TECHNOLOGY FOR APPLICATION DEVELOPMENT

By bhavana Singh
Assistant Manager

AGENDA

- ★ Introduction
- ★ App Development
- ★ Types of Application
- ★ Technologies
- ★ Android Development
- ★ Components of Android Studio
- ★ UI Component
- ★ Attributes
- ★ Flutter
- ★ Conclusion

App Development

APP Development :

App development is the process in which developers create an application to be used on smartphones, tablets and other mobile devices. While app development sometimes involves creating a web-based app or a desktop version of the app, the majority of app development projects are deployed to mobile and tablet devices.

Steps Of App development:

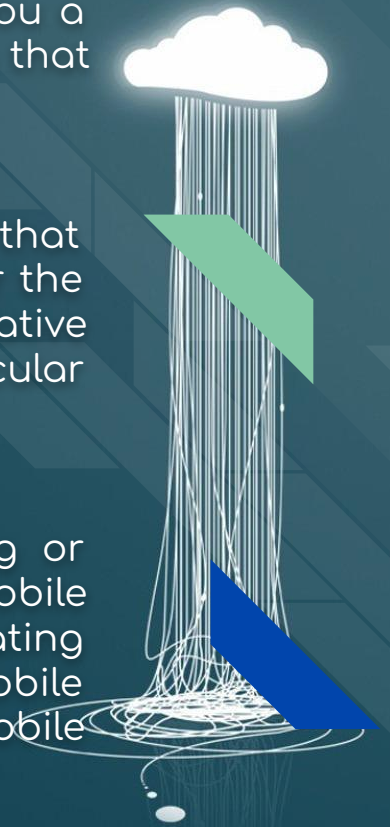
- ★ Data gathering
- ★ Data Analysis
- ★ Designing
- ★ Development
- ★ Testing
- ★ Deployment

Types of App Development

Desktop Application Development:
Desktop Application Development Company can provide you a user-friendly, efficient, and customized desktop application that can run offline, independent of your web browser.

Web Application Development:
Web application development is creation of app programs that reside on remote computers and are delivered to user over the Internet. These apps are sometimes contrasted with native applications that are developed specifically for a particular platform or device and are then installed on it.

Mobile Application Development:
Mobile application development is a process of creating or making a mobile application to run on different mobile platforms. Actually, a mobile platform is a mobile operating system. Google mobile platform is Android, and Apple's mobile operating system is called iOS. Windows has Windows mobile and RIM has Blackberry.



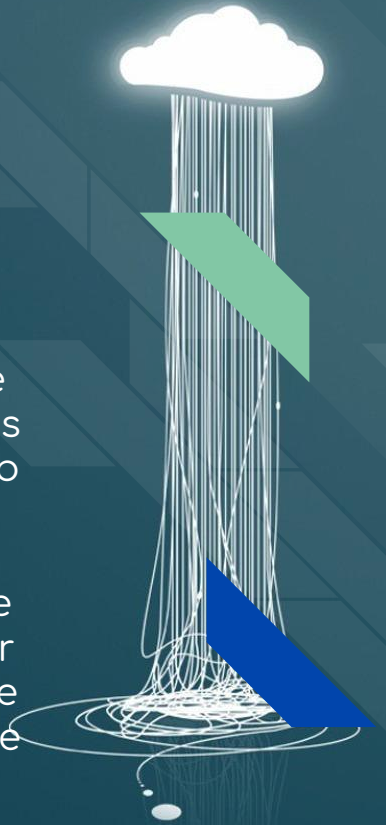
Technologies:

Android Studio :
Android Studio is the official Integrated Development Environment (IDE) for Android app development

Flutter :
Flutter is an open source framework by Google for building beautiful, natively compiled, multi-platform applications from a single codebase

Android Application Development

Android Development is the process of developing or creating computer program or set of programs to perform different task in Android Operating System



Types of Android app :

- ★ **Hybrid Application:** A (hybrid app) is a software application that combines elements of both native apps and web applications. Hybrid apps are essentially web apps that have been put in a native app shell.

- ★ **Native Application**: A native application is a software program that is developed for use on a particular platform or device. Because a native app is built for use on a particular device and its OS, it has the ability to use device-specific hardware and software.

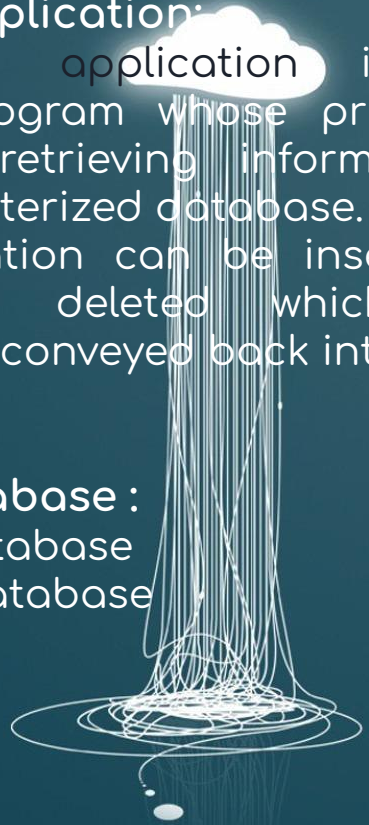
Native Application:

API Based Application :
API is the acronym for Application Programming Interface, which is a software intermediary that allows two applications to talk to each other.
For example: We are connecting our application through DB using API . API is a URL on which we are sending the request for inserting , updating , deleting and getting the data from database

Types of API Request :
- ★ POST
- ★ GET
- ★ PUT
- ★ DELETE

DB Based Application:
A database application is a computer program whose primary purpose is retrieving information from a computerized database. From here, information can be inserted, modified or deleted which is subsequently conveyed back into the database.

Types of database :
- ★ Local Database
- ★ Global Database

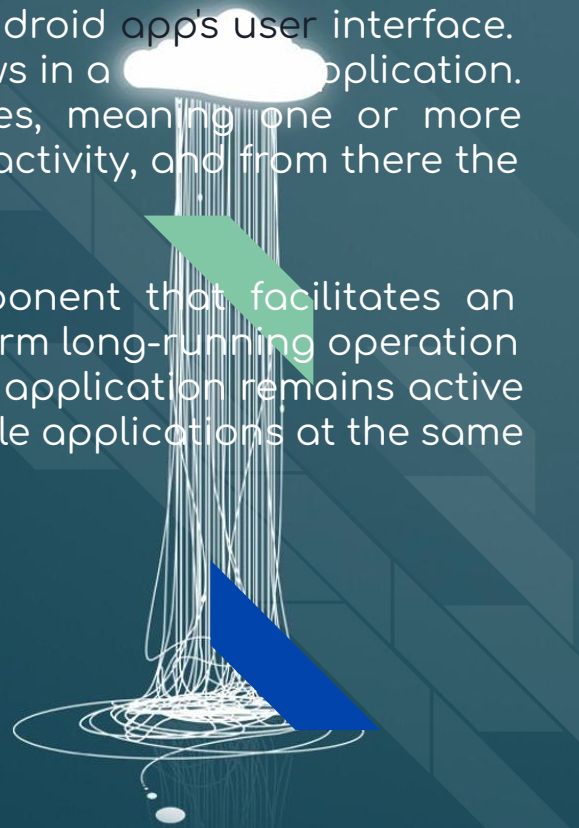


Components Of android Studio-1

- **Activities** : An Android activity is one screen of the Android app's user interface. In that way an Android activity is very similar to windows in a [illegible]pplication. An Android app may contain one or more activities, meaning one or more screens. The Android app starts by showing the main activity, and from there the app may make it possible to open additional activities.

- **Services** : Services in Android are a special component that facilitates an application to run in the background in order to perform long-running operation tasks. The prime aim of a service is to ensure that the application remains active in the background so that the user can operate multiple applications at the same time.

  Types of Services:
  - ★ Background Service
  - ★ Foreground Service
  - ★ Bound Service



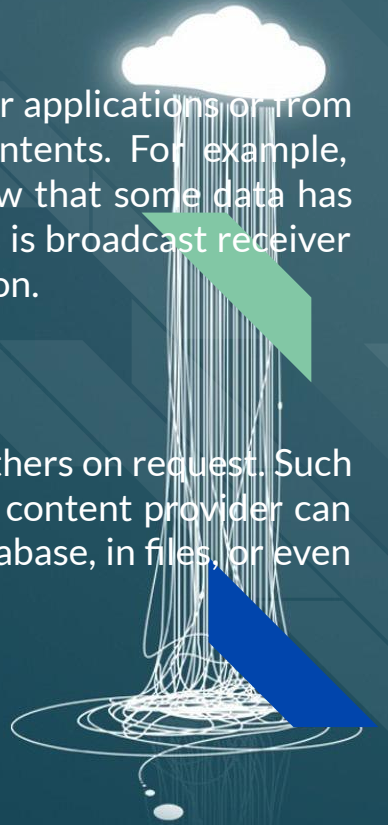
Components Of android Studio-2

Broadcast Receiver :

Broadcast Receivers simply respond to broadcast messages from other applications or from the system itself. These messages are sometime called events or intents. For example, applications can also initiate broadcasts to let other applications know that some data has been downloaded to the device and is available for them to use, so this is broadcast receiver who will intercept this communication and will initiate appropriate action.

Content Provider:

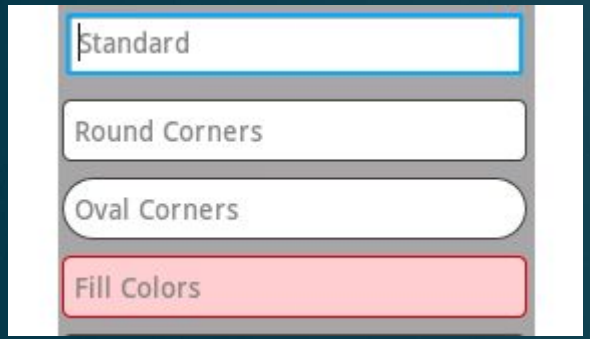
A content provider component supplies data from one application to others on request. Such requests are handled by the methods of the ContentResolver class. A content provider can use different ways to store its data and the data can be stored in a database, in files, or even over a network.

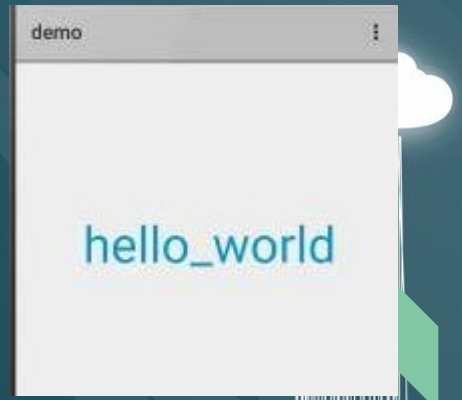


UI Components Of android Studio:

Textview:

A TextView displays text to the user and optionally allows them to edit it. A TextView is a complete text editor, however the basic class is configured to not allow editing.



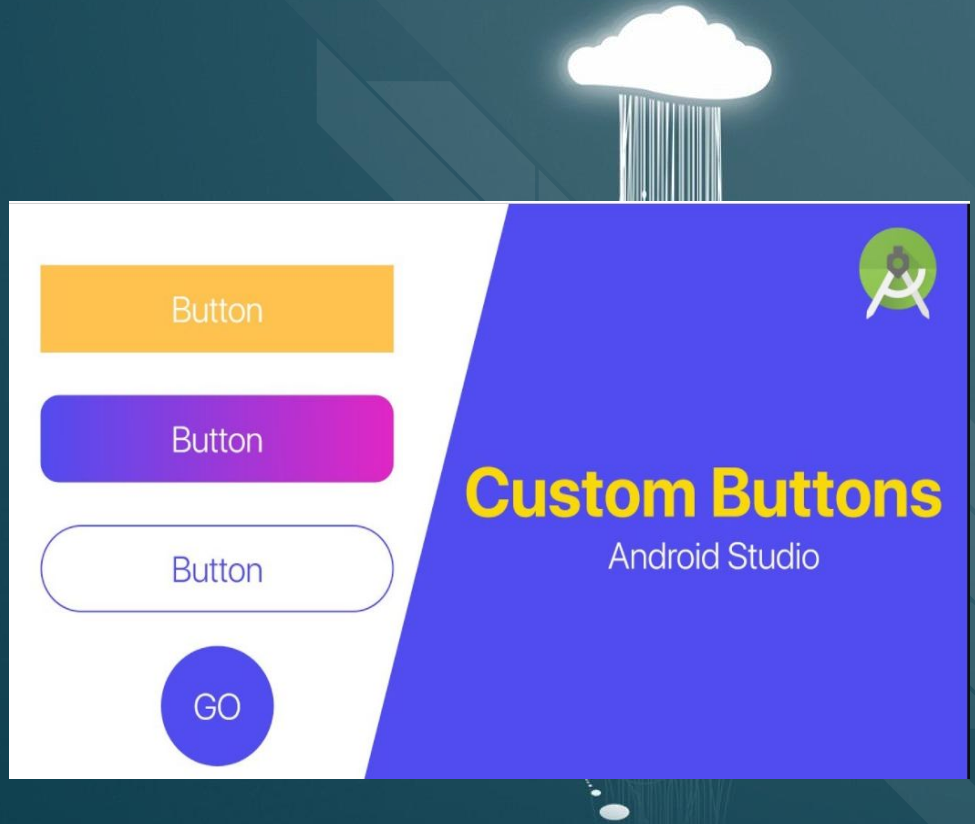


EditText:

A EditText is an overlay over TextView that configures itself to be editable. It is the predefined subclass of TextView that includes rich editing capabilities.

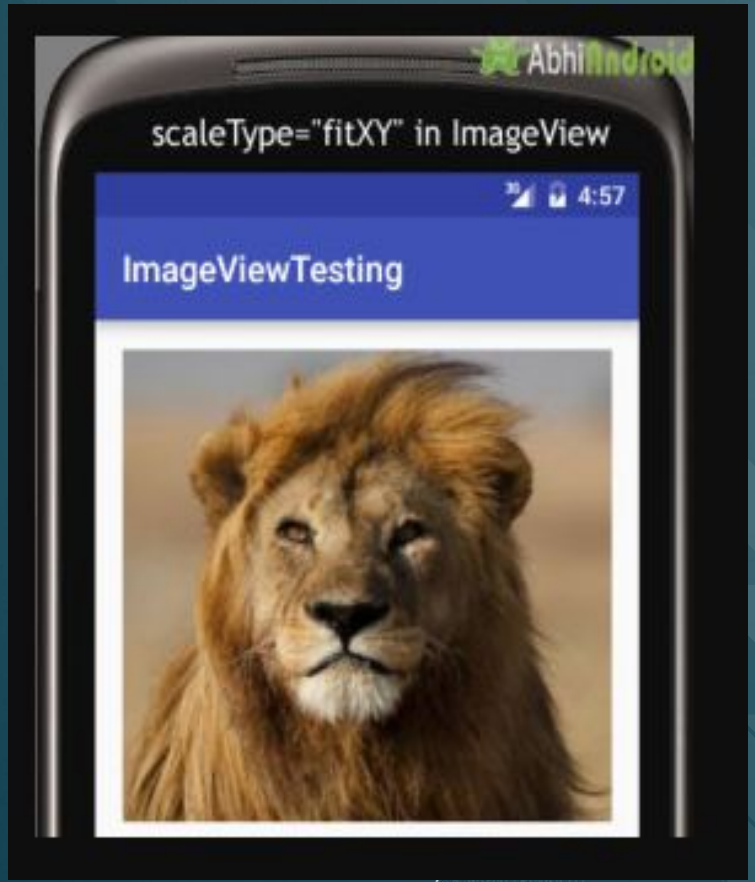
Button:

Android Button represents a push-button. In Android applications, a Button is a user interface that is used to perform some action when clicked or tapped. It is a very common widget in Android and developers often use it.



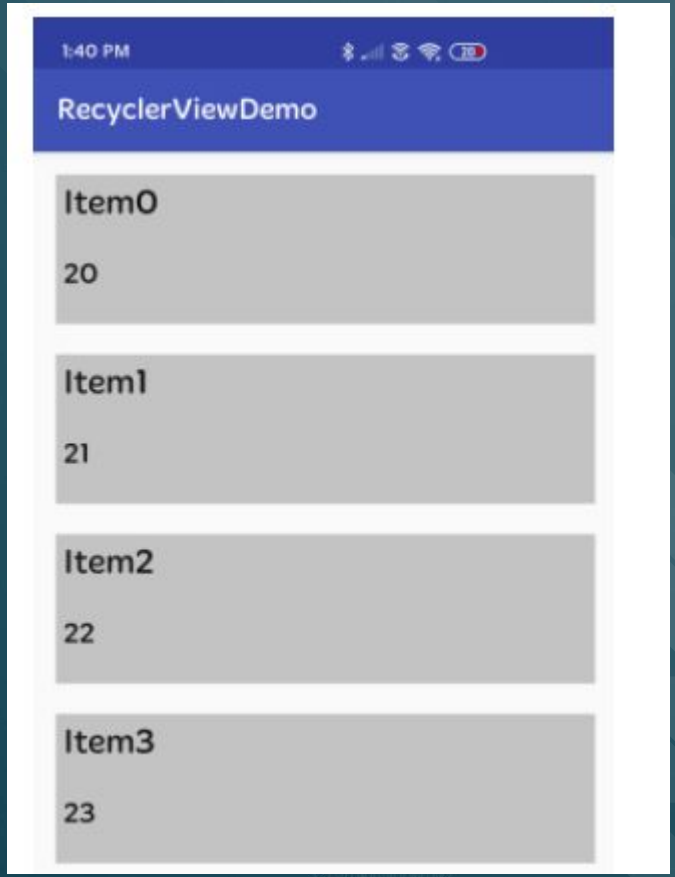
ImageView:

ImageView class is used to display any kind of image resource in the android application either it can be android an Android, ImageView class is used to display an image file in application. Image file is easy to use but hard to master in Android, because of the various screen sizes in Android devices. An android is enriched with some of the best UI design widgets that allows us to build good looking and attractive UI based application.



RecyclerView:

RecyclerView is the ViewGroup that contains the views corresponding to your data. It's a view itself, so you add RecyclerView into your layout the way you would add any other UI element. Each individual element in the list is defined by a view holder object.

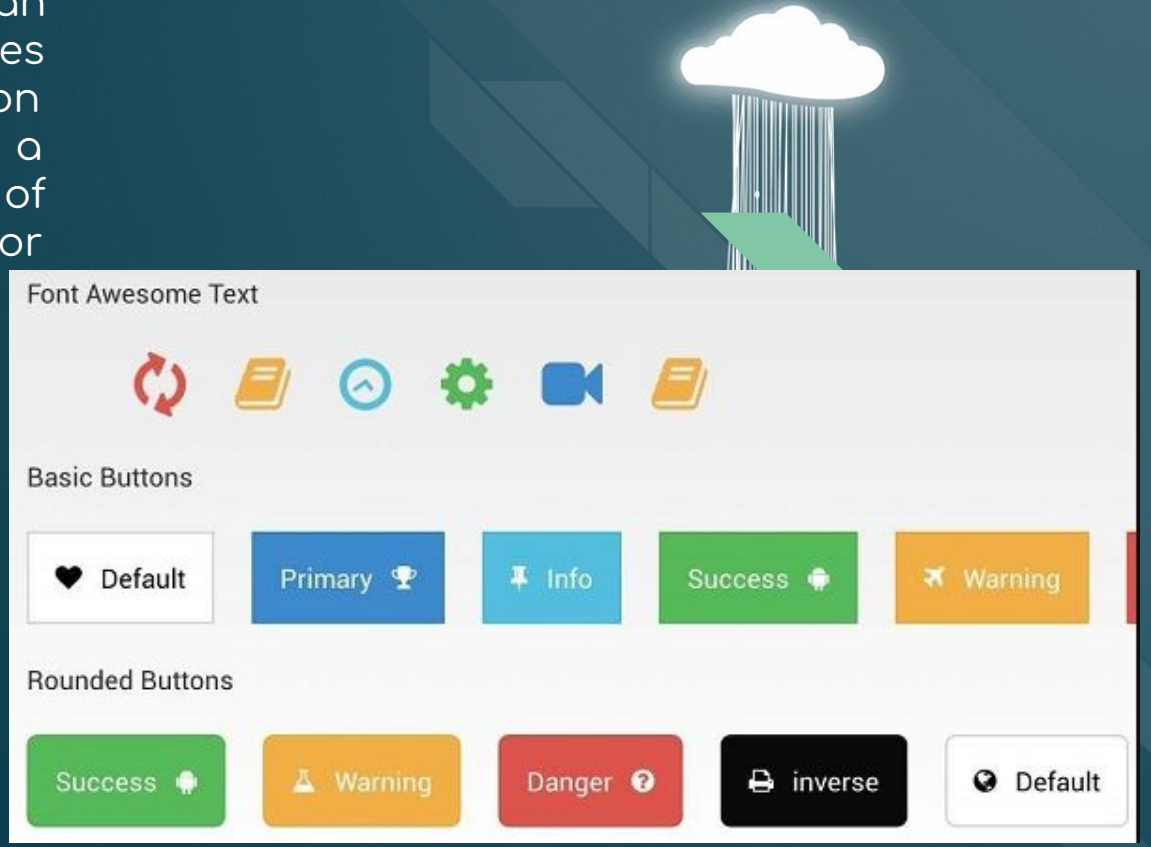


ScrollView :

In Android, a ScrollView is a view group that is used to make vertically scrollable views. A scroll view contains a single direct child only. In order to place multiple views in the scroll view, one needs to make a view group(like LinearLayout) as a direct child and then we can define many views inside it. A ScrollView supports Vertical scrolling only, so in order to create a horizontally scrollable view, HorizontalScrollView is used.

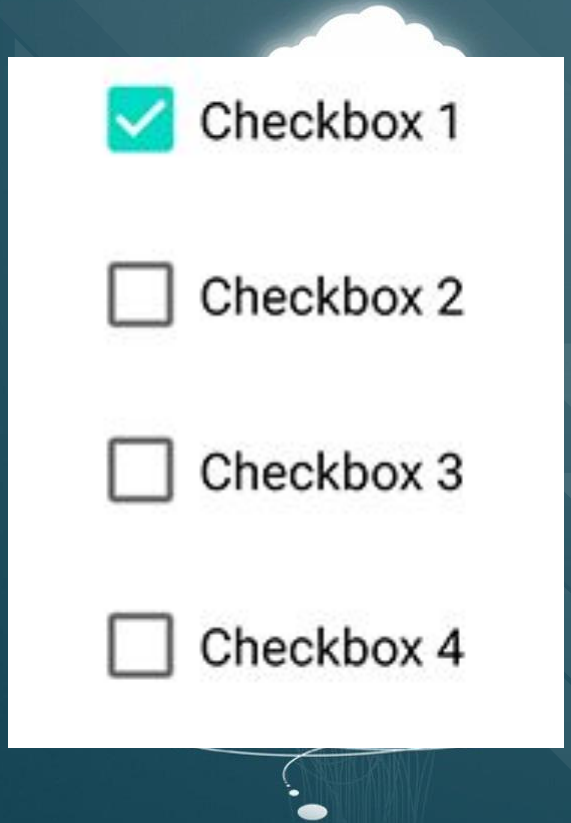
ImageButton:

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.



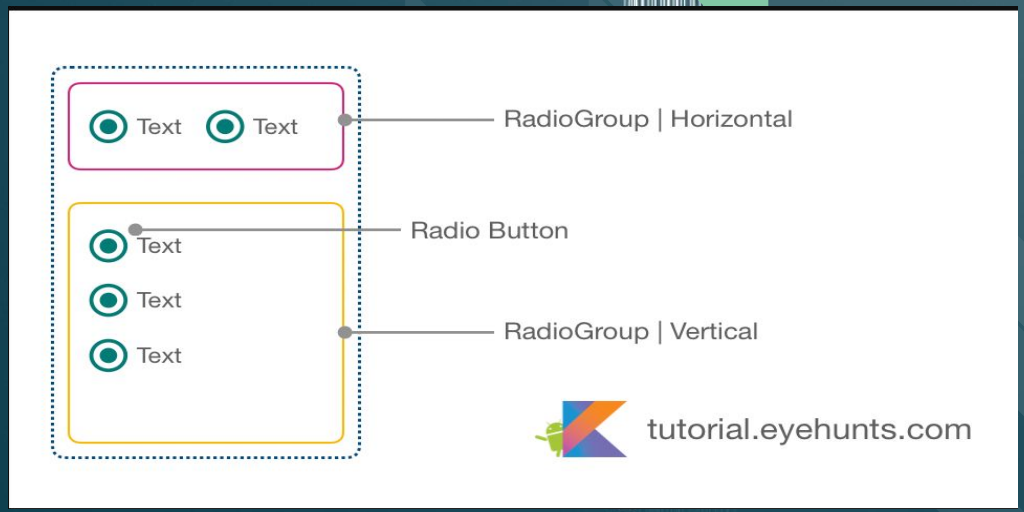
CheckBox:

Android CheckBox is a type of two state button either checked or unchecked. There can be a lot of usage of checkboxes. For example, it can be used to know the hobby of the user, activate/deactivate the specific action etc. Android CheckBox class is the subclass of CompoundButton class.



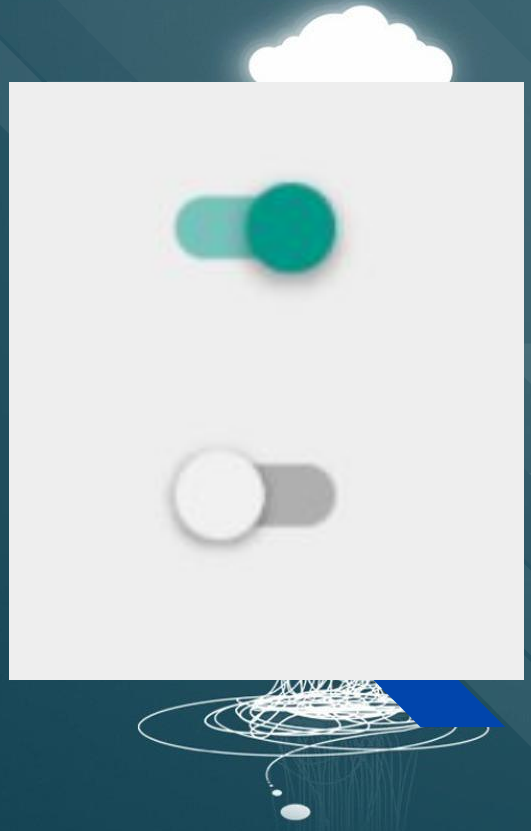
RadioButton/RadioGroup:

Radio buttons allow the user to select one option from a set. ... However, because radio buttons are mutually exclusive, you must group them together inside a RadioGroup . By grouping them together, the system ensures that only one radio button can be selected at a time.



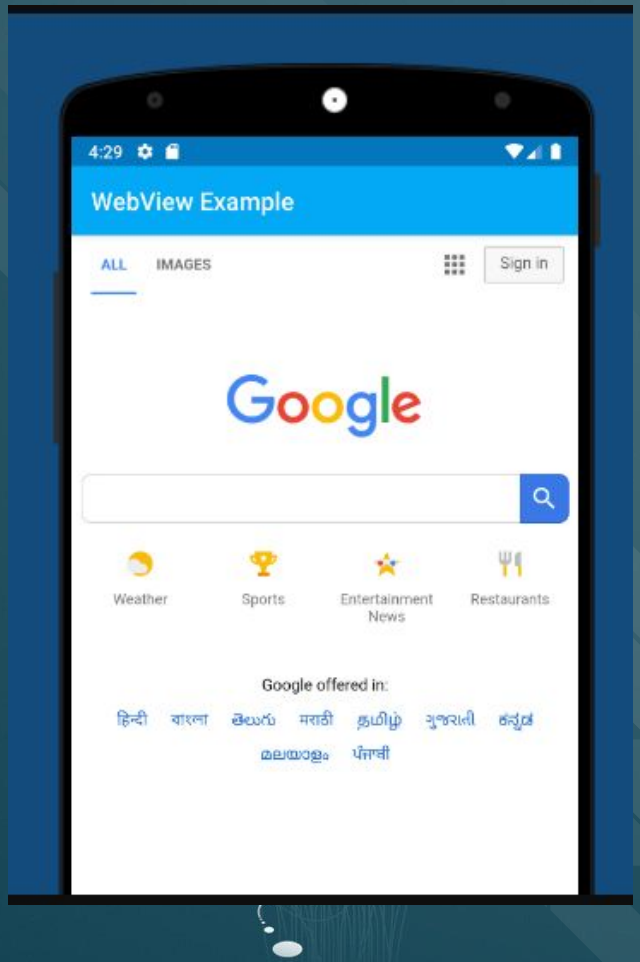
Switch:

In Android, Switch is a two-state toggle switch widget that can select between two options. It is used to display checked and unchecked state of a button providing slider control to user.



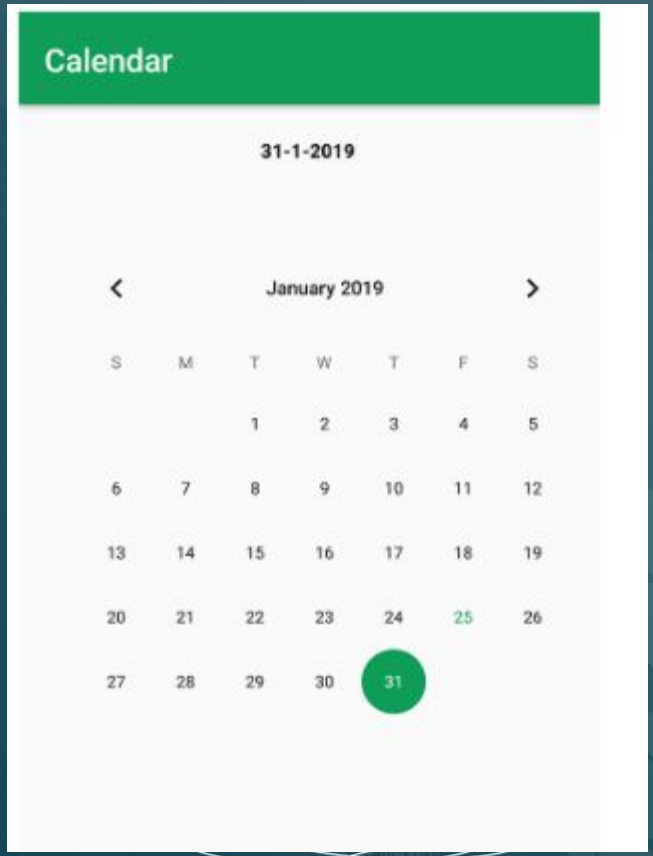
WebView:

WebView is a view that display web pages inside your application. You can also specify HTML string and can show it inside your application using WebView. WebView makes turns your application to a web application.



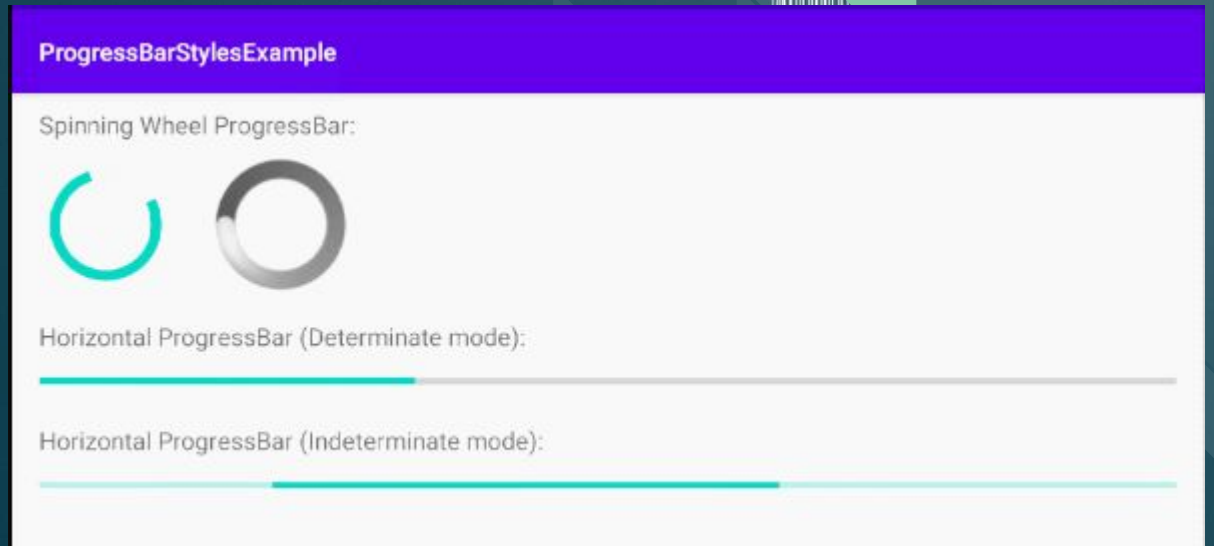
CalenderView:

This article shows how to create an android application for displaying the Calendar using CalendarView. It also provides the selection of the current date and displaying the date.



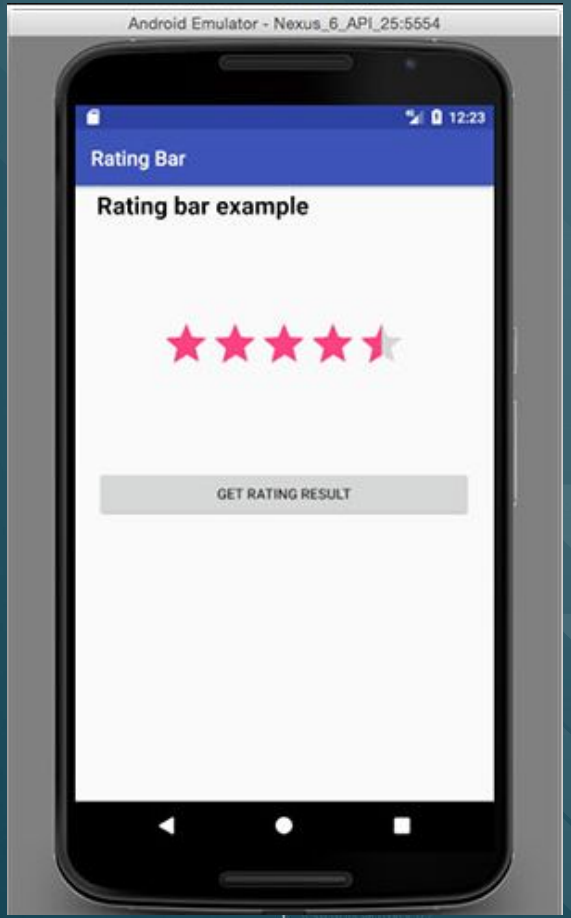
ProgressBar/Horizontal ProgressBar:

In Android, ProgressBar is used to display the status of work being done like analyzing status of work or downloading a file etc. In Android, by default a progress bar will be displayed as a spinning wheel but If we want it to be displayed as a horizontal bar then we need to use style attribute as horizontal.



RatingBar:

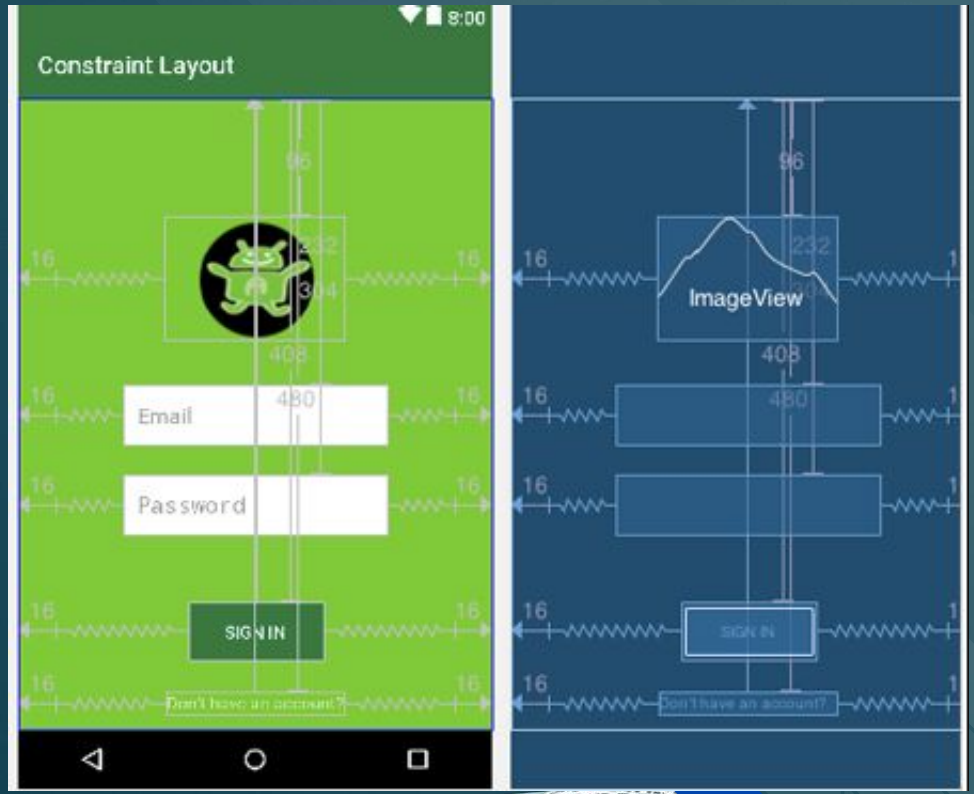
A RatingBar is an extension of SeekBar and ProgressBar that shows a rating in stars. The user can touch/drag or use arrow keys to set the rating when using the default size RatingBar



Layouts:

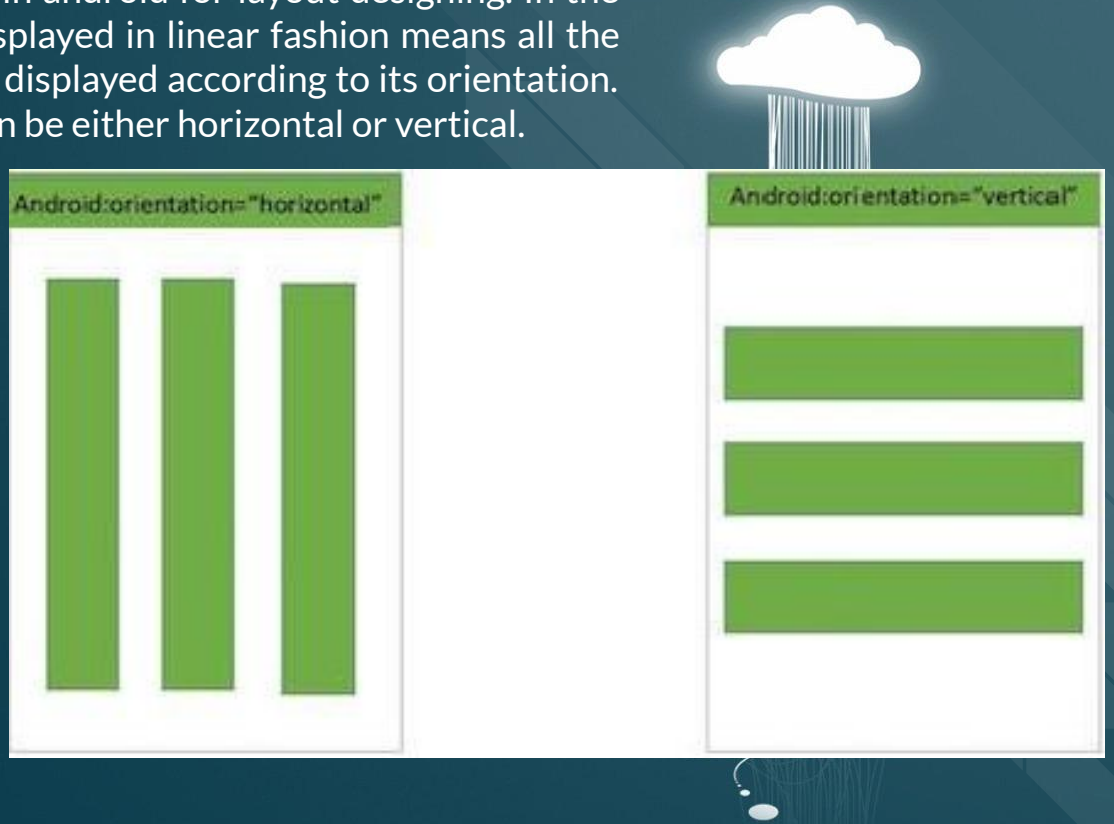
Constraint Layout:

A ConstraintLayout is a ViewGroup which allows you to position and size widgets in a flexible way. ConstraintLayout is available as a support library that you can use on Android systems



Linear Layout :

Linear layout is a simple layout used in android for layout designing. In the Linear layout all the elements are displayed in linear fashion means all the childs/elements of a linear layout are displayed according to its orientation. The value for orientation property can be either horizontal or vertical.



FrameLayout:

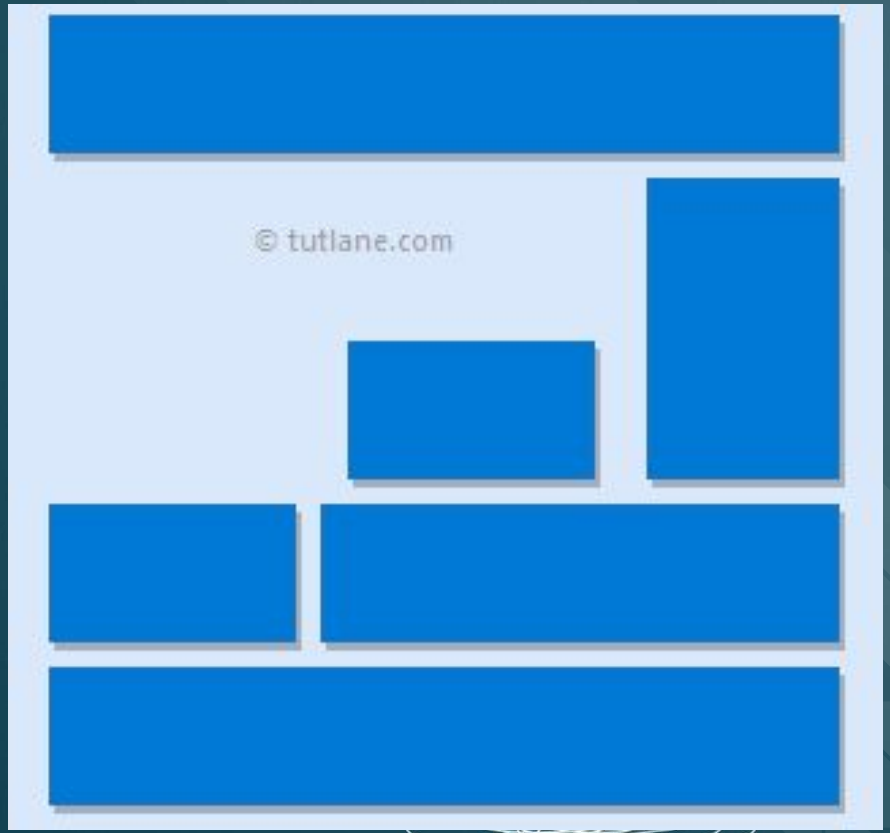
Frame Layout is designed to block out an area on the screen to display a single item. Generally, FrameLayout should be used to hold a single child view, because it can be difficult to organize child views in a way that's scalable to different screen sizes without the children overlapping each other.

Table Layout:

In Android, Table Layout is used to arrange the group of views into rows and columns. Table Layout containers do not display a bor line for their columns, rows or cells. A Table will have as many colu as the row with the most cells.

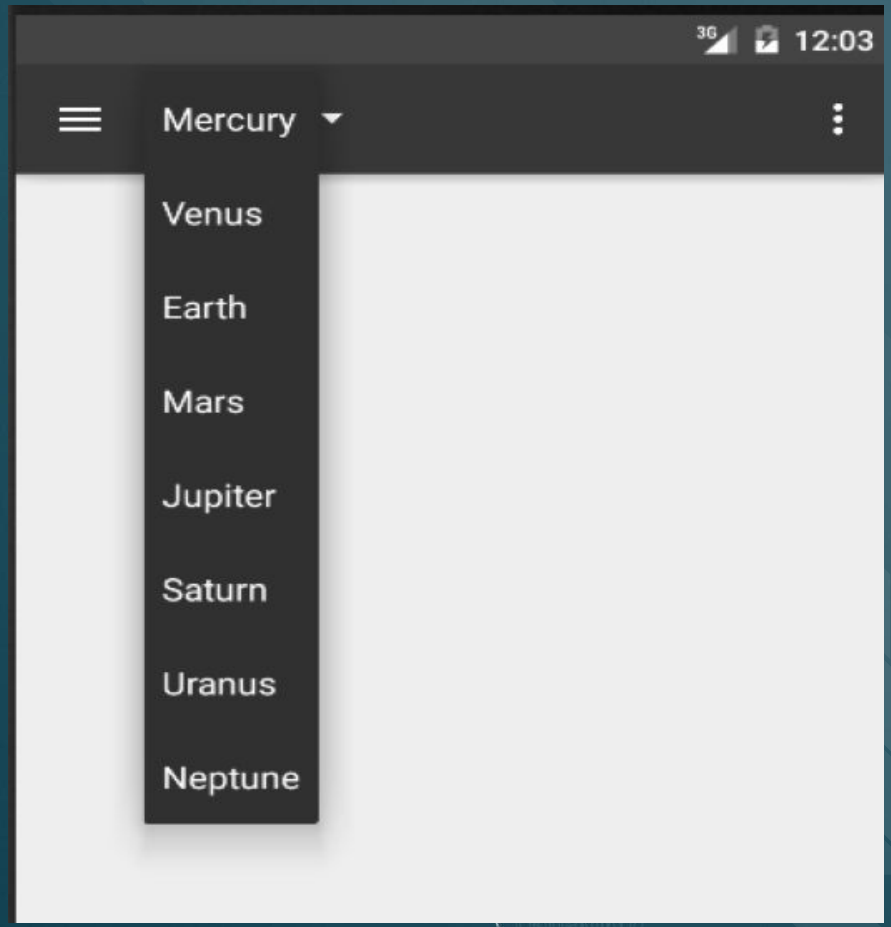
Relative Layout:

Android RelativeLayout enables you to specify how child views are positioned relative to each other. The position of each view can be specified as relative to sibling elements or relative to the parent.



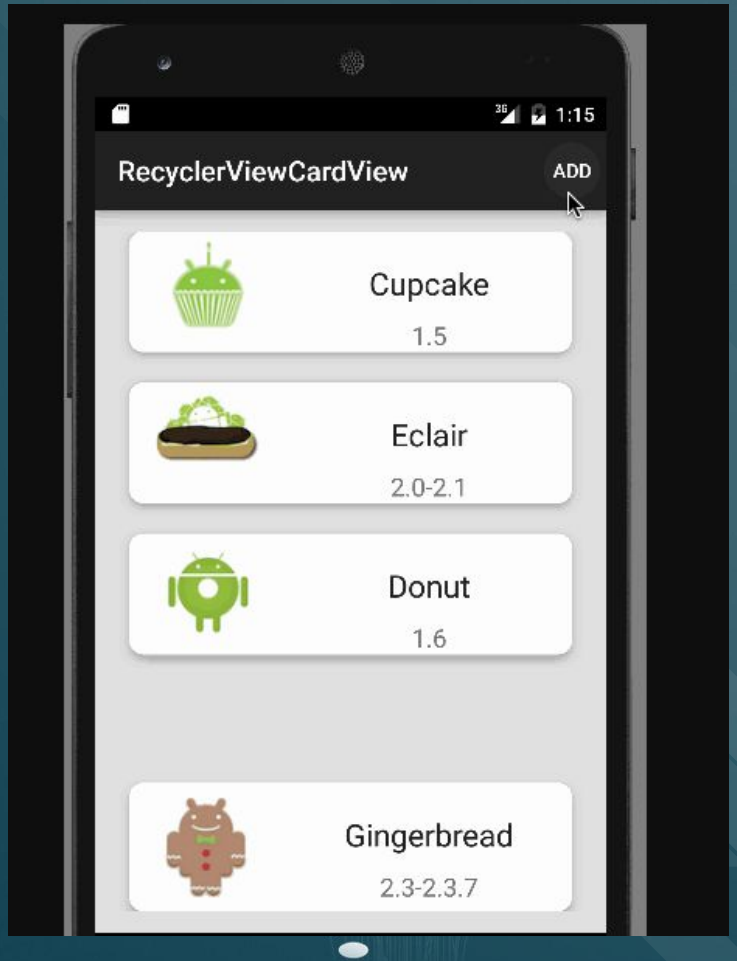
Spinner:

Android Spinner is a view similar to the dropdown list which is used to select one option from the list of options. It provides an easy way to select one item from the list of items and it shows a dropdown list of all values when we click on it. The default value of the android spinner will be the currently selected value



CardView :

CardView is a new widget in Android that can be used to display any sort of data by providing a rounded corner layout along with a specific elevation. CardView is the view that can display views on top of each other. The main usage of CardView is that it helps to give a rich feel and look to the UI design.

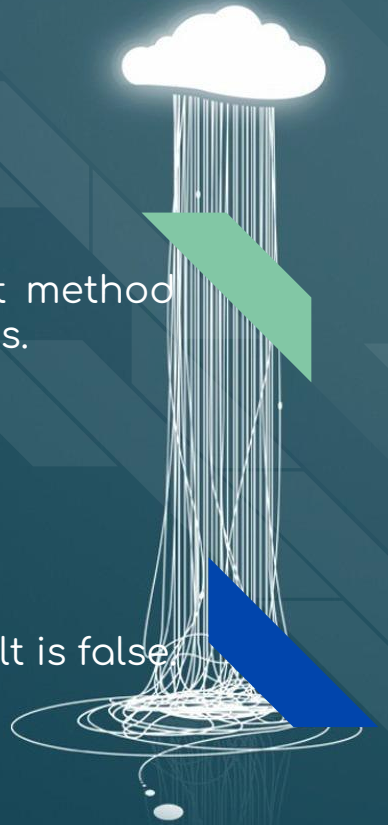


Attributes:

★ **android:id**
This is the ID which uniquely identifies the control.

★ **android:capitalize**
If set, specifies that this TextView has a textual input method and should automatically capitalize what the user types.
Don't automatically capitalize anything - 0
- Capitalize the first word of each sentence - 1
- Capitalize the first letter of every word - 2
- Capitalize every character - 3

★ **android:cursorVisible**
Makes the cursor visible (the default) or invisible. Default is false.



- **android:editable**
  If set to true, specifies that this TextView has an input method.

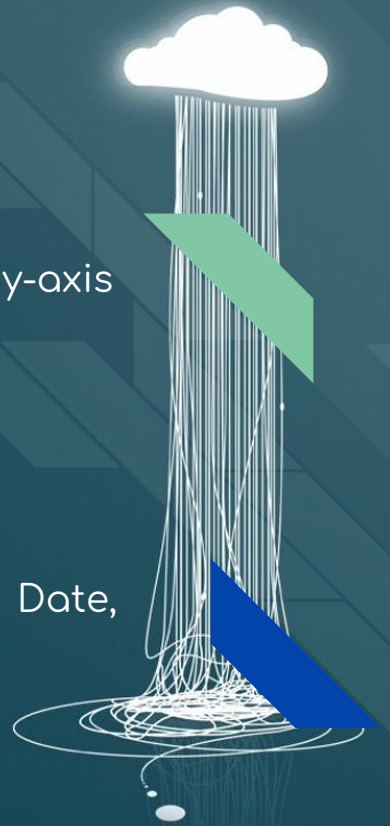
- **android:fontFamily**
  Font family (named by string) for the text.

- **android:gravity**
  Specifies how to align the text by the view's x- and/or y-axis when the text is smaller than the view.

- **android:hint**
  Hint text to display when the text is empty.

- **android:inputType**
  The type of data being placed in a text field. Phone, Date, Time, Number, Password etc.

- **Android:text**
  Text to display.

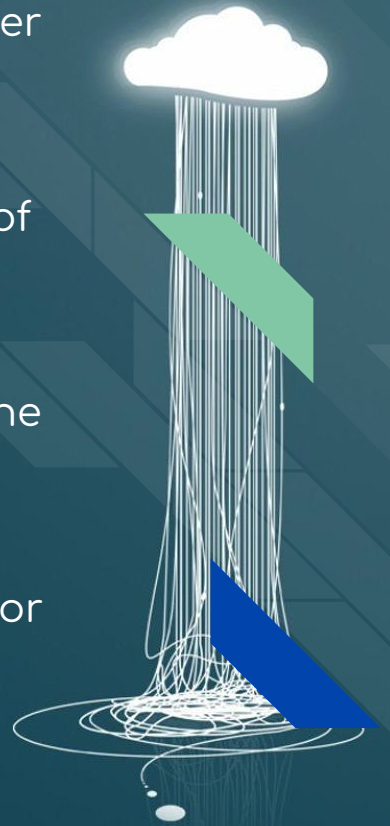


★ **android:textAllCaps**
Present the text in ALL CAPS. Possible value either "true" or "false".

★ **android:textColor**
Text color. May be a color value, in the form of "#rgb", "#argb", "#rrggbb", or "#aarrggbb".

★ **android:textColorHint**
Color of the hint text. May be a color value, in the form of "#rgb", "#argb", "#rrggbb", or "#aarrggbb".

★ **android:textSize**
Size of the text. Recommended dimension type for text is "sp" for scaled-pixels (example: 15sp).

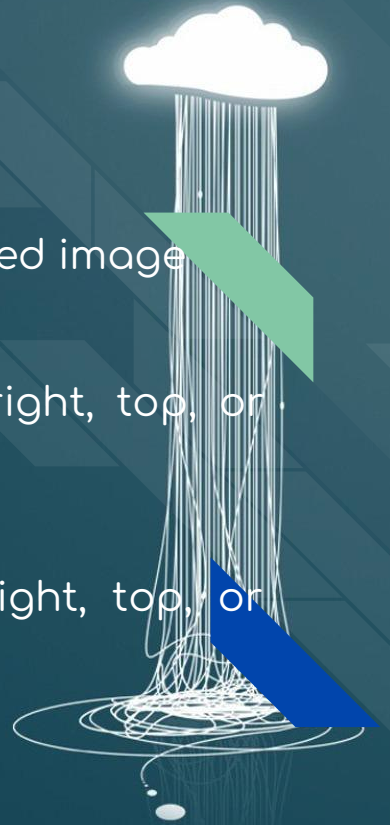


- **android:src/app:srcCompat**
  To add the file path of the inserted image

- **android:background**
  To provide a background color to the inserted image

- **android:padding**
  To add padding to the image from left, right, top, or bottom of the

- **Android:margin**
  To add margin to the image from left, right, top, or bottom of the



Flutter

Flutter transforms the app development process. Build, test, and deploy beautiful mobile, web, desktop, and embedded apps from a single codebase.
Flutter is an open-source UI software development kit created by Google. It is used to develop cross platform applications for Android, iOS, Linux, Mac, Windows, Technologies for flutter are React Native, Dart etc.

Advantage of Flutter:

- ★ Same UI and Business Logic in All Platforms
- ★ Reduced Code Development Time
- ★ Similar to Native App Performance
- ★ Custom, Animated UI of Any Complexity Available
- ★ The Potential Ability to Go Beyond Mobile

Flutter

Develop for Mobile, Web and Desktop

Native Mobile Apps:
Loved and used in production by thousands of Developers, Startups and Enterprises

Flutter for Web:
In Beta now, but has strong support from the developer community

Desktop Apps:
MacOS Support: In Alpha
Windows and Linux: Under Development

THANK YOU