

★ ASSIGNMENT - 3 ★

Q : 1 Explain Hardware - imposed criteria for Android Application.

Ans : Hardware Imposed Design
Consideration :-

Design for small and portable mobile opportunities Development.
offer exciting software.

at Be Efficient :-

optimize Always need to
it runs your code so that
quickly and Responsively

Expect Limited capacity
Limited RAM
Limited Permanent storage capacity
small screens with low

Resolution .

low processing power
Compared to Desktop or Laptop
Computers , mobile devices
have Relatively above
mentioned .

Q: 2 Explain Design criteria for Android Application.

Ans :- Hardware Design consideration

- Low Processing Power
- Limited Ram
- Limited permanent storage capacity
- Screen with low resolution.

Design Demands.

- It should be fast
- It should be efficient
- It should use less storage as possible.
- It should present consistent UI for all devices.

Q: 3 Write a short note on Activity

Ans : An Activity Represents a single screen with a interface.

For example, an Email / application might have one activity that shows a list of new emails, another activity to compose an email,

and another activity for
Reading emails

If you have worked with
C, C++ or Java Programming
language than you must
have seen that your
Program starts from function.

Very similar way Android
system initiates its Program
with a call in a Activity starting
with callback a method.
an onCreate()

Q : 4 Explain Android Activity Life
Cycle with Diagram.

Ans : states in Activity Diagram.

~~Running :- Activity interacts with user.~~

~~Paused :- Activity is still visible
but obscured, instance
is running by the system.~~

~~but is killed~~

but might be

Stopped : Activity is not visible
 instance is running visible
 but might be killed by
 the system.

Killed : Activity has been
 terminated by the system
 of its by a call to
 finish () method.

methods :

onStart() :- called when the
 Activity becomes visible to
 the user.

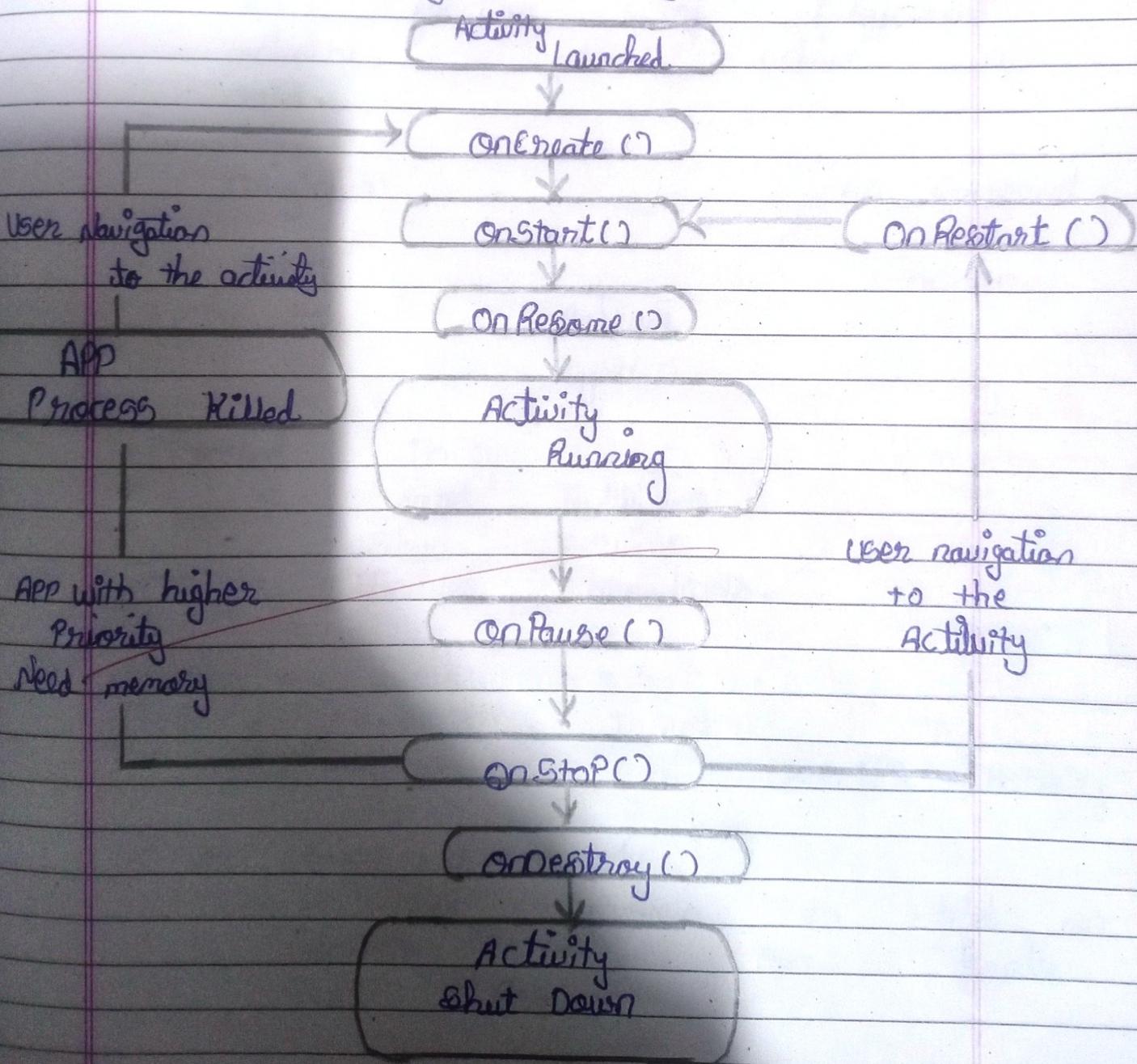
onResume() :- called when the
 Activity starts interacting
 with the user.

onPause() :- called when the current
 Activity is being Paused and
 the previous activity is
 no longer visible to
 the user.

onStop() :- called when the
 Activity is no longer
 visible to the user

`OnDestroy()` : Called before the Activity is Destroyed by the system.

`onRestart` : Called when the Activity has been stopped & is Restarting Again.



Q: 5 what is intent? How to link Activities using different intent?

Ans : An intent in the Android OS is a software mechanism that allows user to coordinate the functions of different activities to achieve a task.

originator

APP Component

Intent

Action

Android system

Intent

Do :

1. Start Activities
2. Start services
3. Deliver broadcasts.

Two types

1. Explicit

Intent

Intent

- starts a specific Activity

2. Implicit Intent

- Asks

Activity

the

System to find an
that can handle
Request.

Starting Activities.

- To start a specific Activity
- Create an intent intent intent = new intent (this, Activity Name . class);
- Use Intent to start the Activity startActivity (intent);

Q :- 6 How to Pass Data between Activities using intent ?

Ans : Suppose we have two Activity activity - first and activity - second which is Destination Activity.

we can send data using the PutExtra() method and to get data from the second Activity using the getStringExtra() method.

PutExtra() method is used for sending the data . data in key value pair key & value

can be int, str, float etc.

getstring Extra () methods is
for getting the Data that
is sent by the above
method According to the
Data type of Value,
there are other method
like get Int Extra (), getFloatExtra

Q : ? what is layout ? list
different types of layout.

Ans : A layout define the
visual structure for an
Android user interface and
can be created either
Declaring your layout using
simple xml file activity
Main.xml

Linear	Layout
Relation	Layout
Table	Layout
Absolute	Layout
Frame	Layout
Grid	view.

Q: 8

Explain linear layout with example.

Ans :

Linear layout is the most basic layout in android studio that aligns all the children sequentially either in a horizontal manner or a vertical manner by specifying the android : orientation attribute.

If one applies orientation vertical elements are arranged in vertical if apply horizontal element arranged in horizontal manner.

Example :

<?xml version = "1.0" encoding = "
UTF - 8" ?>
<LinearLayout

xmlns : android = "http://schemas.
android.com/apk/res/android".

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

xmlns : tools = "http://schemas.android.com/tools".

android : layout_width = "match_parent"

android : layout_height = "match_parent"

android : orientation = "vertical"

tools : context = ".MainActivity" />

<Button

android : layout_width = "match_parent"

android : layout_margin = "10dp"

android : layout_height = "wrap_content" />

<Button

android : layout_width = "match_parent"

android : layout_margin = "10dp"

android : layout_height = "wrap_content" />

<Button

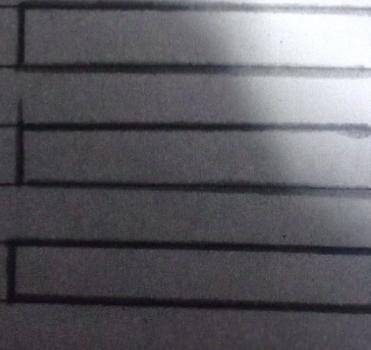
android : layout_width = "match_parent"

android : layout_margin = "10dp"

android : layout_height = "wrap_content" />

</LinearLayout>

Linear Layout



Q: ? Explain Relative layout with example.

Ans : Relative layout allows to relative positive the widget to each other. This can be used for complex layouts.

A simple usage for single Relative layout is if component. Just add one component to the Relative layout and set the android:layout_centerInParent attribute to true.

Example :

```
<?xml version = "1.0" encoding = "utf-8"?>
```

```
<RelativeLayout
```

~~android:layout_width = "fill_parent"~~~~android:layout_height = "fill_parent"~~~~xmlns:android = "http://schemas.~~~~android.com/apk/res/android"}~~

of Button

```
    android:id = "@+id/btn1"
```

```
    android:layout_width = "wrap_content"
```

android : text = "TOP LEFT"

android : layout_align Parent left = "true"

android : layout_align Parent TOP = "TOP" />

Button

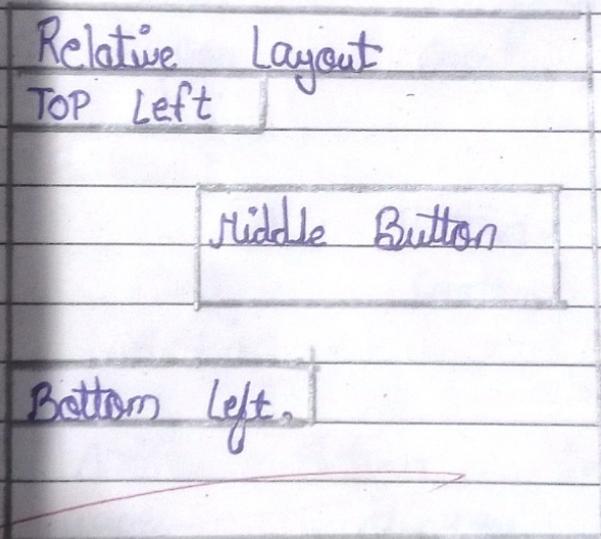
android : layout_width = "fill Parent"

android : layout_height = "wrap_content"

android : layout_centerVertical = "true"

android : layout_centerHorizontal = "true"

{/Relative layout}



Q : 10 List and explain different
5 GUI Controls briefly

Ans : GUI Controls are

1. TextView
2. Push Button
3. Image Button

2. EditText
4. Toggle Button
6. Check Box

7. Radio Button

8. Radio Group

1. TextView :

TextView is a simple UI element.

→ It is used to display static text, which cannot be changed by the user.

2. EditText :

EditText is a widely used UI elements in Android. whenever you want to take some input from the user you can use EditText.

3. Push Button :

A Button is a control used to perform an action.

An Action is performed when a user clicks the button.

4. Toggle Button :

A Toggle Button displays checked / unchecked states as a button.

It is basically an on/off button with a light indicator.

5. Image Button :

It is similar to the Button view, except that it also displays an image.

6. Check Box :

A special type of Button that has two states checked or unchecked.

When you should use checkboxes presenting users with a group of selectable options that are not mutually exclusive.

7. Radio Button :

The Radio Button has two states either checked or unchecked

A Radio Group is used to group Radio button together one or more views. Only one Radio button to be checked within the Radio group.