

Assignment No.2

Page

Q.1 a) Explain layout and its types

→ (1) Layout basically refers to the arrangement of elements on a page. These elements are likely to be images, texts or styles.

(2) It is used to define the UI that holds UI controls or widgets that will appear on the screen of an android app.

(3) There are no of layouts provided by android which you will use in almost all the android app to provide different view, look and feel.

* Types:-

(1) Linear Layout: It is a view group that align all children in a single direction, vertically or horizontally.

(2) Relative Layout: It is a view group that display child view in relative position.

(3) Table Layout: It is a view that groups views into rows and columns.

(4) Absolute Layout: It enables you to specify the exact location of its children.

(5) Frame Layout: It is a placeholder on screen that you can use to display a single view.

⑥ List View: It is a view group that displays a list of scrollable items.

⑦ Grid View: It is a view group that displays items in a two-dimensional, scrollable grid.

Q1(b) Explain User interface screen elements?

→ 1) A view is an object that draws something on the screen that the user can interact with and a view group is an object that holds other view objects in order to define layout of UI.

2) There are no. of UI controls provided by android that allow you to build the graphical UI:

a) TextView: This control is used to display text to the user.

b) EditText: It is a predefined subclass of TextView that includes rich editing capabilities.

c) Datepicker: The Datepicker view enables users to select a date of the day.

d) Timepicker: It view enables users to select a time duration of day.

e) Spinner: A drop-down list that allows user to select one value from a set.

f) ProgressBar: It provides visual feedback about some ongoing task.

g) Radio button: It has two states: either checked or unchecked.

b) RadioGroup: It is used to group together one or more RadioButton.

i) ToggleButton: An on/off button with light indicator.

ii) Button: A push-button that can be pressed, or clicked by the user to perform an action.

Ques. 2) Describe Animation.

→ ① Animations can add visual cues that notify users about what's going on in your app.

② They are especially useful when the UI changes state, such as when new content loads or new actions become available.

③ Animations also add a polished look to your app, which gives it a higher quality look and feel.

④ The animations are basically of three types as follows:

a) property animation: → This is the most commonly used animation. It is one of the robust frameworks which allows animating almost everything. It can be used to add animation in checkbox, Radio button, and widgets other than any view.

b) View animation: → Ability to perform animation on views.

It can be used to add animation to a specific view to perform tweened animation on views. Example can be seen in expandable Recycler View.

c) Drawable Animation : →

It is used if you want to combine one image over other.

Q. Important methods of Animations are:

a) startAnimation() : This method will start the animation.

b) clearAnimation() : This method will clear the animation running on a specific view.

Q. 26) Explain working with different type of resources.

→ ① Animation resources : →

Refine pre-determined animations. Tween animations are saved in res/anim/ and accessed from the R.anim class.

Frame animations are saved in res/drawable/ and accessed from the R.drawable class.

② color state list resource : →

Define a color resource that changes based on the view's state. Saved in res/color/ and accessed from the R.color class.

③ drawable resources : →

Define various graphics with bitmaps or XML. Saved in res/drawable/ and accessed from the R.drawable class.

④ Layout resource : →

Define the layout for your application UI. Saved in res/layout/ and accessed from R.layout class.

⑤ Menu resource : →

Define the contents of your application menus. Saved in res/menu/ and accessed from R.menu class.

⑥ String resources : →

Define strings, string arrays and plurals and include string formatting and styling. Saved in res/values/ and accessed from R.string, R.array and R.plurals classes.

⑦ Style resource : →

Define the look and format for UI elements.

⑧ Font resources : →

Define font families and include custom fonts in XML.

Q.3(a) Define APIs and explain their types.

→ * APIs : →

- ① Application program interface (API) is a code for a programmer that they use in their applications.
- ② This code or (API) allows you to add specific functionalities to your application.
- ③ In other word, we can say that APIs are the set of protocols and tools used for building the application.

→ Types : →

There are four main types of APIs:

- ① Open APIs: Also known as public APIs, there are no restrictions to access these types of APIs because they are publicly available.
- ② Partner APIs: A developer needs specific rights or licenses in order to access this type of API because they are not available to the public.
- ③ Internal APIs: Also known as private APIs, only internal systems expose this types of API. They are usually designed for internal use within a company.
- ④ Composite APIs: This type of API combines different data and service APIs. Main uses are to speed up the process of execution and improve the performance.

Ques. 3b) What are the steps and procedure for publishing an android app.

→ Step 1: Sign up:

Sign up for an account on the Android developer console, create an account cost \$ 25.

Step 2: Create new application:

Select the publish an android application option and fill out the details

Step 3: Prepare multimedia:

Screenshots of app, Hi-res icon and feature graphic.

Step 4: Prepare code for release:

- Remove log statements.
- Remove the android:debuggable attribute from your manifest file.
- Set the android:versionCode attribute in manifest.xml.
- Set the android:versionName attribute in the manifest tag in manifest.xml

Step 5: Build a release-ready APK.

The release-ready APK is different from the debug APK in that it is signed with certificate that is owned by the developer.

Android studio → Build → Generate Signed APK

A Java keystore (JKS) is a repository of public-private key pairs. You must sign an APK with the same key pair.

Step 6: Upload APK

Go back to the developer console and click on "Manage Releases". Then create a production release and upload your signed APK. Google will perform a check on the APK. My app was using an icon for the launcher icon, which is not bueno, then change it to PNG and recreate the signed APK.

Step 7:

Complete the checklist on the left until all the items have a green checkmark.

The console re-evaluates the checklist every time you click save draft in the top right.

Q.4 (a) Write short note on:

SQLite with queries on CRUD.

- ① SQLite is an open source SQL database that stores data in the form of file on a device.
- ② SQLite supports all the relational db features to store data permanently.
- ③ CRUD means nothing but an abbreviation for the basic operations that perform in any database. And the operations are Create, Read, update, delete.
- ④ Query for creating a table: →

CREATE TABLE employees (

id INTEGER NOT NULL CONSTRAINT employees_pk

PRIMARY KEY AUTOINCREMENT,

name VARCHAR (200) NOT NULL,

department VARCHAR (200) NOT NULL,

joiningdate datetime NOT NULL,

salary double NOT NULL

);

- ⑤ Creating a new Record: →

INSERT INTO employees

(name, department, joiningdate, salary)

VALUES

('Bimal', 'Technical', '2017-09-3' 10:00:00', '40000');

- ⑥ Reading All existing Records: →

SELECT * FROM employees;

⑦ Reading specific Record :→

```
SELECT * FROM employees WHERE id=1;
```

⑧ Updating a Record :→

```
UPDATE employees
```

```
SET name = 'Belal Haque'
```

```
department = 'Research and development'
```

```
salary = '100000'
```

```
WHERE id=1;
```

⑨ Deleting a Record :→

```
DELETE FROM employees WHERE id=1;
```

Q.8(b) Explain the following w.r.t android testing

① manual testing :→

- Manual testing involves human interaction with app under different conditions or situations to analyze how it responds.

- A human performs the tests step by step without test scripts.

- Manual testing is the process in which QA analysts execute test one-by-one in an individual manner.

- The purpose of manual testing is to catch bugs and features issues before a software application goes live.

- Manual testing should be used to perform

Exploratory testing, Usability testing and Ad-hoc testing to exhibits the best results.

② Automated Testing : →

- Android automated testing requires no human interaction as code is automatically executed and reported.
- Tests are executed automatically via test automation frameworks, along with other tools and sw.
- Automation testing should be used to perform regression testing, load testing, performance testing and repeated execution for best results.
- Automation testing is used to increase the efficiency, effectiveness and coverage.

Q.5 a) observe the following GUI and write an XML file using relative layout to create the same.

		Green color q write some text
Yellow	orange	sky blue

Image
Style: bold, Font: Arial Text color: white Background: dark green.

1. <?xml version = "1.0" encoding = "utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
2. xmlns: android = "http://schemas.android.com/apk/res/android"
3. xmlns: app = "https://schemas.android.com/apk/res-auto"
4. xmlns: tools = "http://schemas.android.com/tools"
5. android: layout-width = "match-parent"
6. android: layout-height = "match-parent"
7. android: background = "drawable / fourth"
8. android: visibility = "visible"
9. tools: context = ".MainActivity"
10. tools: visibility = "visible" >

<EditText

 android: id = "@+id/myTextBox"
 android: layout-width = "336 dp"
 android: layout-height = "101 dp"
 android: background = "#FF0000"
 android: text = "Hello everyone"
 app: layout-constraintBottom-toBottomOf = "parent"
 app: layout-constraintEnd-toEndOf = "parent"
 app: layout-constraintHorizontalBias = "0.115"
 app: layout-constraintStart-toStartOf = "parent"
 app: layout-constraintTop-toTopOf = "parent"
 app: layout-constraintVerticalBias = "0.317" >

< TextView

 android : id = "@+id /myBox1"

 android : layout_width = "91 dp"

 android : layout_height = "95 dp"

 android : background = "@ color /white"

 android : gravity = "center"

 android : text = "@ string /yellow"

 android : textColor = "#827717"

 android : visibility = "visible"

 app : layout_constraintBottom_toBottomOf = "parent"

 app : layout_constraintEnd_toEndOf = "parent"

 app : layout_constraintHorizontal_bias = "0.115"

 app : layout_constraintStart_toStartOf = "parent"

 app : layout_constraintTop_toTopOf = "parent"

 app : layout_constraintVertical_bias = "0.317"

 tools : visibility = "visible" />