Basic of Android

(- Prathmesh Prashnat Joshi)

UNIT-1

1. What is Android ?

- Android is an open source and linux based operating system for mobile and computer devices like smart phones and tablet computers . Android is developed by Open Handset Aillance lead by Google and other some compnies .
- Android offers a unified approch for application developers to develope an application for mobile devices that powers by the android .
- The first version of Android Swafter Development Kit (ASDK) are realised in 2007 and the first commercial version of android 1.0 are realised in september 2008 .Then on 27 june 2012 in Google I/O conferance , google declar abut next version of andriod that is android 4.1 jellybean , that is an incremental update .
- The source code for android is available in free and opensource licence . most of the times it is publish under apache tomcat .

2.Why an Android / Advantages of Android ?

- open source .
- large developers and community reached .
- incerased marketing .
- internally intigrated .
- decreased cost of application development .
- higher success ratio .
- reached development environment .

3. Features of android ?

- Beautiful User interface
- Connectivity
- Storage
- Media Support
- Messanging
- web browser

- multi tasking
- multi touch
- molti lenguage
- Support GCM (Google Cloud Messaging)
- Direct WIFI
- Resiable Widget

4.Android Application ? catogries of Android Apllications ? History of Android Versions ?

A) Android Applications

- Android application are develpoed under the java and kotlin languages using Android Software development Kit . Basically java is famous and mostly use language .
- It is easy to Published and sold out on online stores like Google Play store , slide Me , Open mobile Store .

B) catogries of Android Application .

MUSIC SPORTS NEWS MULTIMEDIA
FOOD REFERANCE BOOKS SOCIALMEDIA
TRAVEL WEATHER NEVIGATION
FINANCE

C) history of Android versions

- 1.6 Donut
- 2.0 Eclor
- 2.2 froyo
- 2.3 gingerbead
- 3.0 hunnycomb
- 4.0 ICECreamSandwitch
- 4.1 jellyBean
- 4.4 kitkat
- 5.0 lolipop
- 6.0 macrshmalous
- 7.0 naugot
- 8.0 oreo
- 9.0 pie

5. Android IDE's? Android Architecture ?

- There are so many technique are use to develope the android application , but basically the java language are use to develope the android application with help of Android Software Development Kit .
- Following are some powerfull and mostly using the tools for developing android application .
  1) Android studio
  2) Eclips IDEs
- Android opperating system is an stack of software components which are divided into 4 layers and 5 parts .
  layers are as follows .

5.Applications component
4.Application frmework
3. Android Runtime
2.Libraries
1.Linux Kernel

1. Linux Kernal .

- it is an bottom layer of the architecture
- it is provide the levels of abstraction between hardware and there hardware drivers example camera, media player , sound .
- Kernel also handels the things in that linux is gooded like networking and memory management .

2. Libraries

- this level is top of the linux kernel and it is a second layer of the architecture .
- This layer is set of the use full libraries which are use for the andrioid system .
- This libraries contains the web browser engins , sql data base , ssl ,sgl files , lib files and media framework files .

3. Android Runtime.

- This is a third layer of architecture but it is in secind level .
- This section is provide key component called as Dalvik Vartual Machine is an kind of JVM which specially designed for the Android .

- The Dalvik Vartual Machine use the linux kernal features like memory management and networking and multi Threading which is use full in java .
- Dalvik Vartual Machine have there own instance .
- its provide a core libraries.

4.Application Framework .

- Appliction framework layer provides the many higher level services for the application in the form of java classes , Application developer can allow use that services in there applications .
- Android provides the following key services .

a. Activity manager = Control the appkication life cycle and activitis.
b. Content Provider = Allow the Application publish and share with the other applications.
c. Resource Manager = Provide access to non coded embedded resources such as string and color settings .
d. Notification manager = Allow the Application to display the Alert or notification message.
e. View System = use for create the user interface.

5.Application component.

- It is an top most and final level and layer of Architecture .
- any user can write and install ther own application in this layer only.
- It is an a established building blokes of application . This components are loosely cuppled by the manifest file .

Following are the component :-

Activies
services
Brodcast Reciver
Content provider

i.Activies :-

- This content Represent the single screen Ui.

- The Activity perform action on screen .
- this class implments the subclass of Activity class.

```
public class MainActivity extends Activity {
}
```

ii.services :-

- this service component running on background to perform the lengthy operations .
- This implements the subclass of Service class.

```
public class MyService extends Service {
}
```

iii.Brodcast Reciever :-

- Broadcast Reciever component simply response the broad cast messages from one application to another application .
- this implements the sub class of Broadcast Receiver class and each message are broadcast as intent object .

```
public voild MyReciver extends BroadcastReciver{
public void reciver(context,intent){
}
}
```

iv.Content provider :-

- This component supplies the data from one application to another application .
- Such request are handle by the method contentResolver .
- implements the subclass of ContentProvidr class .
- the data is stored in the any data base or file .

```
public void MyContentProvider extends ContentProvider{
public void onCreate(){
}
}
```

Additional component

- Fragments :- Represent the portion in activity .

- view :- represent the ui elements drawn on screen .
- layout :- view the hierarchy on control screen format .
- intent :- use to message worning component .
- Resources :- it is use to handle external elements like string ,color , picture.
- Manifest :- it is a configure file for application .

6.Anatomy of Android Application ?

Before you run your application you are known some directories and files in the projrect-

i. **.java** :- this file contain the file related to .java socrece files .that . java file contain an MyActivity.java file having an activity class that run when app is launched with icons

ii. **res/drawable-hdpi** :- that is the directory that for drawable object of high-density screen.

iii. **res/layout** :- this derictory file are use to define your application user interface.

iv. **res/values :-** this directory file are use to define other xml files that is a collection of the other resources like string and color defination s .

v.**Andriod Manifest.xml** :- this is a Manifest file that describes the fundamaental charactoristics about app and defines each components.

vi. **Bulid.gradle** :- this is an auto generated file that contain the compilesdk version , building tools version, application id, minsdk version , version code and version name .

7. Main Activity file ?

- The MainActivity code is an code of java file MainActivity.java
- This is actual application file which is ultimatelly get converted to a Dalvik exacutable and run application .

public MainActivity extends AppCompatActivity{
@override

```
protected void onCreate(BundlesavedInstancestate){
super.onCreate(savedInstanceState);
setContentView(R.Layout.activity_main);
}
}
```

- here. R.Layout.activity_main refers to Activity_main.xml file located in res/layout folder.
- the onCreate is a one method that figured when the activity is loaded.

8. Manifrest file ?

- What ever component you developed in your application you must declare it as a part of manifest.xml which is a root of the application project directory.
- This file is an interface between the Android os and your application .
- It is manditory to declare component in this file other wise os not consider the application .

```
 <? xml version-"1.0" ?>
<manifest:android="link" package="address">
<application
 android:allowBackup="true'
android:icon="hhh.kj"
android:lable="kasddyu">
<activityandroid :name=".MainActivity">
</activity>
</application>
</manifest>
```

- Following is an a list of tags which you will use in your manifest file to specify different component :-

a. <activity> element for Activities
b. <services> element for services
c. <receiver> element for broadcast receiver
d. <provider> element for content provider

9. Strings file ?

- This string.xml file are located in the res/value folder and contains the text that your folder is use.

```
<resources>
<stringname="app.name">hello</string>
 <stringname="menu_settings">settings</string>
</resources>
```

10. Layout file ?

- The activity_main.xml is a layout file available in res/layout/directory, that is referance by your application at time of building User interface.
- you can modify this file frequently by cahnging the layout of your application .

```
<linearlayoutxmls:android
android.layout : width="match_Parent"
android.layout : height="match_Parent"
>
<textView
android.layout : width="100sp"
android.layout : height="100sp"
>
</linearlayout>
```

11. Organize resource in Android Studio?

```
MyProject/
app/
manifest/
    AndroidManifest.xml
Java/
    MyActivity.java
res/
    drawable/
        icon.png
    layout/
        acrtivity_main.xml
        info.xml
    values/
        string.xml
```

UNIT-2

12.Alternative Resources.

- your applicatioon should provide alternative resources to support specific configuratiion .
- ex. we can include some alternative resource for drawable resource like images.
- Android detects the currrent devices configuration and loads the appropriate resources for your application.
- To specify configuration specific alternatives for a set of resources , follow the following steps.
- Create a new directory in res/named in form <resources_name>-<config_qualifier>. here the resource name is any resource mentioned in the above table,like layout . the qualifier will specify an individual configuration for which these resource are to be used.
- Save the respective alternative resource in new directory. this resource file name is exactly the same as the default resource file.

Example:-

```
MyProject/
app/
manifest/
        AndroidManifest.xml
java/
        MyActivity.java
res/
        drawable/
                icon.png
                background.png
        drawable-hdpi/
                icon.png
                background.png

        layout/
```

info.xml
value/
String.xml

13.Accessing Resources.

During your Application development you will need to access defined resource either in your code or in layout xml file.

i. Access resource in code :-

to access res/drawable/img.png or set MyImageView use should use following code .

```
1 ImageView imageview=(ImageView)findViewById(R.id.myimageview);
2 imageView.setImagResource(R.drawable.myimage);
```

here , first line code use od R.id.imageview to get imageview defined with id myimageview in layout file .
second line code make use of R.drawable.image to get image with name myimage available in drawable sub-directory under/res.

ii. Access resource in xml :-

consider the following resource XML res/valuestring.xml files that includes a color resourece and a string resource.

```
<? xml version="1.0" ?>
<resource>
<colorname="red">#f00</color>
<stringname="hello">hello</string>
</resource>
```

now use this resource file use in flayout file to set text color and text stirng.

```
< ? xml version="1.0">
<TextView
 android:textcolor="@color/red"
android:text="@string/hello"
>
```

14. Intent & filter

An android intent is an a abstract discreption of operation to b performed .it is used with startActivity to launch the activity .

Intent Types

1.Explicit :
2. Impplicit :

1.Explicit:

Explicit intent is going to connected internal world of application ,suppose if you want to connect one activity to another activity ,we can do this quote by explicit intent . This intents designate the terget component by the name and they are typically used for application -internal-message.such as activity starting a subordinate services or launching a sister activity.

```
Intent i=new Intent(FirstActivity.this,SecondActivity.class);
startingActivity(i);
```

2.Implicit:

This intent do not name a terget and the field for the component name is left back . Implicit intent are often use to activate component in the other application.

```
Intent readi=new Intent();
readi.setAction(androi.content.Intent.Action_view);

readi.setData(contentsContents.Content.CONTENT_URL);
startActivity(readi);
```

The target component which recives the intent can use the getExtras() method to get extra data sent by source component.

```
Bundle extras=getIntent().getExtras();
String val1=extras.getstring("key1");
String val2=extras.getstring("key2");
```

Intent Filter

- Android os use filter to pinpoint the set of Activities , Services and Brodcast Receiver that can handle intent with help of set of actions, categories, data schema accosiated with an intent .
- we can use <intent-filter> element in manifest.xml file to list out the action , catogries and data types accosiated with the Activity,Services and Brodcast receiver.
- <data> element specifies the data type expected by the activity to be called.
- There may be situation that an intent can pass through the filter of more than one activity or services the user may be ask which component will be activate .
- An exception is rised when no target is found

UNIT-3

15. Android layout type ?

The number of layout provide by android that you can use in your application.

i. Linear Layout :- in liear layout we can arrange the all display child view in single direction ,either vertically or horizintally .

ii. Relative layout :- in this layout we can arrange the all display child view in relative position .

iii. Table layout :- in this layout we can arrange the all display child view in form of rows and colums .

iv. List view :- it is a view group that display list of scrollable item.

v.Grid view :- it is a view group that display item in two dimension and scrollable grid. .

vi. Frame layout :- the frame layout is an place holder on screen that you use to display a single view.

vii. absolute layout :- it anable you to specify the exact location of its children.

16. Layout Attribute ?

The layout has a set of attributes which defines the visual properties of lauyout.

some of the attributes are shown below :-

i. android:id = this is id for the view to uniqualy identificcation .

ii. androd:layout_width = width of layout

iii. androd:layout_height = hight of layout

iv. androd:layout_marginTop = this is extra space from top side of layout

v. androd:layout_marginBottom = this is extra space from bottom side of layout

vi. androd:layout_marginleft = this is extra space from left side of layout

vii. androd:layout_marginRight = this is extra space from Right side of layout

viii. androd:layout_padding = this is padding field for the layout (top , bottom , left , right)

ix. androd:layout_gravity = this specifies how child views are positiion.

x. androd:layout_weight = this specifies how much of the extra spece in the layout should be allocate to the view .

xi. androd:layout_x = this specifies the x-cordination of layout.

xii. androd:layout_y = this specifies the y-cordination of layout.

View Identification

Te view object may have a unique ID assigned to it which will identify the view uniquely with in the tree. syntax for id in xml tag is ;

android:id="@+id/my_button"

- @ symbole indicates that xml parser should parse and expand the rest of the Id string and identify it as an ID resourse.

- The + symoble use after @ mens that this is a new name that must be created and added to our resourse .
- we can capture id by following syntax

  Button myButton=(Buttin)findViewById(R.id.my_button);

17. Android UI Controller ?

There are number of ui controller provided by android that allow you to build the graphical user interface for your app.

i. TextView :- This control is used to display text to the user .

ii. EditText :- EditText is a predfined subclass of TextViews that includes rich editing capabilites.

iii. Buttom :- A push-bottom that can be pressed,or clicked,by the user to perform an action.

iv. ImageButtom :- this is use to show the image with an Button.

v. checkBox :- this is on/off switch that can be togglled by the user . web can use checlk box to chosse one or more than one option for user .

vi.ToggleButton :- it is an on/off Button with light indictor.

vii. RadioButton :- Radio button has a two states either checked or unchecked.

viii Radiogroup :- Radio Group is use to group together one or more Radio buttons.

viii. spinner :- it is drop-down list which allow user to use one value from it.

ix. Timepicker :- enables users to select a time of the day.

x. DatePicker :- use to ensbles user to select the date of the day .

18. Android Event Handler ?

- Event is an a useful way to collect the data about a user interaction with interactive components of application.
- the android framework maintains an event queue as first in first out (fifo) base.
- We can capture that event in our program and take appropriate action as per requirement .

- Event listner :- an event listner is an interface in the view class that contains a single callback method. These method will be called by the Android framework when the view to which the listner has been register is toggled by user interaction with the item in the UI .

i. onClick() :- this is called when user either click or touch or focus upon any widget like button , text.

ii. onLongClick() :- this is called when user either click or focus or touch any widthget for one or more then two secod.

iii. onFocuschange() :- this is called when widthget looses the focus. user goes away from the view item

iv. onKey() :- This is called when user is focused on the item and press or released any hareware key.

v. onTouch() :- This is called when user press the key or released the key, or

any movement gesture on the screen .

vi. onMenuItemclick() :- This is called when user select a menu item .

vii. onCreateContextMenu() :- this is called when context menu is being built.

- Event listner Registration :- Event Registration is the process by which an Event Handler gets register with Event listner. so event handler is called when the listner is fare the event.

  1. using anonymous inner class.
  2. Activity class implements the listner interface.
  3. using layout file to specify event hanler directory.

- Event Handler :- When an event happens and we have registered an event listner for that event , the event listner call the Event Handler , which is the method that actually handles the event.

a. Touch Mode

User can interact with their device by using hardware key or button or touching the screen. touching the screen put the device in the touch mode . the user can interact with it by touching on -screen virtual buttons .touch mode is use by calling metod of view class isInTouchMode().

b. Focus.

The view or widget is highilighted or display a flashing curcor when its in focus.this is indecates that it is ready to accept the input from user .

isFocusable() = return true or false.
isFocusableInTouchMode() = checks if view is focusable in touch mode.

c. Style and themes.

The Style resource define the format and look for the UI.
The style can be applied for the individual view or to entire activity or

application.

The style is defined in the Xml resources that is separate from the xml that specifies the layout .

The xml file contaions the res/value/directory of your project and will have <resources> as root node which mannditory for the style file .The name of xml file is arbitrary, but you must use .xml extension

You can use multiple style per file using thre <style> attribute but each have ther own name and identfication number .

```
<? xml version ="1.0"   ?>
<resource>
<stylename="CustomFontStyle">
<itemname="android:textSize">12pt</item>
</style>
</resource>
```

use:

once style is defined , you can use it in your xml layout file using Style attribute.

```
<? xml version="1.0" ?>
< LinearLayoutxmls
android:layout_width="matcho_parent"
>
<TextView
android:id="@+id/text_id"

style="@style/CustomFontStyle"/>
</LinerLayout>
```

Style Inheritance:

Android is support the style inheritance as very much similer to cascading style sheet in web design.

we can use the inherite properties from an existing style and then define only the properties that you want to change or add.

Applying colours and theme Atteribute

color resource can then be applied to some theme attribute , such as the windows background and primarry text color by adding <item> element to your custom theme.

These attribute can define in your style.xml files.

```
<resources>
<stylename="MycustomTheme"..>
<itemname="android:windowsBackground">@color/my_custom_color
</item>
</style>
</resource>
```

Android Themes

- The theme is noting but the style applied to a entire activity or application, rather than an individual view.
- When style is applied to a theme , every view in the Activity or application will apply each style property that it support.
- To set the theme for all the activities of your application, open the AndroidManifest.xml file and edit the <applicatiion> tag to include android:theme attribute with the style name .
- But if you want to apply theme to only one activity then you can write android:theme attribute in <acticvity> tag only.
- Android provide number of default themes that can inherites by using parent attribute .

Default Style & Themes

- Android provide number of default themes that can we use in our application .
- we can find the referance of all available styles in R.style class.
- To use these styles in you applicaton replace all underscore in style name with period
- we can see the following source code for android style and theme .
  i. Android Style(style.xml)
  ii. Android Themes (themes.xml)

Custom Components

- Implementing own components in pre built in components with extending subclass with own defined class.
- Android offers a great list of pre-built widget like button, textview , EditText,Listview,checkbox. which can use directly in our application .but there is some situation when we arre not satisfied with existing functionality of any of the available widgets .
- Android provide us the mean we can create our own costum compoments which we can customize in our application as per our need.
- if we can make some change of small adjustment in the available weidgets so we can make subclass or layout or override this method whish can gives as a approprite control and function on screen element

view
1.image view
2.textview i. button
3.viewgroup i. linearlayout ii. framelayout iii. ....