MAD Question and Answers….

1. What is OHA?

Ans:

1. Open Handset Alliance (OHA)
2. It is formed on 10 October 2007.
3. Created for purpose of developing open mobile device standards.
4. It has approx. 84 member companies, including HTC, Dell, Intel, Motorola, and Google
5. Its main product is Android Platform
6. OHA members are primarily mobile operators, handset manufacturers, software development firms, semiconductor companies and commercialization companies.
7. Members share a commitment to expanding the commercial viability of open platform development.

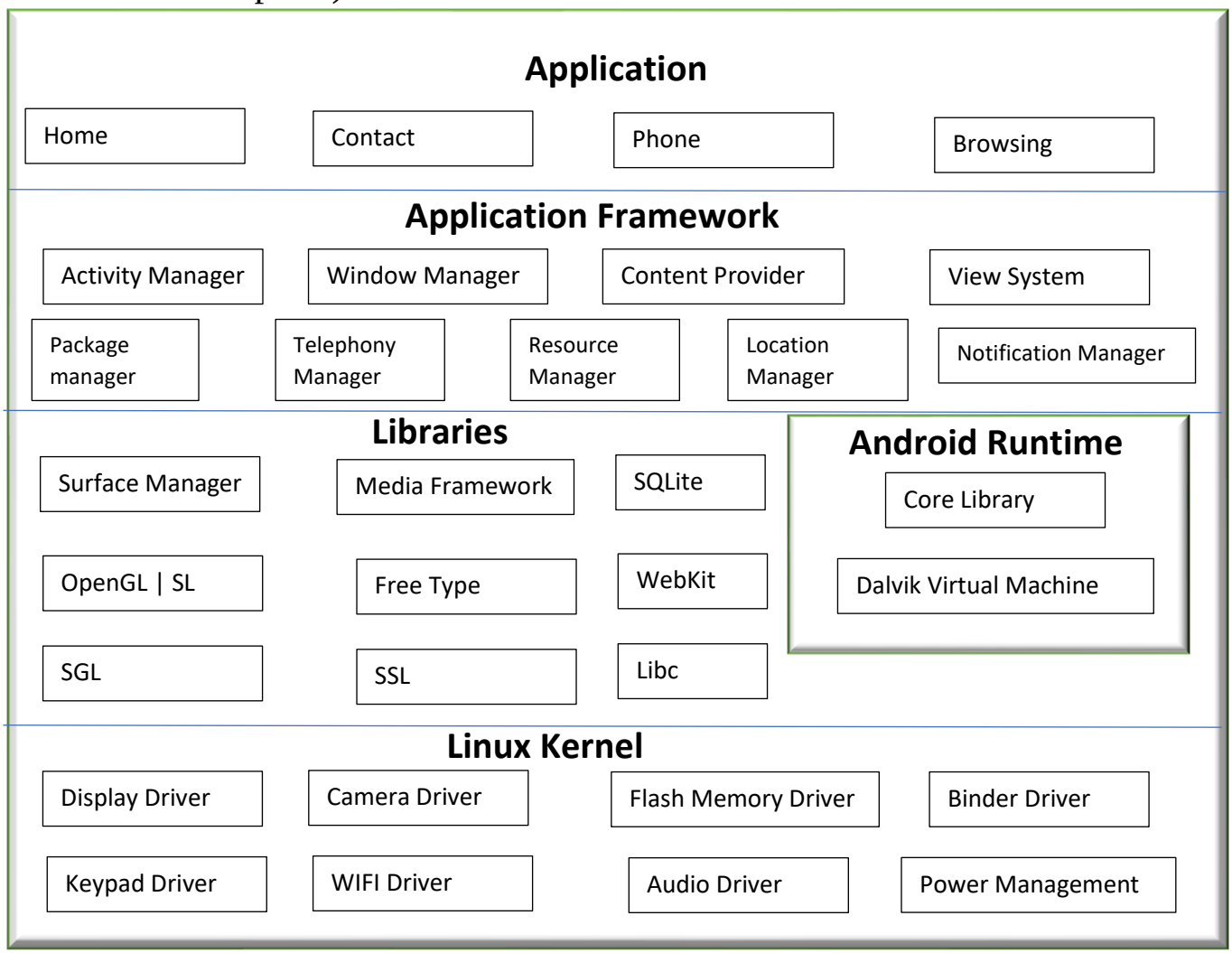
2. Feature of Android? (Enlist for 4 marks) (explain for 8 marks)

Ans:

Sr.No.	Feature & Description
1	**Beautiful UI** Android OS basic screen provides a beautiful and intuitive user interface.
2	**Connectivity** GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, NFC, and WiMAX.
3	**Storage** SQLite, a lightweight relational database, is used for data storage purposes.

4	**Media support** H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC, AAC 5.1, MP3, MIDI, Ogg Orbis, WAV, JPEG, PNG, GIF, and BMP.
5	**Messaging** SMS and MMS
6	**Web browser** Based on the open-source Web Kit layout engine, coupled with Chrome's V8 JavaScript engine supporting HTML5 and CSS3.
7	**Multi-touch** Android has native support for multi-touch which was initially made available in handsets such as the HTC Hero.
8	**Multi-tasking** User can jump from one task to another and same time various application can run simultaneously.

3. Draw and Explain Android Architecture (Draw for 2 marks, for 4/8 marks explain)



- Android architecture contains different components which supports any android device need.
- It contains open-source Linux Kernel which contains collection of number of C/C++ libraries
- This Kernel Provides DVM to all android device to provide platform for running android applications
- Following are main components of android architecture:
    - Application: (Top Level of Architecture)
    - Application framework: (Contain important classes to create android application)
    - Libraries: (includes libraries contains C/C++ core libraries and java-based libraries)

- Android Runtime: (contains DVM helps to run and create multiple instances of applications)
- Linux Kernel: (Heart of Android Architecture. It manages all available drivers which required during runtime)

4. Difference Between android and Windows.

Android OS	**Windows OS**
Developed and owned by Google LLC.	Developed and Owned by Windows Incorporation
Launch in 2008	Launch in 1985
Current Stable version Android 11	Current Stable version Windows 11
Kernel is Linux Based	Kernel is hybrid with Modules here
It doesn’t charge for any version.	It charges for its original version
Most used Operating System overall	Most used Operating System in personal Computers
Specially designed for Mobile Devices	Specially designed for PC of all Companies
Its target device is smartphones, and tablet computer.	It is for workstation, personal computer, media centre, tablets, and embedded system

5. Tools and Software required for developing android application.

Ans:

1. Windows 7/8/10(64 bit)
2. 4 GB RAM minimum, 8 GB RAM recommended
3. 2 GB available space on disk, 4 GB recommended.
4. 1280 x 800 recommended screen resolution.
5. Java 7 JDK.
6. Gradle or Apache ANT.
7. Android studio/ IntelliJ Idea/ Eclipse IDE and Android SDK.

6. Difference between JVM and DVM.

Ans:

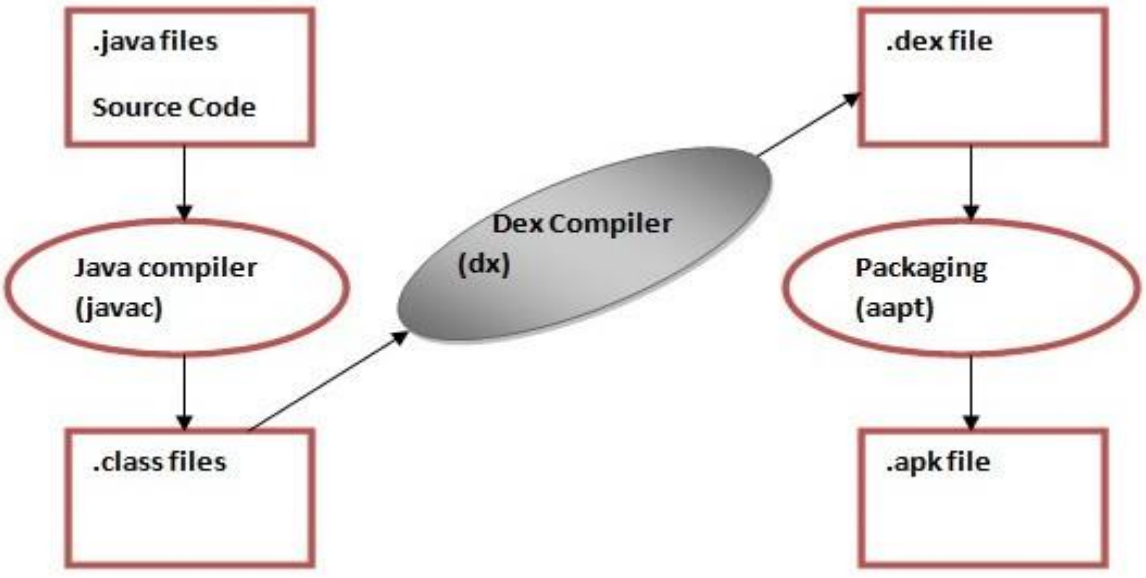
JVM (Java Virtual Machine)	**DVM (Dalvik Virtual Machine)**
Stack Based Virtual Machine	Register Based Virtual Machine
Java source code first convert into bytecode(.class) then convert into machine code	It first converts all source code file into bytecode(.class) then it converts all bytecode file into Dalvik bytecode (classes.dex) then it creates an .apk file
More information required for Data Loading and Manipulating, as well as methods loading.	Instruction size is larger as it need to encode source code and use designation register.
Executable file for the device is .jar file	Executable file for the devices is .apk file
Compiled bytecode size is compact.	Compiled bytecode size is larger

Support multiple operating systems	Supports android operating system.

7. Explain DVM with Diagram (for 4 marks only explain. For 8 marks draw Diagram).

Ans:

1. DVM is a Dalvik Virtual Machine
2. It is used in android programming to create .apk file from given source code.
3. DVM is a Register based Virtual Machine.
4. That's why its instructions size is larger as compare to JVM.
5. It can compiled all source code files and convert it into bytecode (.class) file as like JVM.
6. Then DEX Converter (dx tool) convert bytecode in to Dalvik bytecode (classes.dex)
7. Then this file is use to create .apk file
8. It only works with Android Operating System.



8. Define Emulator.

Ans:

1. The Android emulator is an Android Virtual Device (AVD)
2. which represents a specific Android device.
3. We can use the Android emulator as a target device to execute and test our Android application on our PC.
4. The Android emulator provides almost all the functionality of a real device. We can get the incoming phone calls and text messages.
5. It also gives the location of the device and simulates different network speeds.
6. Android emulator simulates rotation and other hardware sensors. It accesses the Google Play store, and much more

9. Explain ADT.

Ans:

1. It's a plugin for Eclipse IDE that is designed to give you powerful, integrated environment in which to build Android Application.

10. Explain AVD (steps for 4 marks).

Ans:

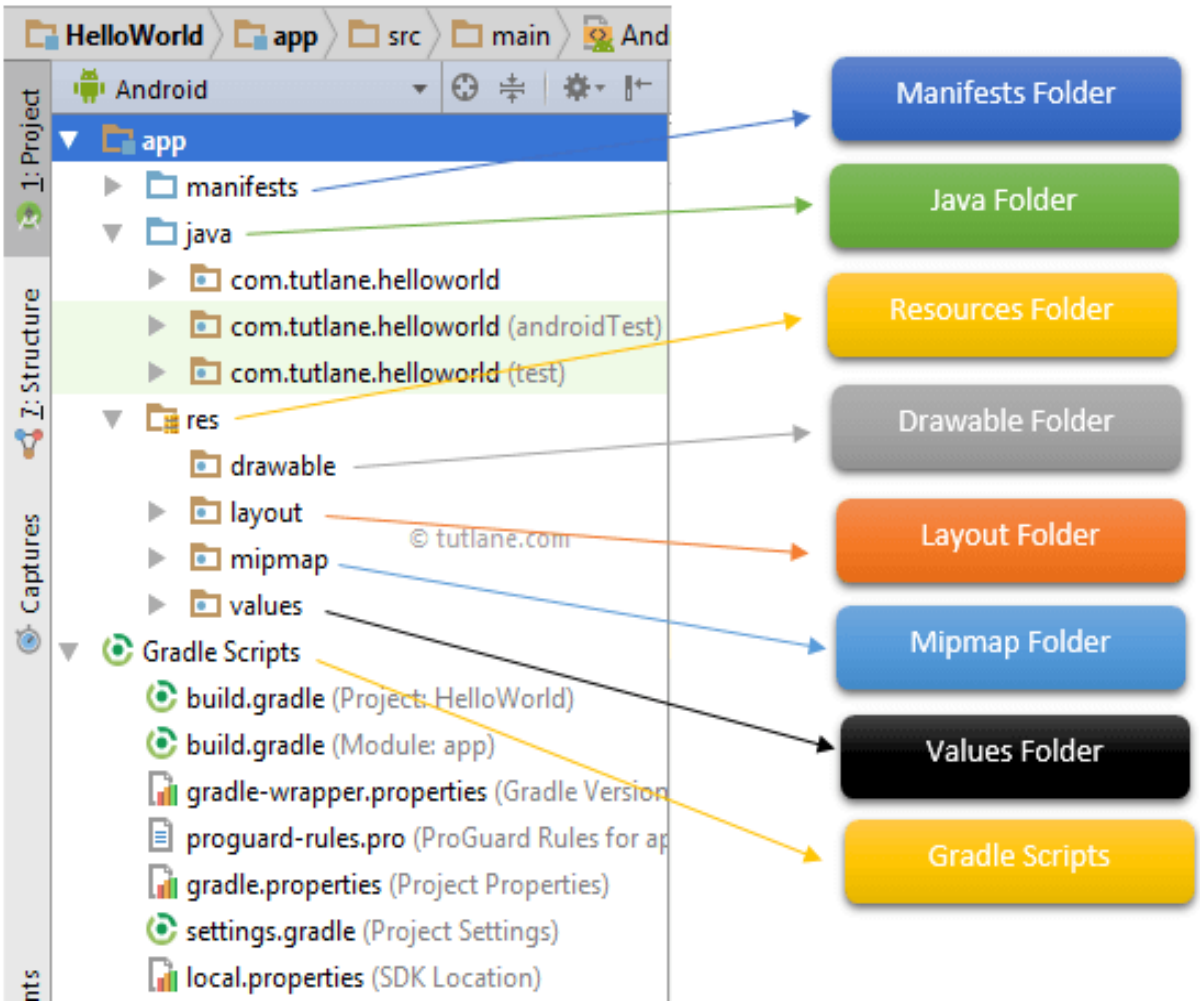
1. AVD stands for Android Virtual Device.
2. It's a configuration that defines the characteristics of an android phone, tablet, Wear OS, Android TV, Automotive OS device that you want to simulate in android emulator.
3. Steps:
    a. Click on No device option. From the ribbon.
    b. Then clock on create device from drop down list.
    c. Then click on Create Virtual Device from dialogue box.
    d. Then select the device type from left panel then select android OS version on that device.
    e. Then installation of that version starts.
    f. After complete installation click on next.
    g. Then set our device's name, orientation and some other settings then click finish.
    h. Then our virtual device AVD had created successfully.
    i. The name of our AVD then displayed on the place of NO DEVICE.

11. Explain ADT Plugins.

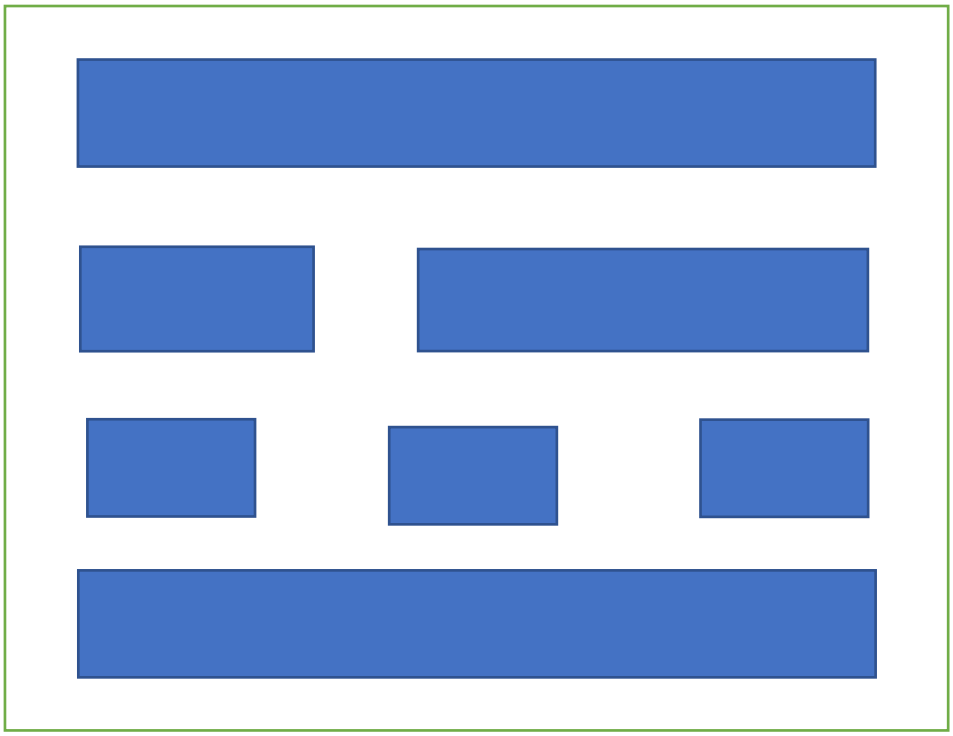
Ans:

1. Download Android SDK.
2. Install Android SDK
3. Open Android SDK Manager.
4. Install Android Version and Extra for SDK
5. Download and Install Eclipse.
6. Run Eclipse.
7. Add ADT Plugin Repository.
8. Install ADT Plugin.
9. Access ADT Plugin preference.
10. Setup ADT Plugin.
11. Access ADT Android Virtual Device Manager.
12. Add ADT Virtual Device.
13. Now we can program and run our first application.

12. Draw Hierarchical Directory Structure of android program



13. Write XML code for following output.



```
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
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:background="#F8F5F5"
    tools:context=".MainActivity">

    <TableRow
        android:id="@+id/tableRow1"
        android:layout_width="match_parent"
        android:layout_height="69dp"
        android:background="#F8F5F5"
        android:textAlignment="textStart"
        android:visibility="visible"
        tools:visibility="visible">

        <Button
            android:id="@+id/button1"
            android:layout_width="match_parent"
```

```
        android:layout_height="match_parent"

        android:text="Row1" />
    </TableRow>

    <TableRow
        android:id="@+id/tableRow2"
        android:layout_width="168dp"
        android:layout_height="69dp"
        android:layout_marginTop="100dp"
        android:background="#EFF0F4"
        android:visibility="visible"
        tools:visibility="visible">

        <Button
            android:id="@+id/button4"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:text="Row 2 Col 1" />
    </TableRow>

    <TableRow
        android:layout_width="229dp"
        android:layout_height="69dp"
        android:layout_marginStart="180dp"
        android:layout_marginTop="100dp"
        android:background="#FFFFFF"
        android:visibility="visible"
        tools:visibility="visible">

        <Button
            android:id="@+id/button3"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:text="Row 2 Col 2" />
    </TableRow>

    <TableRow
        android:layout_width="93dp"
        android:layout_height="69dp"
        android:layout_marginStart="10dp"
        android:layout_marginTop="200dp"
        android:background="#F9F9FA"
```

```
    android:visibility="visible"
    tools:visibility="visible">

    <Button
        android:id="@+id/button5"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="Row 3 Col 2" />

</TableRow>

<TableRow
    android:layout_width="93dp"
    android:layout_height="69dp"
    android:layout_marginStart="160dp"
    android:layout_marginTop="200dp"
    android:background="#F1F2F6"
    android:visibility="visible"
    tools:visibility="visible">

    <Button
        android:id="@+id/button6"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="Row 3 Col 2" />
</TableRow>

<TableRow
    android:layout_width="93dp"
    android:layout_height="69dp"
    android:layout_marginStart="310dp"
    android:layout_marginTop="200dp"
    android:background="#EFF0F6"
    android:visibility="visible"
    tools:visibility="visible">

    <Button
        android:id="@+id/button7"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="Row 3 Col 3" />
</TableRow>
```

```
    <TableRow

        android:layout_width="match_parent"
        android:layout_height="69dp"
        android:layout_marginTop="300dp"
        android:background="#F0F1F4"
        android:visibility="visible"
        tools:visibility="visible">

        <Button
            android:id="@+id/button8"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:text="Row 4" />
    </TableRow>


</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

14. Develop an Android Application to place Name, age, Mobile on screen using Absolute Layout.

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
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:background="#F8F5F5"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView2"
        android:layout_width="129dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="30dp"
        android:layout_marginTop="50dp"
        android:layout_padding="40dp"
        android:text="Name:"
        android:textSize="35dp" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="129dp"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:layout_marginTop="100dp"
        android:layout_padding="40dp"
        android:text="Age:"
        android:textSize="35dp" />

    <TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="30dp"
        android:layout_marginTop="150dp"
        android:layout_padding="40dp"
        android:text="Phone No:"
        android:textSize="35dp" />
```

```
    <EditText
        android:id="@+id/editTextTextPersonName"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="200dp"
        android:layout_marginTop="50dp"
        android:ems="10"
        android:inputType="textPersonName"
        android:textSize="20dp"
        android:text="Name" />

    <EditText
        android:id="@+id/editTextTextPersonName2"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="200dp"
        android:layout_marginTop="100dp"
        android:ems="10"
        android:inputType="number"
        android:textSize="20dp"
        android:text="Age" />

    <EditText
        android:id="@+id/editTextTextPersonName3"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="200dp"
        android:layout_marginTop="150dp"
        android:ems="10"
        android:inputType="phone"
        android:textSize="20dp"
        android:text="phone No" />

</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

15. Develop App to write Hello World.

Ans:

1. Manifest file:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.tutorialspoint7.myapplication">

   <application
      android:allowBackup="true"
      android:icon="@mipmap/ic_launcher"
      android:label="@string/app_name"
      android:supportsRtl="true"
      android:theme="@style/AppTheme">

      <activity android:name=".MainActivity">
         <intent-filter>
            <action android:name="android.intent.action.MAIN"
/>
            <category
android:name="android.intent.category.LAUNCHER" />
         </intent-filter>
      </activity>
   </application>
</manifest>
```

2. Java File.

```
package com.example.helloworld;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
   }
}
```

3. Layout file.

```
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent" >

   <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_centerHorizontal="true"
      android:layout_centerVertical="true"
      android:padding="@dimen/padding_medium"
      android:text="@string/hello_world"
      tools:context=".MainActivity" />

</RelativeLayout>
```

16. Enlist Different Android Layout.

Ans:

1. **Linear Layout:** it's a view group that align all children in a single direction vertically or horizontally.
2. **Relative Layout:** is a view group that displays child view in relative positions.
3. **Table Layout:** is a view group that allows place children in row and column.
4. **Absolute Layout:** enables you to specify the exact location of children.
5. **Frame Layout:** is a place holder on a screen that you can use to display single view

17. Explain any 4 UI components of Android.

Ans:

Sr.No.	UI Control & Description
1	TextView This control is used to display text to the user.
2	EditText EditText is a predefined subclass of TextView that includes rich editing capabilities.
3	AutoCompleteTextView The AutoCompleteTextView is a view that is similar to EditText, except that it shows a list of completion suggestions automatically while the user is typing.
4	Button A push-button that can be pressed, or clicked, by the user to perform an action.
5	ImageButton An ImageButton is an AbsoluteLayout which enables you to specify the exact location of its children. This shows a button with an image (instead of text) that can be pressed or clicked by the user.
6	CheckBox An on/off switch that can be toggled by the user. You should use check box when presenting users with a group of selectable options that are not mutually exclusive.

7	ToggleButton An on/off button with a light indicator.
8	RadioButton The RadioButton has two states: either checked or unchecked.
9	RadioGroup A RadioGroup is used to group together one or more RadioButtons.
10	ProgressBar The ProgressBar view provides visual feedback about some ongoing tasks, such as when you are performing a task in the background.
11	Spinner A drop-down list that allows users to select one value from a set.
12	TimePicker The TimePicker view enables users to select a time of the day, in either 24-hour mode or AM/PM mode.
13	DatePicker The DatePicker view enables users to select a date of the day.

.