@+id/tvTodoAddedAt"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_below="@id/tvTodoContent"
            android:layout_marginVertical="5dp"
            android:text="@string/default_todo_added_at" />

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/tvTodoAddedAt"
        android:orientation="horizontal">

        <Button
            android:id="@+id/btnTodoAction"
            style="@style/Widget.MaterialComponents.Button.OutlinedButton"
```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/default_todo_action" />

        <Button
            android:id="@+id/btnTodoDelete"
            style="@style/Widget.MaterialComponents.Button.OutlinedButton"
            android:textColor="@android:color/holo_red_dark"
            android:layout_marginStart="10dp"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/default_todo_delete" />
    </LinearLayout>
</RelativeLayout>
</com.google.android.material.card.MaterialCardView>

```

app/src/main/res/layout/dialog_layout.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:paddingHorizontal="20dp"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <EditText
        android:id="@+id/etNewContent"
        android:inputType="text"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Content" />
</LinearLayout>

```

app/src/main/res/values/strings.xml

```

<resources>
    <string name="app_name">DatabaseCRUD</string>
    <string name="default_todo_added_at">YYYY-MM-DD HH:MM:SS</string>
    <string name="default_todo_content">Todo</string>
    <string name="default_todo_action">Mark as Done</string>
    <string name="default_todo_delete">Delete</string>
</resources>

```

app/src/main/res/values/dimens.xml

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <dimen name="default_card_image_height">150dp</dimen>
    <dimen name="default_card_content_margin">10dp</dimen>
    <dimen name="card_margin">25dp</dimen>

```

```

        <dimen name="logo_dim">200dp</dimen>
    </resources>
app/src/main/java/com/example/databasecrud/MainActivity.java
package com.example.databasecrud;

import android.app.AlertDialog;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import androidx.swiperefreshlayout.widget.SwipeRefreshLayout;

import com.google.android.material.floatingactionbutton.FloatingActionButton;

import com.example.databasecrud.adapters.RVAdapter;
import com.example.databasecrud.repos.TODORepository;

public class MainActivity extends AppCompatActivity implements SwipeRefreshLayout.OnRefreshListener {
    private TODORepository todoRepository;
    private RecyclerView rvTodos;
    private RVAdapter todosAdapter;
    private SwipeRefreshLayout swipeLayout;

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

        todoRepository = new TODORepository(this);

        todosAdapter = new RVAdapter(this, todoRepository);

        swipeLayout = (SwipeRefreshLayout) findViewById(R.id.swipeLayout);
        swipeLayout.setOnRefreshListener(this);

        rvTodos = (RecyclerView) findViewById(R.id.rvTodos);
        rvTodos.setAdapter(todosAdapter);
        rvTodos.setLayoutManager(new LinearLayoutManager(this));

        FloatingActionButton fabAdd = (FloatingActionButton)
        findViewById(R.id.fabAdd);

```

```

        fabAdd.setOnClickListener(v -> {
            View layout =
                getLayoutInflater().inflate(R.layout.dialog_layout, null);
            EditText etNewContent = (EditText) layout.findViewById(R.id.etNewContent);

            AlertDialog dialog = new AlertDialog.Builder(this)
                .setView(layout)
                .setTitle("Add New Todo")
                .setPositiveButton("Save", (_dialog, which) -> {
                    String content = etNewContent.getText().toString().trim();
                    if (content.length() == 0) {
                        Toast.makeText(this, "No Text Entered",
                            Toast.LENGTH_SHORT).show();
                    } else {
                        todoRepository.create(content);
                        loadTodos();
                    }
                })
                .setNegativeButton("Cancel", (_dialog, which) -
                    > _dialog.dismiss())
                .create();
            dialog.show();
        });
        loadTodos();
    }

    @Override
    public void onRefresh() {
        loadTodos();
    }

    private void loadTodos() {
        swipeLayout.setRefreshing(true);
        todosAdapter.setTodos(todoRepository.getAll());
        swipeLayout.setRefreshing(false);
    }
}

```

app/src/main/java/com/example/databasecrud/adapters/RVAdapter.java

```
package com.example.databasecrud.adapters;
```

```

import android.content.Context;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

```

```
import androidx.annotation.NonNull;
```



```
import androidx.recyclerview.widget.RecyclerView;

import com.google.android.material.card.MaterialCardView;

import java.util.ArrayList;

import com.example.databasecrud.R;
import com.example.databasecrud.models.Todo;
import com.example.databasecrud.repos.TodoRepository;
```

```
public class RVAdapter extends RecyclerView.Adapter<RVAdapter.ViewHolder> {
    private final Context ctx;
    private final TodoRepository repository;
    private ArrayList<Todo> todos = new ArrayList<>();

    public RVAdapter(Context context, TodoRepository repository) {
        this.ctx = context;
        this.repository = repository;
    }
}
```

```

@NonNull
@Override
public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
    View view = LayoutInflater
        .from(parent.getContext())
        .inflate(R.layout.todo_item, parent, false);
    return new ViewHolder(view);
}

@Override
public void onBindViewHolder(@NonNull ViewHolder holder, int position) {
    holder.cvParent.setOnClickListener(v ->
        Log.d("DEBUG", todos.get(position).getId() + " Clicked")
    );
}

```

```

holder.tvTodoContent.setText(todos.get(position).getContent());
holder.tvTodoAddedAt.setText(todos.get(position).getAddedAt().toString

    if (todos.get(position).isDone()) { holder.btnTodoAction.setText("Mark as Not
        Done");
    } else {
        holder.btnTodoAction.setText("Mark as Done");}

holder.btnTodoAction.setOnClickListener(v -> {
    repository.update(todos.get(position).getId()); Toast.makeText(ctx, "Todo
    State Toggled", Toast.LENGTH_SHORT).show();

});

    holder.btnTodoDelete.setOnClickListener(v -> {
        repository.delete(todos.get(position).getId());
        Toast.makeText(ctx, "Todo Deleted",
        Toast.LENGTH_SHORT).show();});}

});
}

() { return todos.size();
}

void setTodos(@Override
ArrayList<Todo> todos) { this.todos = todos;
    notifyDataSetChanged();
}
public static class ViewHolder extends RecyclerView.ViewHolder {
    private final TextView tvTodoId;
    private final TextView tvTodoContent;
    private final TextView tvTodoAddedAt;
    private final Button btnTodoAction;
    private final Button btnTodoDelete;
    private final MaterialCardView cvParent;

    public ViewHolder(@NonNull View itemView) {
        super(itemView);
        tvTodoId = itemView.findViewById(R.id.tvTodoId);
        tvTodoContent = itemView.findViewById(R.id.tvTodoContent);
        tvTodoAddedAt = itemView.findViewById(R.id.tvTodoAddedAt);
        btnTodoAction = itemView.findViewById(R.id.btnTodoAction);
        btnTodoDelete = itemView.findViewById(R.id.btnTodoDelete);
        cvParent = itemView.findViewById(R.id.cvParent);
    }
}

```

```
    }  
}
```

app/src/main/java/com/example/databasecrud/models/ToDo.java

```
package com.example.databasecrud.models;
```

```
import java.time.LocalDateTime;
```

```
import java.util.Date;
```

```
public class ToDo {
```

```
    private String id;
```

```
    private String content;
```

```
    private LocalDateTime addedAt;
```

```
    private boolean isDone;
```

```
    public ToDo(String id, String content, LocalDateTime addedAt, boolean isDone) {
```

```
        this.id = id;
```

```
        this.content = content;
```

```
        this.addedAt = addedAt;
```

```
        this.isDone = isDone;
```

```
    }
```

```
    return "ToDo{" +
```

```
        "id='" + id + '\'' +
```

```
        ", content='" + content + '\'' +
```

```
        ", addedAt=" + addedAt +
```

```
        ", isDone=" + isDone +
```

```
        '}';
```

```
    () { return id;
```

```
    }
```

```
    public void setId(String id) {
```

```
        this.id = id;
```

```
    }
```

```
    public String getContent() {
```

```
        return content;
```

```
    }
```

```
    public void setContent(String content) {
```

```
        this.content = content;
```

```
    }
```

```
    public LocalDateTime getAddedAt() {
```

```
        return addedAt;
```

```
    }
```

```
    public void setAddedAt(LocalDateTime addedAt) {
```

```

        this.addedAt = addedAt;
    }

    public boolean isDone() {
        return isDone;
    }

    public void setDone(boolean done) {
        isDone = done;
    }

    @Override
    public String toString() {

        return "Todo{" +
            "id='" + id + '\'' +
            ", content='" + content + '\'' +

            ", addedAt=" + addedAt +

            ", isDone=" + isDone +

            '}';
    }
}

```

app/src/main/java/com/example/databasecrud/repos/ToDoRepository.java

```
package com.example.databasecrud.repos;
```

```

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.util.Log;

```

```

import java.time.LocalDate;
import java.util.ArrayList;
import java.util.UUID;

```

```

import com.example.databasecrud.models.Todo;
import com.example.databasecrud.services.DBService;

```

```

public class ToDoRepository {
    private SQLiteDatabase db;
    private DBService dbService;

    public ToDoRepository(Context context) {

```

```

        this.dbService = DBService.getInstance(context, "todos", 1);
        this.db = this.dbService.getWritableDatabase();
    }

    public boolean create(String content) {
        try {
            ContentValues contentValues = new ContentValues();
            contentValues.put("Id", UUID.randomUUID().toString());
            contentValues.put("Content", content);
            contentValues.put("AddedAt", LocalDateTime.now().toString());
            contentValues.put("IsDone", false);
            db.insert("todos", null, contentValues);
            return true;
        } catch (Exception e) {
            return false;
        }
    }

    public ArrayList<Todo> getAll() {
        ArrayList<Todo> todos = new ArrayList<>();
        Cursor res = db.rawQuery("SELECT * FROM TODOS", null);
        res.moveToFirst();
        while (!res.isAfterLast()) {
            todos.add(
                new Todo(
                    res.getString(res.getColumnIndex("Id")),
                    res.getString(res.getColumnIndex("Content")),
                    LocalDateTime.parse(res.getString(res.getColumnIndex(
ex("AddedAt"))),
));
                    res.moveToNext();
                }
            res.close();
            for (Todo todo: todos) {
                Log.d("TODO", todo.toString());
            }
            return todos;
        }

        public Todo get(String id) {
            Cursor res = db.rawQuery("SELECT * FROM TODOS WHERE Id = " + id, null);
            Todo todo = new Todo(
                res.getString(res.getColumnIndex("Id")),
                res.getString(res.getColumnIndex("Content")),
                LocalDateTime.parse(res.getString(res.getColumnIndex("AddedAt"
))),
            res.getInt(res.getColumnIndex("IsDone")) > 0
            );
            res.close();
            return todo;

```

```

    }
    public void update(String id) {
        db.execSQL("UPDATE TODOS SET IsDone = NOT IsDone WHERE Id = '" + id +
            "'");
    }
    public void delete(String id) {
        db.execSQL("DELETE FROM TODOS WHERE Id = '" + id + "'");
    }
}

```

app/src/main/java/com/example/databasecrud/services/DBService.java

```
package com.example.databasecrud.services;
```

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

```
import androidx.annotation.Nullable;
```

```
public class DBService extends SQLiteOpenHelper {
    private static DBService instance = null;

    public DBService(@Nullable Context context, @Nullable String name, @Nullable SQLiteDatabase.CursorFactory factory, int version) {
        super(context, name, factory, version);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL(
            "CREATE TABLE TODOS ("
                + "Id INT,"
                + "Content VARCHAR(255),"
                + "AddedAt DATETIME,"
                + "IsDone BOOL"
            + ");"
        );
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS TODOS;");
        onCreate(db);
    }

    public static DBService getInstance(Context context, String name, int version) {
        if (instance == null) {
            instance = new DBService(context, name, null, version);
        }
    }
}

```

```

        }
        return instance;
    }
}
app/build.gradle
plugins {
    id 'com.android.application'
}

android {
    compileSdkVersion 30
    buildToolsVersion "30.0.3"

    defaultConfig {
        applicationId "com.example.databasecrud"
        minSdkVersion 29
        targetSdkVersion 30
        versionCode 1
        versionName "1.0"

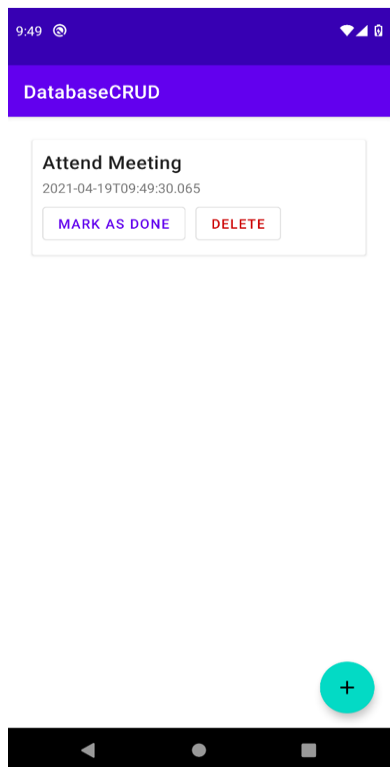
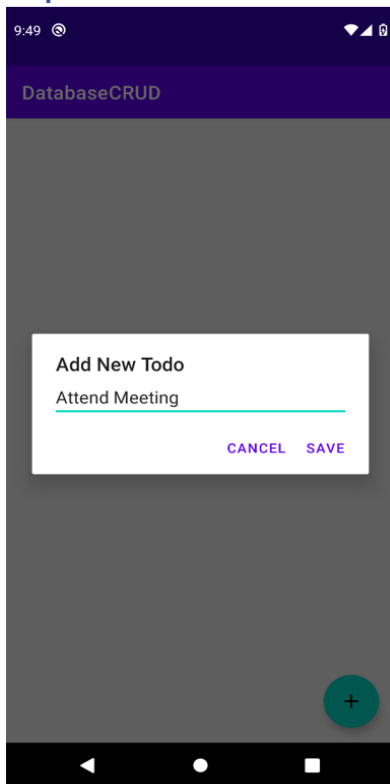
        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
    }

    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android-
optimize.txt'), 'proguard-rules.pro'
        }
    }
    compileOptions {
        sourceCompatibility JavaVersion.VERSION_1_8
        targetCompatibility JavaVersion.VERSION_1_8
    }
}

dependencies {
    implementation "androidx.swiperefreshlayout:swiperefreshlayout:1.1.0"
    implementation 'androidx.appcompat:appcompat:1.2.0'
    implementation 'com.google.android.material:material:1.3.0'
    implementation 'androidx.constraintlayout:constraintlayout:2.0.4'
    testImplementation 'junit:junit:4.+'
    androidTestImplementation 'androidx.test.ext:junit:1.1.2'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.3.0'
}

```


Output:

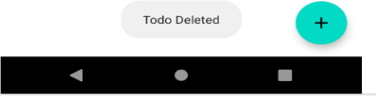
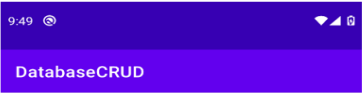
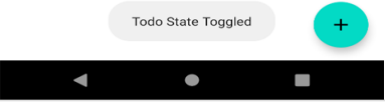




Attend Meeting
2021-04-19T09:49:30.065

MARK AS NOT DONE

DELETE



----- *

