

The Android Development Goals

Android Studio

1. Introduction

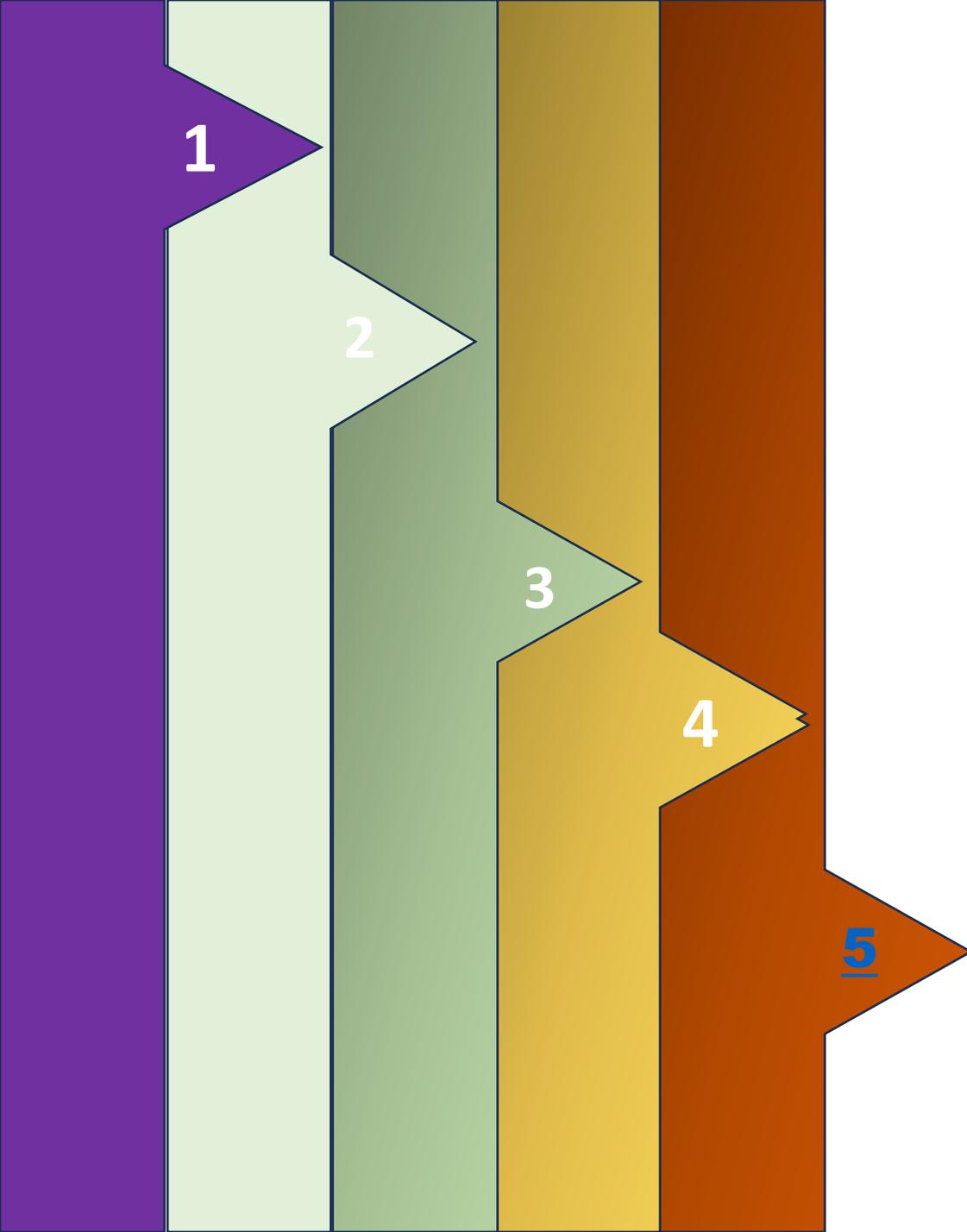
2. Design Process

Tasks

3. ToDo List

4. TapImage

5. Conclusion



Introduction

- Android Studio is the official integrated development environment (IDE) for Android app development.

What is Android Studio?

- Android Studio is an IDE specifically designed for developing Android apps.
- It is based on IntelliJ IDEA
- and provides a powerful and flexible environment for

Building, Testing, and Debugging Android apps.



Android Studio



Design Process

In this Software Android studio

- We need to create new project
- Then it shows the main_activity.xml page
- It is a layout page where we can drag and paste the buttons and texts here

Palates

- Here the plain texts and buttons etc are present

Attributes

- Here we are making changes like
Id's , Text size , Layout_Constraints, etc

Build

in menu we having build to run and check errors

Emulator

On click Running devices after successfully build and shows the Output



Task 1

ToDo List

Here, the on click view app shows the Texts which we has entered in the Plain TextView.

- By clicking on Button (new) creates new notes.
- By clicking on Button (delete) deletes current notes.
- By clicking on Button (save) saves whole notes.



Resource Manager

- app
 - manifests
 - AndroidManifest.xml
 - java
 - com.example.todo
 - DatabaseHelper
 - MainActivity
 - com.example.todo (androidTest)
 - com.example.todo (test)
 - java (generated)
 - res
 - drawable
 - layout
 - mipmap
 - values
 - xml
 - res (generated)
 - Gradle Scripts
 - build.gradle (Project: ToDo)
 - build.gradle (Module :app)
 - proguard-rules.pro (ProGuard Rules for Release)
 - gradle.properties (Project Properties)
 - gradle-wrapper.properties (Gradle Wrapper Properties)
 - local.properties (SDK Location)
 - settings.gradle (Project Settings)

Project

Structure

Build Variants

Bookmarks

```

23 public class MainActivity extends AppCompatActivity {
24     // DataBase
25     private DatabaseHelper dbHelper;
26     private LinearLayout containerLayout;
27     private Button newButton;
28     private Button deleteButton;
29
30     // to save button
31     private LinearLayout editTextContainer;
32     private ArrayList<EditText> editTextList;
33     private static final String PREFS_NAME = "MyPrefsFile";
34
35     @RequiresApi(api = Build.VERSION_CODES.O)
36     @Override
37     protected void onCreate(Bundle savedInstanceState) {
38         super.onCreate(savedInstanceState);
39         setContentView(R.layout.activity_main);
40
41         // Getting database file to a variable dbHelper
42
43         dbHelper = new DatabaseHelper(context, this);
44         editTextContainer = findViewById(R.id.containerLayout);
45
46         // method for save button to store all data
47
48         editTextContainer = findViewById(R.id.containerLayout);
  
```

Running Devices: Pixel 4a API 32

Device Manager

Notifications

Gradle

Running Devices

Device Explorer

Resource Manager

- app
 - manifests
 - AndroidManifest.xml
 - java
 - com.example.todo
 - DatabaseHelper
 - MainActivity
 - com.example.todo (androidTest)
 - com.example.todo (test)
 - java (generated)
 - res
 - drawable
 - layout
 - mipmap
 - values
 - xml
 - res (generated)
- Gradle Scripts
 - build.gradle (Project: ToDo)
 - build.gradle (Module :app)
 - proguard-rules.pro (ProGuard Rules for Release)
 - gradle.properties (Project Properties)
 - gradle-wrapper.properties (Gradle Wrapper Properties)
 - local.properties (SDK Location)
 - settings.gradle (Project Settings)

Project

Structure

Build Variants

Bookmarks

```

40
41 // Getting database file to a variable dbHelper
42
43 dbHelper = new DatabaseHelper( context: this);
44 editTextContainer = findViewById(R.id.containerLayout);
45
46 // method for save button to store all data
47
48 editTextContainer = findViewById(R.id.containerLayout);
49 editTextList = new ArrayList<>();
50
51 // logic for save button
52
53 Button saveButton = findViewById(R.id.save);
54
55 //
56
57 saveButton.setOnClickListener(new View.OnClickListener() {
58     @Override
59     public void onClick(View v) { saveEditTextContent(); }
60 });
61
62 loadSavedEditTextContent();
63
64 containerLayout = findViewById(R.id.containerLayout);
65 newButton = findViewById(R.id.newbutton);
66 deleteButton = findViewById(R.id.delete);
67
68 // Giving onclick method to new button to create new texts
69
70 newButton.setOnClickListener(new View.OnClickListener() {
71     @Override
72     public void onClick(View v) { addNewEditableTextView(); }
73 });
74
75
76

```

Running Devices: Pixel 4a API 32

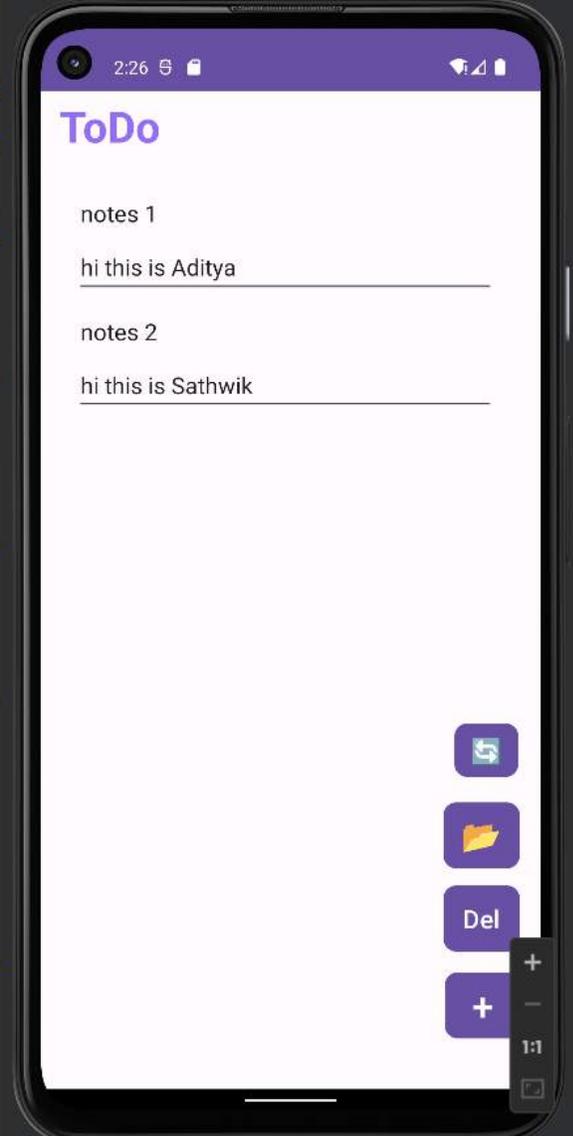
Device Manager

Notifications

Gradle

Running Devices

Device Explorer



Resource Manager

- app
 - manifests
 - AndroidManifest.xml
 - java
 - com.example.todo
 - DatabaseHelper
 - MainActivity
 - com.example.todo (androidTest)
 - com.example.todo (test)
 - java (generated)
 - res
 - drawable
 - layout
 - mipmap
 - values
 - xml
 - res (generated)
 - Gradle Scripts
 - build.gradle (Project: ToDo)
 - build.gradle (Module :app)
 - proguard-rules.pro (ProGuard Rules for Release)
 - gradle.properties (Project Properties)
 - gradle-wrapper.properties (Gradle Wrapper Properties)
 - local.properties (SDK Location)
 - settings.gradle (Project Settings)

Project

Structure

Build Variants

Bookmarks

```

47
48 editTextContainer = findViewById(R.id.containerLayout);
49 editTextList = new ArrayList<>();
50
51 // logic for save button
52
53 Button saveButton = findViewById(R.id.save);
54
55 //
56 saveButton.setOnClickListener(new View.OnClickListener() {
57     @Override
58     public void onClick(View v) { saveEditTextContent(); }
59 });
60
61 loadSavedEditTextContent();
62
63
64 containerLayout = findViewById(R.id.containerLayout);
65 newButton = findViewById(R.id.newbutton);
66 deleteButton = findViewById(R.id.delete);
67
68 // Giving onclick method to new button to create new texts
69
70 newButton.setOnClickListener(new View.OnClickListener() {
71     @Override
72     public void onClick(View v) { addNewEditableTextView(); }
73 });
74
75 // Giving onclick method to delete button to delete created texts
76
77 deleteButton.setOnClickListener(new View.OnClickListener() {
78     @Override
79     public void onClick(View v) { deleteSelectedTextView(); }
80 });
81
82
83
84
85
86

```

Running Devices: Pixel 4a API 32

Device Manager

Notifications

Gradle

Running Devices

Device Explorer



Resource Manager

- app
 - manifests
 - AndroidManifest.xml
 - java
 - com.example.todo
 - DatabaseHelper
 - MainActivity
 - com.example.todo (androidTest)
 - com.example.todo (test)
 - java (generated)
 - res
 - drawable
 - layout
 - mipmap
 - values
 - xml
 - res (generated)
 - Gradle Scripts
 - build.gradle (Project: ToDo)
 - build.gradle (Module :app)
 - proguard-rules.pro (ProGuard Rules for Module :app)
 - gradle.properties (Project Properties)
 - gradle-wrapper.properties (Gradle Wrapper Properties)
 - local.properties (SDK Location)
 - settings.gradle (Project Settings)

Project

Structure

Build Variants

Bookmarks

```

104  }
105  // save button method
106  // 1 usage
107  private void saveEditTextContent() {
108  //  SharedPreferences.Editor editor = getSharedPreferences(PREFS_NAME, Context.MODE_PRIVATE).edit();
109  //  Set<String> editTextValues = new HashSet<>();
110
111  SQLiteDatabase db = dbHelper.getWritableDatabase();
112
113  // Clear existing data
114  db.delete( table: "mytable", whereClause: null, whereArgs: null);
115
116  for (int i = 0; i < editTextContainer.getChildCount(); i++) {
117  EditText editText = (EditText) editTextContainer.getChildAt(i);
118  ContentValues values = new ContentValues();
119  values.put("content", editText.getText().toString());
120  db.insert( table: "mytable", nullColumnHack: null, values);
121  }
122
123  db.close();
124  // 1 usage
125  private void deleteSelectedTextView() { // Delete method
126  int childCount = containerLayout.getChildCount();
127  for (int i = 0; i < childCount; i++) {
128  View childView = containerLayout.getChildAt(i);
129  if (childView instanceof EditText) {
130  EditText noteEditText = (EditText) childView;
131  if (noteEditText.isFocused()) {
132  containerLayout.removeViewAt(i);
133  break; // Stop searching after deleting the first focused TextView
134  }
135  }

```

Running Devices: Pixel 4a API 32

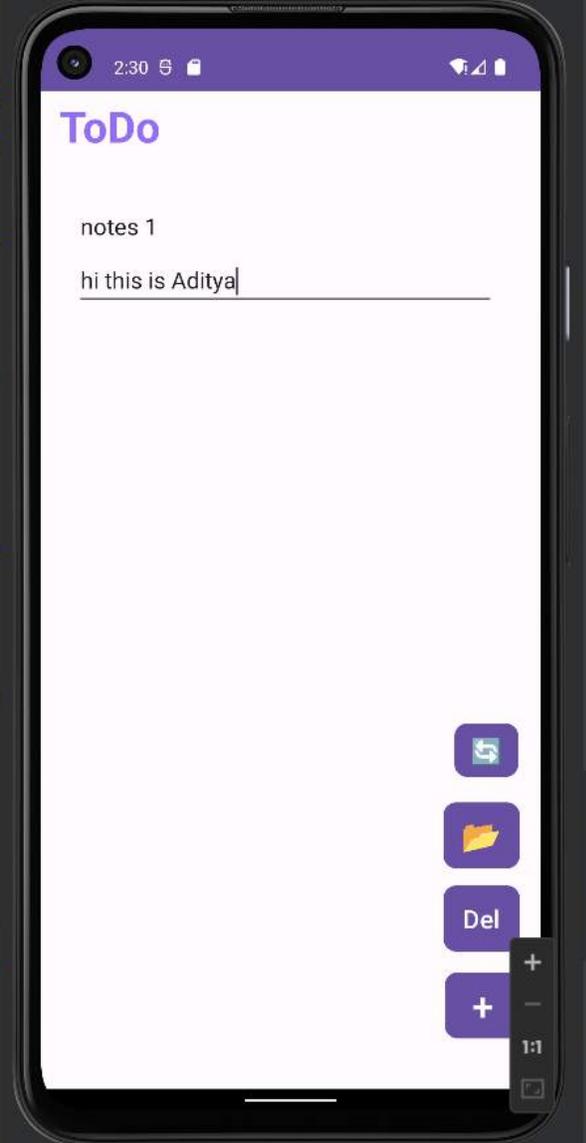
Device Manager

Notifications

Gradle

Running Devices

Device Explorer



Resource Manager

- app
 - manifests
 - AndroidManifest.xml
 - java
 - com.example.todo
 - DatabaseHelper
 - MainActivity
 - com.example.todo (androidTest)
 - com.example.todo (test)
 - java (generated)
 - res
 - drawable
 - layout
 - mipmap
 - values
 - xml
 - res (generated)
 - Gradle Scripts
 - build.gradle (Project: ToDo)
 - build.gradle (Module :app)
 - proguard-rules.pro (ProGuard Rules for)
 - gradle.properties (Project Properties)
 - gradle-wrapper.properties (Gradle Wrapper)
 - local.properties (SDK Location)
 - settings.gradle (Project Settings)

Project

Bookmarks

Build Variants

Structure

```

// now writing logic to the methods
no usages
public void OnAddButtonClick(View view)
{addNewEditableTextView();}

2 usages
private void addNewEditableTextView() {
    EditText noteEditText = new EditText(context: this);
    LinearLayout.LayoutParams layoutParams = new LinearLayout.LayoutParams(
        LinearLayout.LayoutParams.MATCH_PARENT,
        LinearLayout.LayoutParams.WRAP_CONTENT );

    layoutParams.setMargins(left: 0, top: 0, right: 0, bottom: 16); // Adds margin to the texts between
    noteEditText.setLayoutParams(layoutParams);
    containerLayout.addView(noteEditText);
}

// save button method
1 usage
private void saveEditTextContent() {
    SQLiteDatabase db = dbHelper.getWritableDatabase();

    // Clear existing data
    db.delete(table: "mytable", whereClause: null, whereArgs: null);

    for (int i = 0; i < editTextContainer.getChildCount(); i++) {
        EditText editText = (EditText) editTextContainer.getChildAt(i);
        ContentValues values = new ContentValues();
        values.put("content", editText.getText().toString());
        db.insert(table: "mytable", nullColumnHack: null, values);
    }

    db.close();
}

```

Running Devices: Pixel 4a API 32

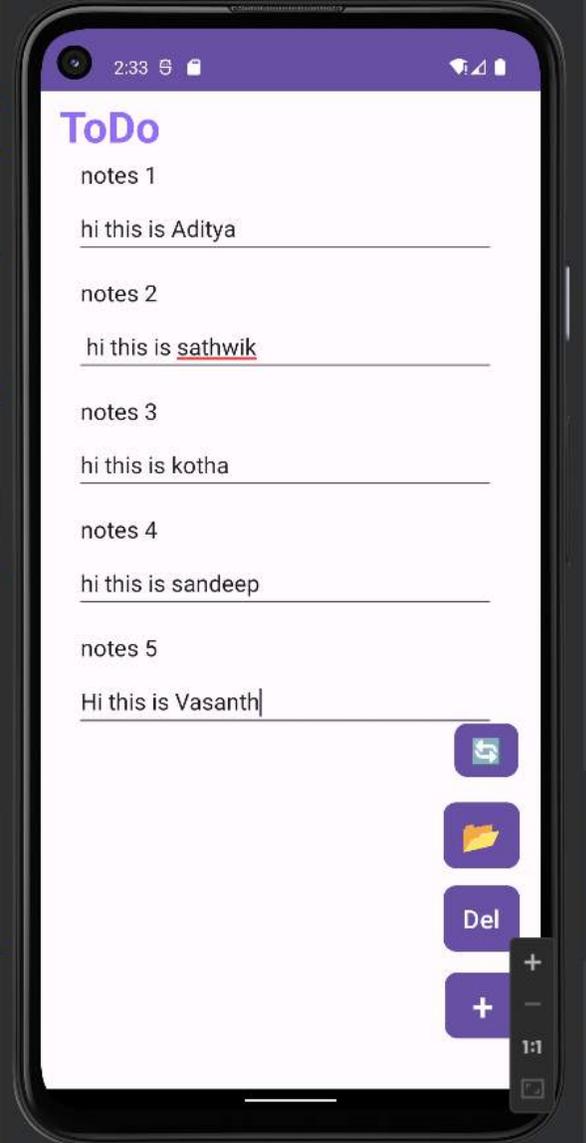
Device Manager

Notifications

Gradle

Running Devices

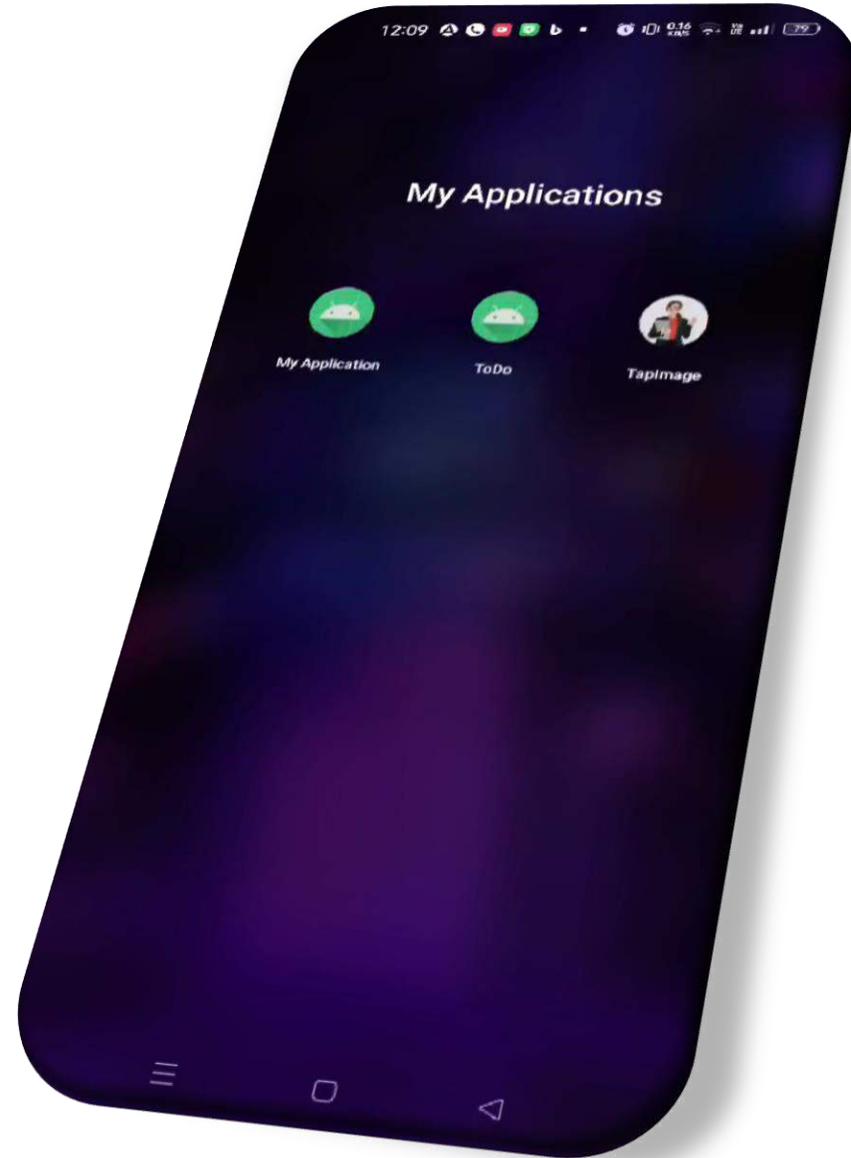
Device Explorer



Task 2

TapImage

- Here, we are going to exhibit simple Onclick open website.
- In this we are clicking on button (Get started) then it takes to chrome and opens website.



File Edit View Navigate Code Refactor Build Run Tools VCS Window Help TapImage - MainActivity.java [TapImage.app.main]

TapImage > app > src > main > java > com > example > tapimage > MainActivity > btn

Android activity_main.xml x MainActivity.java x

Resource Manager Project Structure Build Variants

- app
 - manifests
 - AndroidManifest.xml
 - java
 - com.example.tapimage
 - MainActivity
 - com.example.tapimage (androidTest)
 - com.example.tapimage (test)
 - java (generated)
 - res
 - drawable
 - hiivector.jpg
 - ic_launcher_background.xml
 - ic_launcher_foreground.xml (v24)
 - layout
 - activity_main.xml
 - mipmap
 - ic_launcher (6)
 - ic_launcher_round (6)
 - values
 - colors.xml
 - strings.xml
 - themes (2)
 - xml
 - backup_rules.xml
 - data_extraction_rules.xml
 - res (generated)
 - Gradle Scripts
 - build.gradle (Project: TapImage)
 - build.gradle (Module :app)
 - proguard-rules.pro (ProGuard Rules for ".ap
 - gradle.properties (Project Properties)
 - gradle-wrapper.properties (Gradle Version)
 - local.properties (SDK Location)
 - settings.gradle (Project Settings)

```
1 package com.example.tapimage; // MainActivity.java
2 import ...
8
9 public class MainActivity extends AppCompatActivity {
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_main);
15     }
16
17     // To open a website page using URL from application we use "Intent"
18     public void btn(View view)
19     {
20         String w = "https://swarnandhra.ac.in/index.php";
21         Intent i = new Intent(Intent.ACTION_VIEW, Uri.parse(w));
22         startActivity(i);
23     }
24 }
25
```

Running Devices - TapImage

Running Devices: Pixel 4a API 32

2:03 3G

Welcome

On Click view image



Get Started >

17:75 LF UTF-8 4 spaces

Install successfully finished in 11 s 737 ms. (8 minutes ago)

Conclusion

- Here, we are exhibits the Application which is going to do that On Click button to perform an action
- So the main thing is giving appropriate listeners with correct id's

Error rectification

- Should import correct packages
- The activities should link up with each i.e MainActivity.java to activity_main.xml

