**SAMPLE PROBLEM SET – ANDROID PROGRAMMING Jawad Khan**

# Objectives:

1. When we build and run an android app, the system launches an instance of Activity and loads its layout.
2. The XML that defines the layout for main activity’s UI is named as activity\_main.xml
3. \_AndroidManifest.xml\_ file describes the fundamental characteristics of the app and defines each of its components.
4. build.gradle file is used configure how the Gradle tools compile and build your app.
5. An Intent is an object that provides runtime binding between separate components, such as two activities.
6. The Intent constructor takes two parameters.
7. The first argument of Intent constructor is \_\_\_(a)\_\_\_\_\_ (a) Context, (b) Class
8. The second argument of intent constructor is \_(b)\_\_\_\_ (a) Context, (b) Class
9. The Activity class is a subclass of Context
10. The putExtra() method is used to add value to an Intent.
11. An Intent can carry data types as key-value pairs called extras.
12. The startActivity() method starts an instance of the Activity specified by the Intent
13. To add a custom color, we make entry in file named: colors.xml
14. To show a message box we use function: \_Toast.makeText
15. Suppose we have string with name my\_string and value “number of participants is %1d”. To insert the value in %1d, we use function \_this.getString(R.string.my\_string, 10)\_
16. \_android.app\_\_\_\_\_\_ android library provides access to the application model.
17. \_android.content\_\_\_\_\_\_\_ library facilitates content access, publishing and messaging between applications and application components.
18. android.database is used to access data published by content providers and includes SQLite database management classes.
19. android.text library is used to render and manipulate text on a device display.
20. android.view library provides the fundamental building blocks of application user interfaces.
21. android.widget provides a rich collection of pre-built user interface components such as buttons, labels, list views, layout managers, radio buttons etc.
22. Dalvik Virtual Machine is a kind of Java Virtual Machine specially designed and optimized for Android.
23. Activities dictate the UI and handle the user interaction to the smart phone screen.
24. Services handle background processing associated with an application.
25. Broadcast Receivers handle communication between Android OS and applications.
26. Content Providers handle data and database management issues.
27. A service is a component that runs in the background to perform long-running operations.
28. Broadcast Receivers simply respond to broadcast messages from other applications or from the system
29. A content provider component supplies data from one application to others on request
30. Fragments represents a portion of user interface in an Activity.
31. Views are UI elements that are drawn on-screen including buttons, lists forms etc.
32. External elements, such as strings, constants and drawable pictures are called resources
33. Manifest is the configuration file for the application.
34. The animations are accessed with R.anim class.
35. Image files like .png, .jpg, .gif or XML files that are compiled into bitmaps, state lists, shapes, animation drawable are saved in res/drawable/ folder
36. Images are accessed with R.drawable class
37. onCreate is the first callback and called when the activity is first created.
38. onStart is called when the activity becomes visible to the user.
39. To show a method in console of android studio we use function Log.d
40. A service is bound when an application component binds to it by calling bindService.
41. A broadcast receiver is implemented as a subclass of BroadcastReceiver class

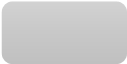
# Questions:

**Q. 1.** Write steps of adding a button in layout. The name of the button should be a string with name btn\_name and value CLICK HERE. Write code of string in strings file.

Q. 2. Suppose we have a layout like this:



Pakistan



CLICK

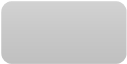


When the button is clicked, the value from above TextView should be written to the below TextView. You need to write the code for the event handler defined for button.

Q. 3. Suppose we have layout like this:



Pakistan



CLICK

When the button is clicked, the message in the TextView should display in a dialog. Write code for event handler of button.

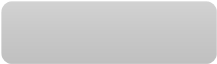
Q. 4. Write code that starts a new activity named DisplayActivity from MainActivity. Send two numbers from MainActivity to DisplayActivity. The two numbers should be declared as static variables.

Q. 5. Write code that sends two strings from MainActivity to DisplayActivity. In DisplayActivity, the strings should be concatenated and assigned to a TextView in DisplayActivity.

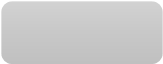
Q. 6. Suppose we have a layout like this



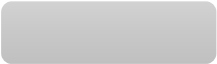
0



COUNTER



SEND



RANDOM



The value is 33

Write code for button RANDOM such that when user click button, a random number from one to hundred is shown in text box. Write code for button COUNTER such that when the user click the button, the value in the TextView start incrementing. Write code for button SEND such that when user click on button, the value in TextView is passed to a new activity (DisplayActivity) shown in the right window above. For the TextView in the right window, you need to define a string “The value is %1d” and assign to TextView.

1. 7. Suppose that you have a string defined in strings.xml. The name of the string is country\_name and value is “Pakistan”. Write a program that shows the value of string in a dialog box.

Q.8. Suppose you have two TextViews, each containing a number, and a button to add the values of the two TextViews. When the button is clicked, the values of the TextViews are added and result should be shown in console. (Hint: Log.d)

Q. 9. Suppose you have an EditText field and two buttons. The first button is labeled as BLUE and the second button is labeled as GREEN. When the BLUE button is clicked, the color of text in EditText should changed to BLUE, and when GREEN button is clicked, the color of text in EditText should change to GREEN. Write also the code of defining the two colors in colors.xml file.

Q. 10. Write a program that overrides the onStart() and onStop() events of activities. In onStart event, the message should show in console “This is onStart() event”. In onStop event, the message should show in console “This is onStop() event”.

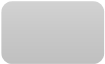
Q. 11. Create a Service class named HelloService. In this class define the two methods to start and stop the service. The prototypes of the methods are: public int onStartCommand(Intent intent, int flags, int startId) and public void onDestroy()

Q. 12. Suppose you want to define a broadcast receive for the message android.intent.action.BOOT\_COMPLETED. Write code to register this message in AndroidManifest.xml. Extend a class MyReceiver from BroadReceiver and define onReceive method in that class. Show the message in a dialog window in onReceive method. The prototype of onReceive method is public void onReceive(Context context, Intent intent);

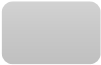
Q. 13. The following layout has three number buttons, a plus and equal operator, and a EditText initialized with a zero “0” string.



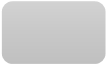
0



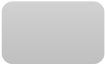
2



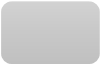
+



1



3



=

The user should be able to enter a string of numbers like 12232213. The user need to enter a number click on + operator and then input another number. When user click on equal, the result of sum should display in EditText.