1. What is View and ViewGroup? Discuss your answer using examples.

All user interface elements in an Android app are built using View and ViewGroup objects.

**View**:

A View is an object that draws something on the screen that the user can interact with. View objects are the basic building blocks of User Interface(UI) elements in Android

For examples are EditText, Button, CheckBox etc.

**ViewGroup**:

A ViewGroup is an object that holds other View (and ViewGroup ) objects in order to define the layout of the interface. The view group is the base class for layouts and views containers.

For example, LinearLayout is the ViewGroup that contains Button(View), and other Layouts also.

1. What is LinearLayout? What is the function of android:layout\_weight and android:layout\_gravity attribute?

**LinearLayout**:

It is a view group that aligns all children in a single direction, vertically or horizontally. You can specify the layout direction with the android:orientation attribute.All children of a [LinearLayout](https://developer.android.com/reference/android/widget/LinearLayout.html) are stacked one after the other, so a vertical list will only have one child per row and a horizontal list will only be one row high.

**android:layout\_weight:**

This attribute assigns an "importance" value to a view, and allows it to expand to fill any remaining space in the parent view. Views' default weight is zero. you can specify a size ratio between multiple views

**android:layout\_gravity:**

Gravity specifies how a component should be placed in its group of cells. The default is [TOP](https://developer.android.com/reference/android/view/Gravity.html#TOP).

1. Define constrained Layout. Design the “Reminder” Lab layout using ConstrainedLayout; add XML coding and screen shot.
2. Describe difference between ListView and RecyclerView. Discuss their pros and cons over one another.

There are two methods to implement a scrolling list:

|  |  |
| --- | --- |
| **ListView** | **RecyclerView** |
| simple usage. it’s the foundation of the ExpandableListView | integrated animations for adding, updating and removing items this adds complexity |
| doesn’t embrace the usage of the ViewHolder pattern | enforces the recycling of views by using the ViewHolder pattern |
| available OnItemClickListener | no OnItemClickListener |
| There is no need of extending other classes or implementing interfaces | In RecyclerView, you have no other option, but to implement the View Holder pattern |
| default adapters. the only way to handle ListView is through the ListView object or inside the adapter. | It’s more efficient by default, the layout is separated and we have more possibilities over the data set inside the adapter. |
| ListView gives you an option to add divider using dividerHeightparameter | RecyclerView enable you to customize divider (spacing) between two elements using RecyclerView.ItemDecorationclass. |
| ListView is a good old widget which has been included in the Android SDK API 1. Until Android Lollipop’s release. | it was introduced with the Android Lollipop and it proved to be a game changer |

1. How spinner and list view are similar? Write proper coding of using static and dynamic spinner in any activity.

Both have a data source (or adapter). It appears that Spinner can be shown in drop-down form, while ListView can show all data on the view directly. The data for both are all from an ArrayList. They could be in String type like this:

<string-array name="Cities">

<item >Beijing</item>

<item >Tokoy</item>

<item >New York</item>

<item >London</item>

<item >Paris</item>

<item >Berlin</item>

<item >Moscow</item>

<item >Hongkong</item>

</string-array>