Android DEV

- Package name should be unique

Android Menifest

Decides how the application will look.

Drawable

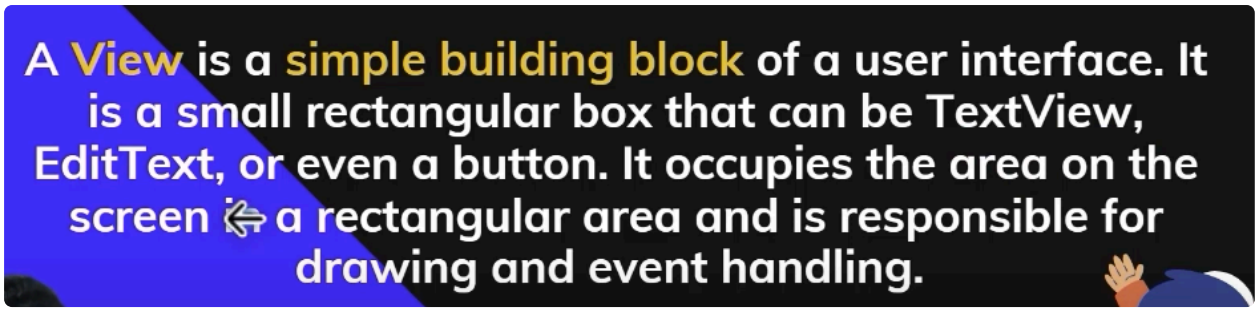
pictures wagiyra yehi save krenge.

Layout

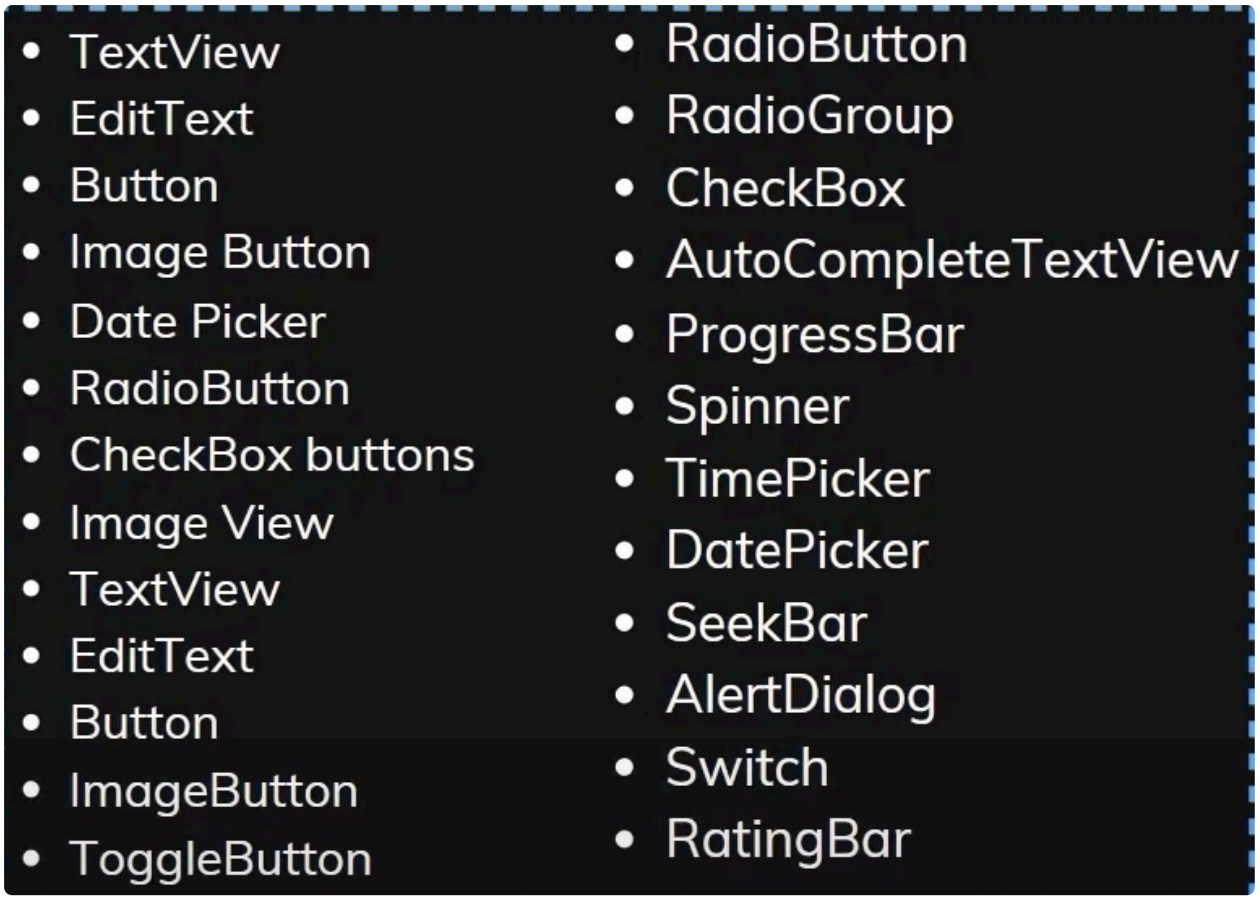
it Consist of frontend files and menifest contains backend files of app

frontend uses - xml

View - is a simple building block in UI



View Types

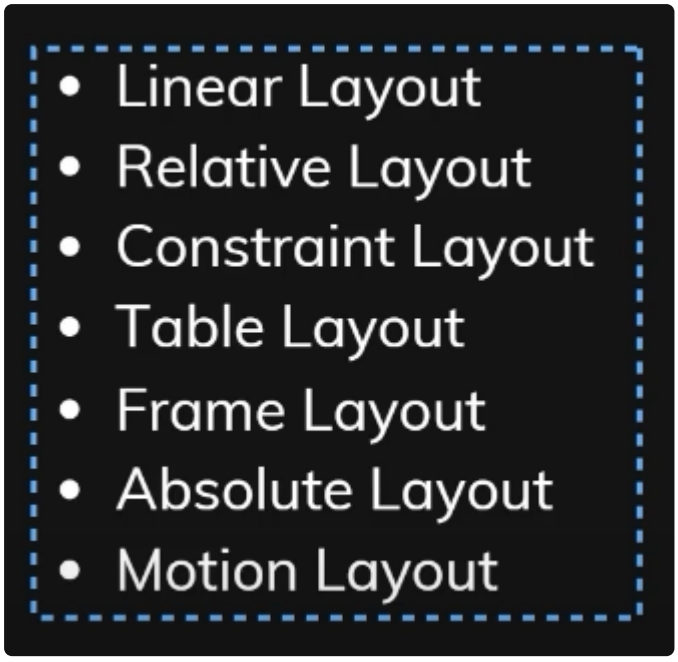


Layout

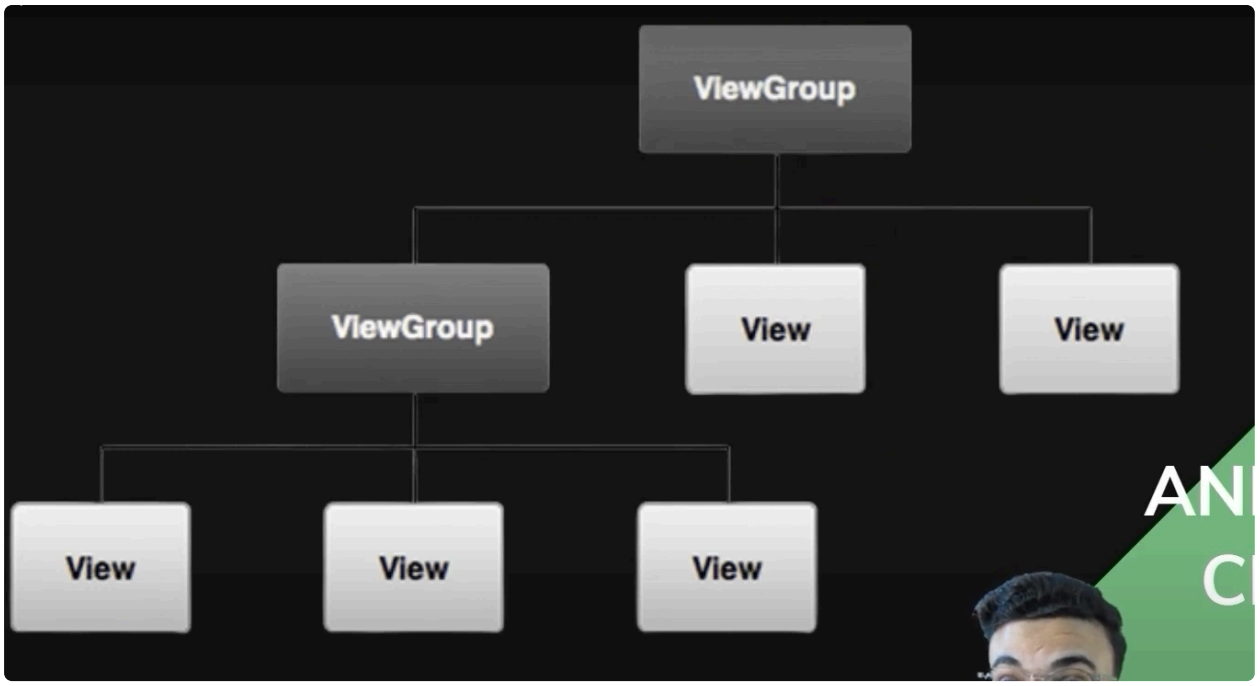
It hold all the views basically it is the parent of all the views

Layouts are used to define the actual UI(User interface) of our application. It holds all the elements (i.e. views) or the tools that we want to use in our application. For example, TextView, Button and other UI elements.

Types of layout



View and View GRoup

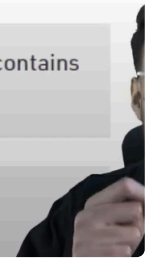


Layout ke andr layout

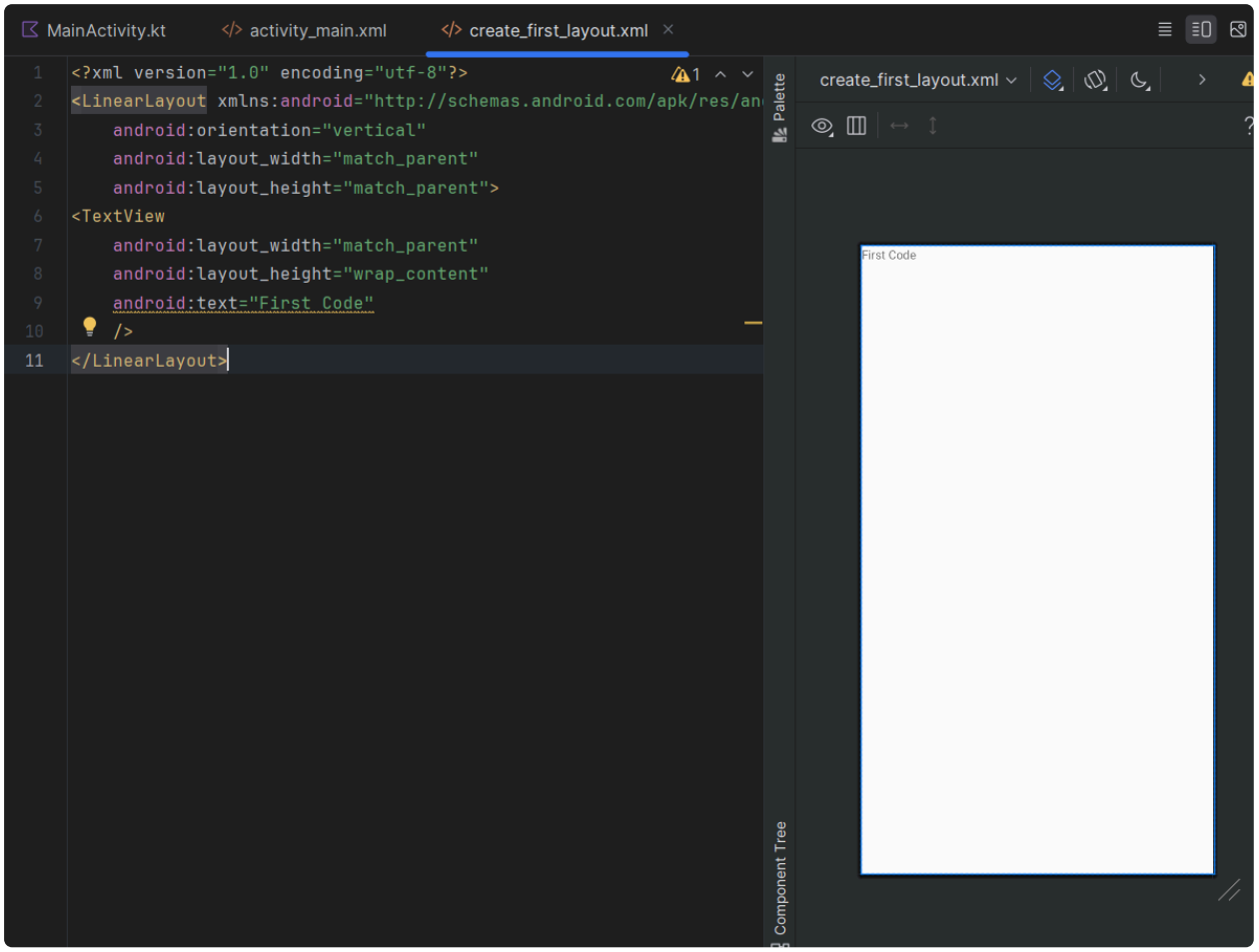
ViewGroup bs layout ka ni hota bhot sare components ka viewGroup ho skta hai

View vs ViewGroup

View	ViewGroup
View is a simple rectangle box that responds to the user's actions.	ViewGroup is the invisible container. It holds View and ViewGroup
View is the SuperClass of All component like TextView, EditText, ListView, etc	ViewGroup is a collection of Views(TextView, EditText, ListView, etc..), somewhat like a container.
A View object is a component of the user interface (UI) like a button or a text box, and it's also called a widget.	A ViewGroup object is a layout, that is, a container of other ViewGroup objects (layouts) and View objects (widgets)
Examples are EditText, Button, CheckBox, etc.	For example, LinearLayout is the ViewGroup that contains Button(View), and other Layouts also.
android.view.View which is the base class of all UI classes.	ViewGroup is the base class for Layouts.



Creating first Layout



1. dp (Density-independent Pixels):

- Purpose: dp is a unit of measurement that is density-independent. It is recommended to specify dimensions in layouts to ensure consistent visual size across different screen densities.
- Calculation: 1 dp is equivalent to one physical pixel on a medium-density screen (160 dpi). The actual size of a dp is adjusted based on the screen's density.

In this above example, the width and height of the TextView are set to 100dp and 50dp, respectively.

2. sp (Scale-independent Pixels):

- Purpose: sp is similar to dp, but it is specifically used for defining text size. It takes into account the user's preferred

text size setting in the device's system settings.

- Calculation: 1 sp is the same as 1 dp, but it also considers the user's font size preference.

3. px (Pixels):

- Purpose: px is the most basic unit and represents a single pixel on the screen. It is not recommended to specify dimensions in layouts due to its lack of density independence.

In this example, the width and height of the ImageView are set to 200px and 100px, respectively.

4. dip (Density-independent Pixels):

- Purpose: dip is an older term for dp, and the two can be used interchangeably. Both

represent density-independent pixels.

Here, the padding is set to 10dip, which means 10 density-independent pixels.

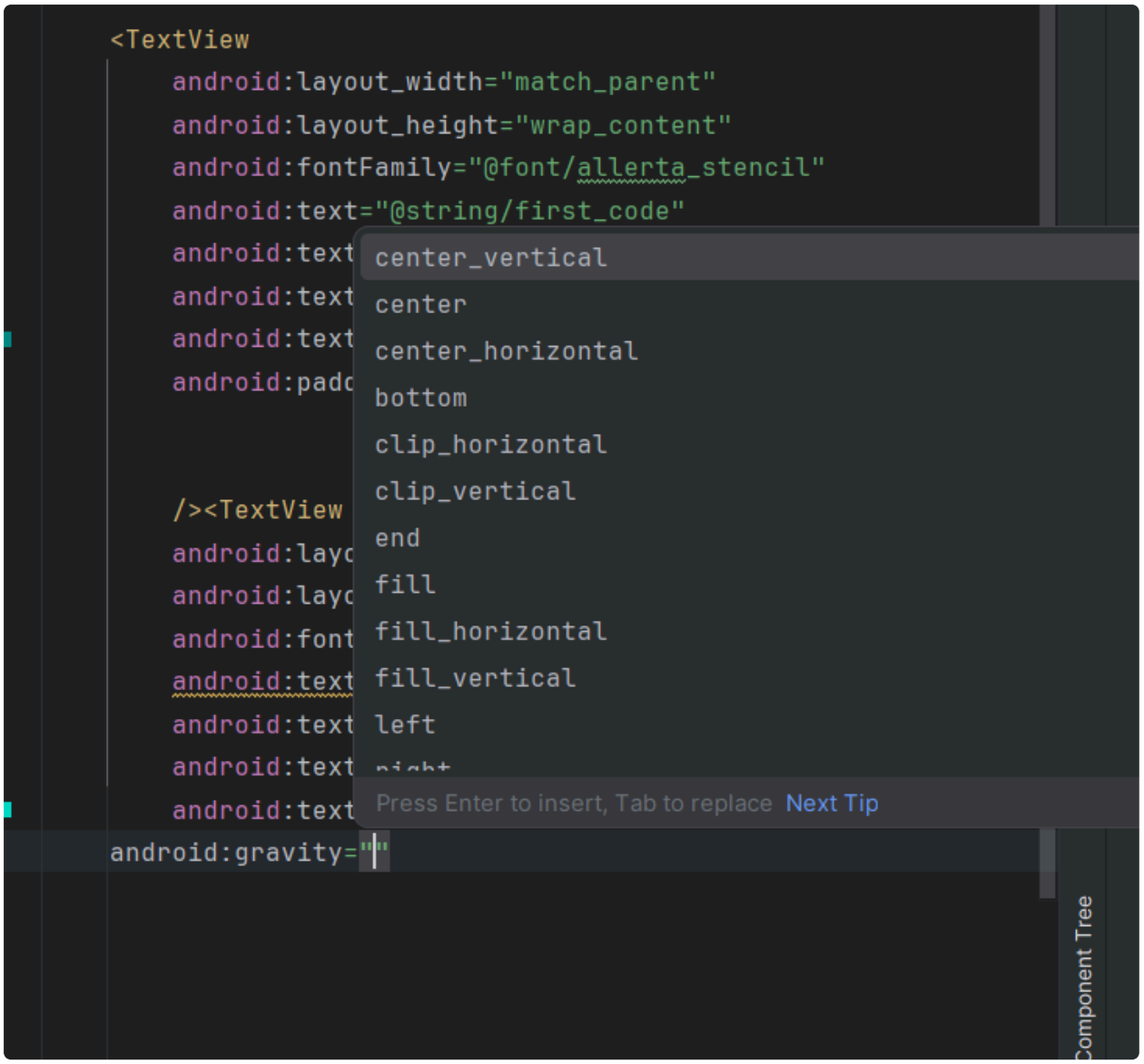
In summary, dp and sp are commonly used for dimensions in layouts and text sizes, respectively, due to their density independence. It is generally recommended to use dp for dimensions and sp for text sizes to ensure a consistent and user-friendly experience across different devices with varying screen densities and user preferences.

Mastering the art of layout

Padding-left == paddingStart kyuki mmanlo language english hai to english left se start hoti hai aur padding start lagyege to left se margin milega mgr agr urdu hoti to wo right se start hoti to padding start me rigth me padding milta.

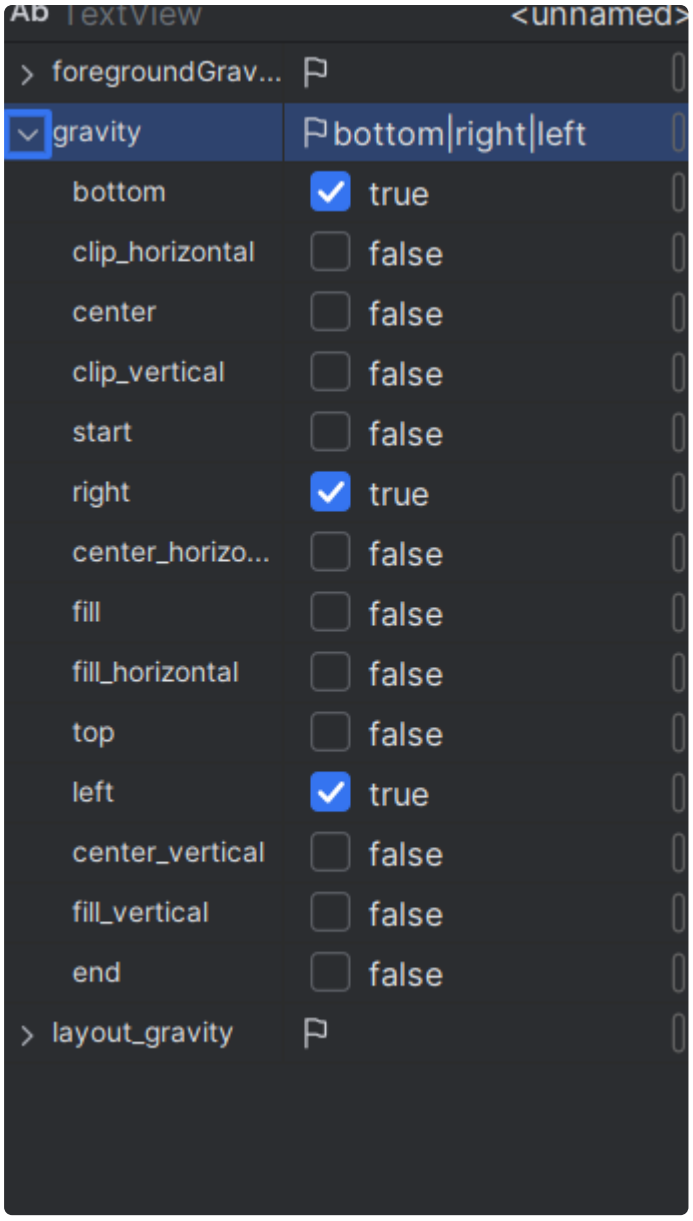
Gravity

Gravity is to define the position of a view like top bottom center etc



Hamesha view ke andr wali cheez pe gravity apply hogi

note - agr ek se jyada gravity ke options use krne hai to attributes me jake krenge



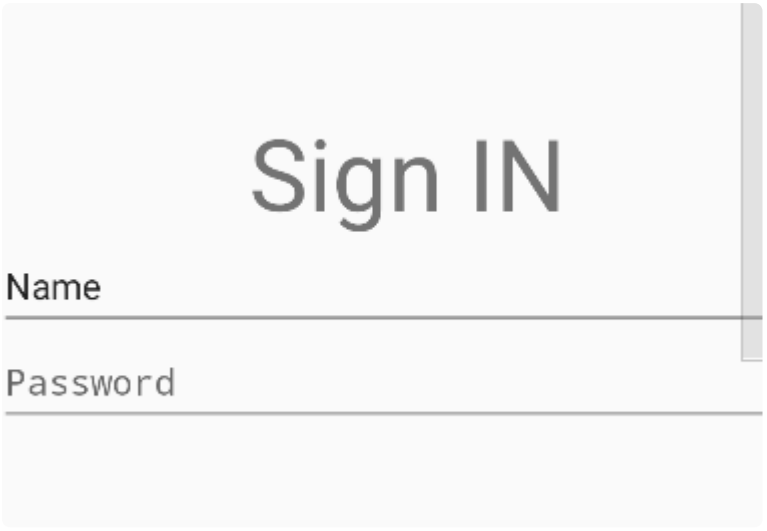
ya fir code me aise

```
android:gravity="bottom|right|left"
```

Edit Text

```
<EditText
    android:id="@+id/editTextText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Name"
    android:inputType="text"
    android:minHeight="48dp"
    android:text="Name"
    android:minEms="10"
    android:maxEms="20"
    />
```

EditText means user can write something here minEms mtlb minimum itne character and maxEms vice versa



Name and password are EditText signIn is textView

Image view Important attributes

agr icon ki color change krne hai to hm tint attribute ka use krte hai

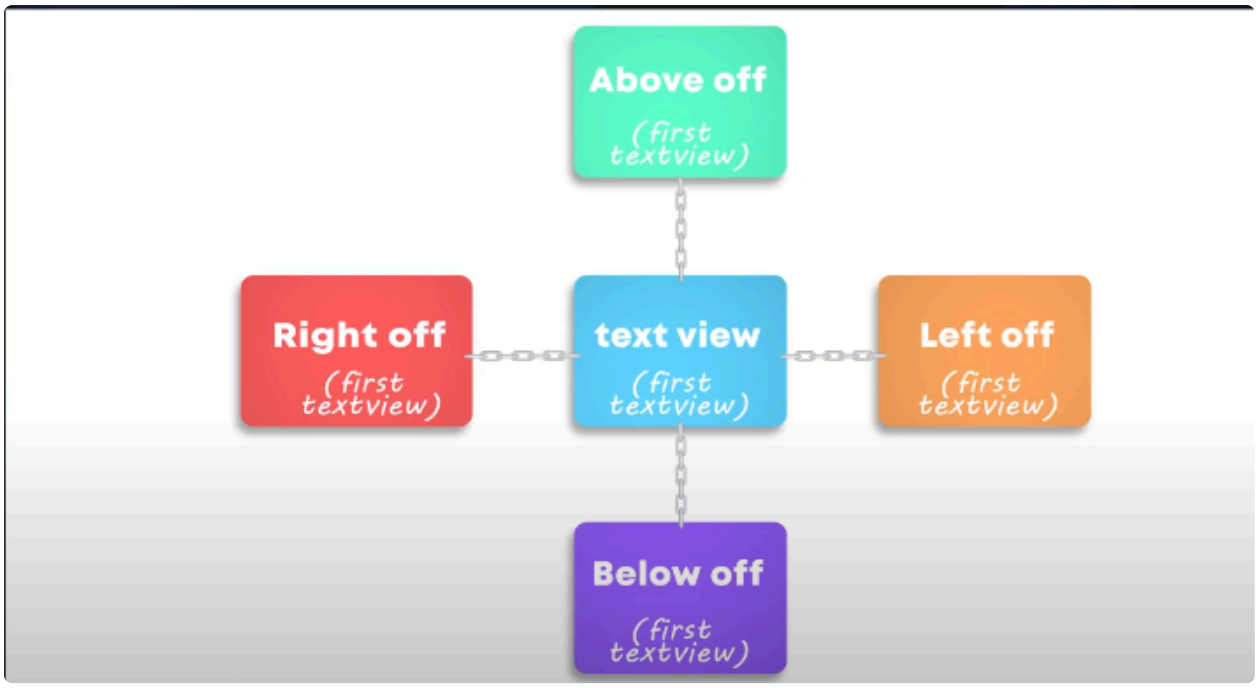
```
<ImageView
    android:id="@+id/imageView2"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:tint="@color/teal_700"
    app:srcCompat="@drawable/home"
    android:layout_gravity="center_horizontal|center_vertical"
    android:layout_marginTop="108dp"
    />
```

Relative Layout

relative layout relativity ke base pe kaam krta hai mtlb ki agr kisi textview ko kahi place krna hai to hm batayenge ki is imageview ke left right bottom top me yaha chle jao

In Android, the `RelativeLayout` is a ViewGroup that allows you to position its child views in relation to each other or to the parent container. It’s especially useful for creating complex layouts where elements need to be aligned with or positioned around other views.

hmko yaha id dena compulsary hai taki hm bata sake is is id ke element ke uopr chale jao



first view is id of the textview
Relative layout me jo upr rhega wohi UI me niche rhega

here we can see ki one is under two

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".RelativeLayoutActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="one"
        android:textColor="@color/teal_700"/>

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="two" />
</RelativeLayout>
```

but here in code it is above two and this is happening due to re;ative layout

just changed the layout form relative to linear

and see the diffrence in the output

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".RelativeLayoutActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="one"
        android:textColor="@color/teal_700"/>

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="two" />
</LinearLayout>
```

Relative Layout important attributes

rigthof, left of,bellow of, kr ke use krte hai

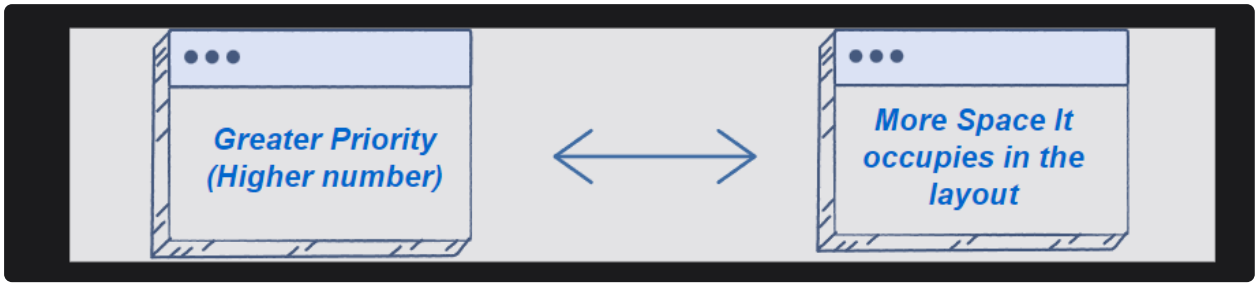
Linear Layouts

ye ek predefined direction deta hai jo ki horizontal ya verticle hoti hai

- yaha hmko sbse phle orientation define krni hoti hai

Weight in linear Layout

Layout Weight in Android is the priority level of space that a child view occupies in a linear layout.
**



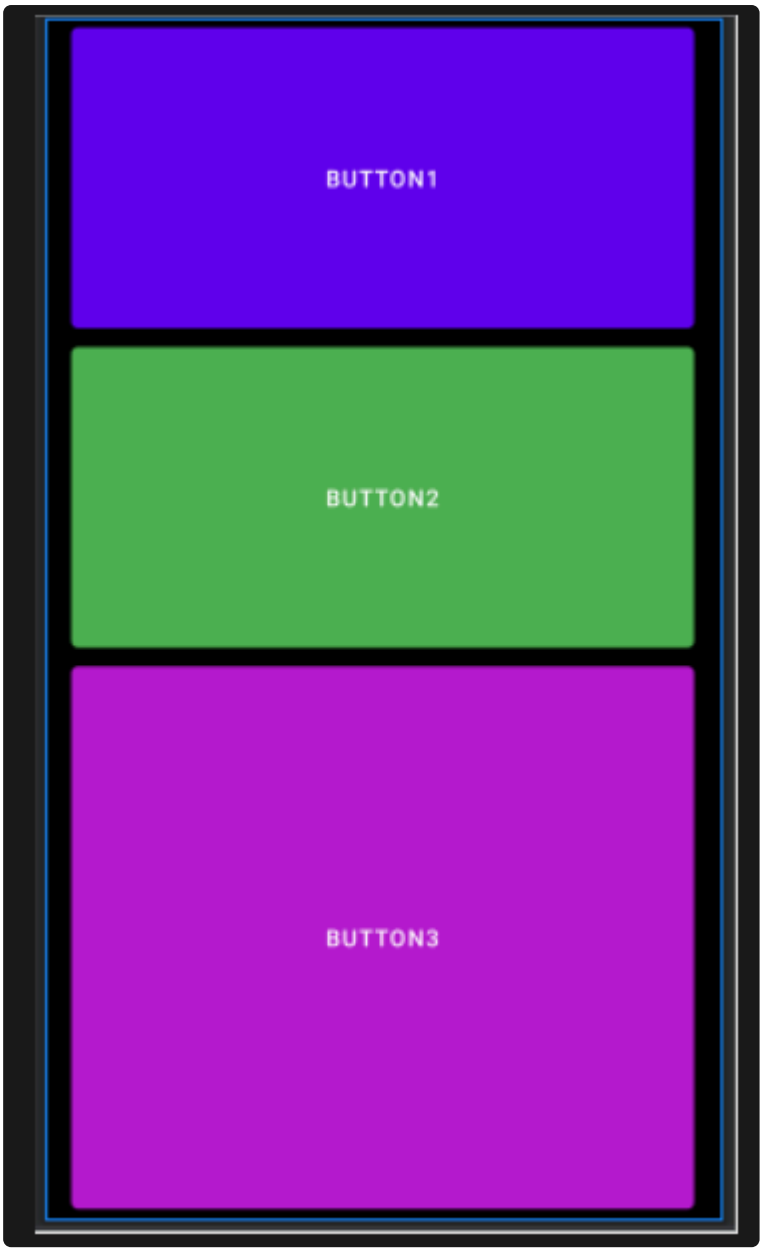

```
<Button
    android:id="@+id/b"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="Button1" />
```

```
<Button
    android:id="@+id/b2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="Button2"
    app:backgroundTint="#4CAF50" />
```

```
<Button
    android:id="@+id/b3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="2"
    android:text="Button3"
    app:backgroundTint="#B51ACF" />
```

- For Button **“@+id/b”** and **“@+id/b2”**, the `layout_weight` is set to **1**. Therefore, both buttons occupy equal space in the layout.
- For **“@+id/b3”**, the `layout_weight` is set to **2**. *Button3* will occupy the most space in the layout since it has a higher `layout_weight` as compared

to *Button1* and *Button2*.

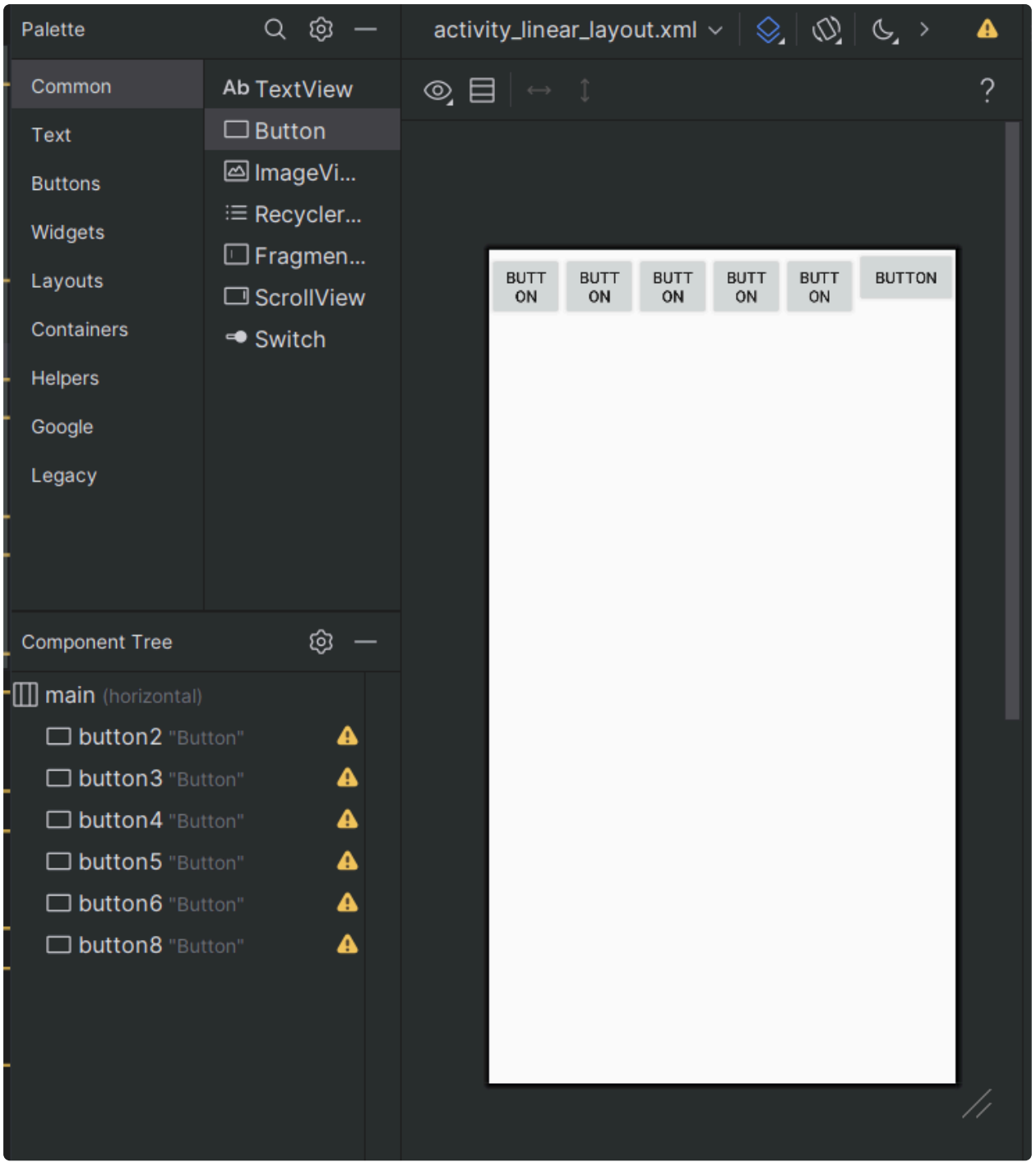


Weigthsum

Weightsum hm layout ke attributes me hi define kr denge ye ye batyega ki kitne element ko same length milegi

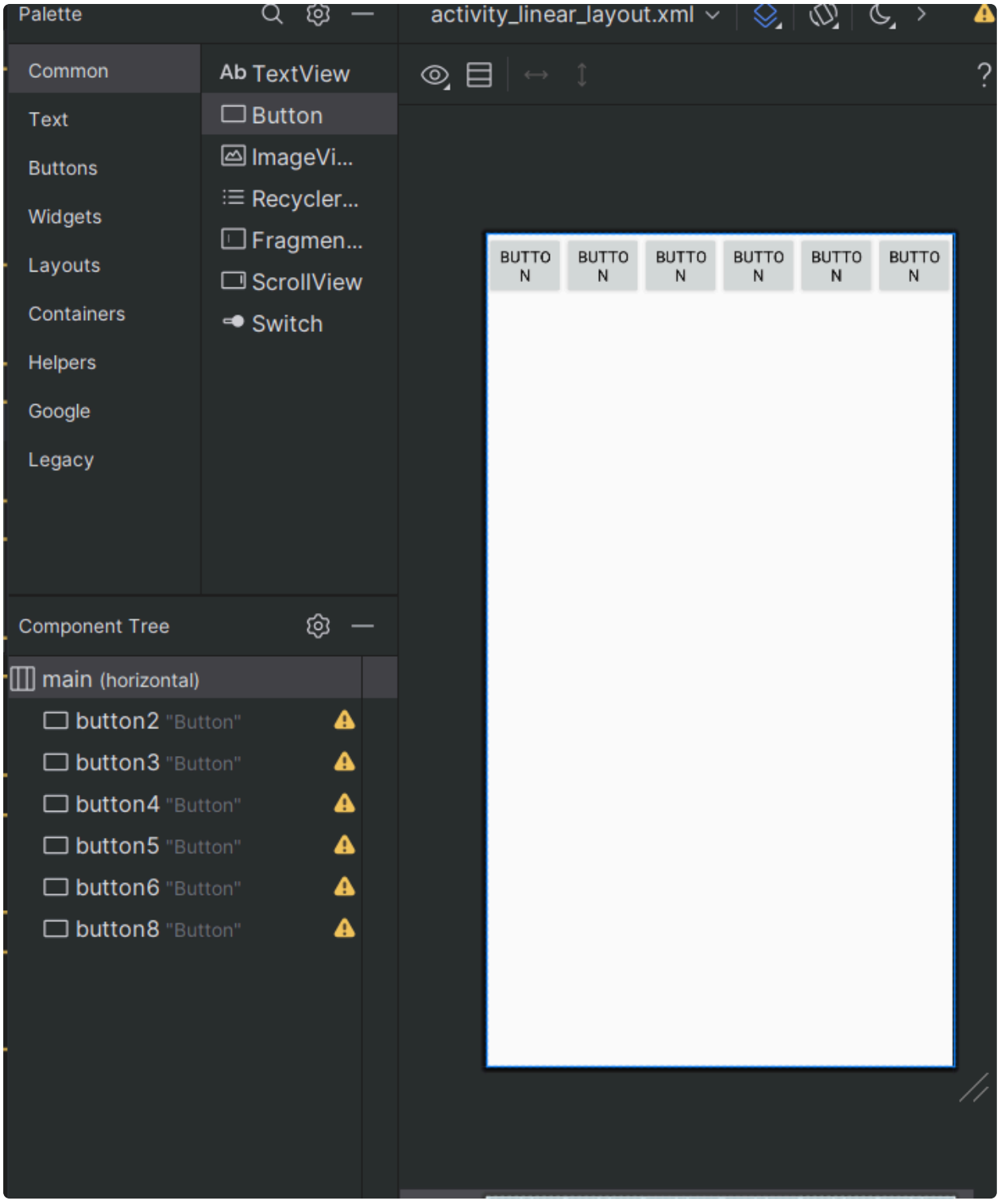
```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".LinearLayoutActivity"
    android:orientation="horizontal"
    android:weightSum="5"
    >
```

yaha hmne 5 define kra hai to 5 elements ko same length milegi



yaha dekho 6 button hai to 5 ke same size hai 6th ka different hai agr Weightsum ko 6 kr de to 6wo bararabr ho jayenge

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".LinearLayoutActivity"
    android:orientation="horizontal"
    android:weightSum="6"
    >
```

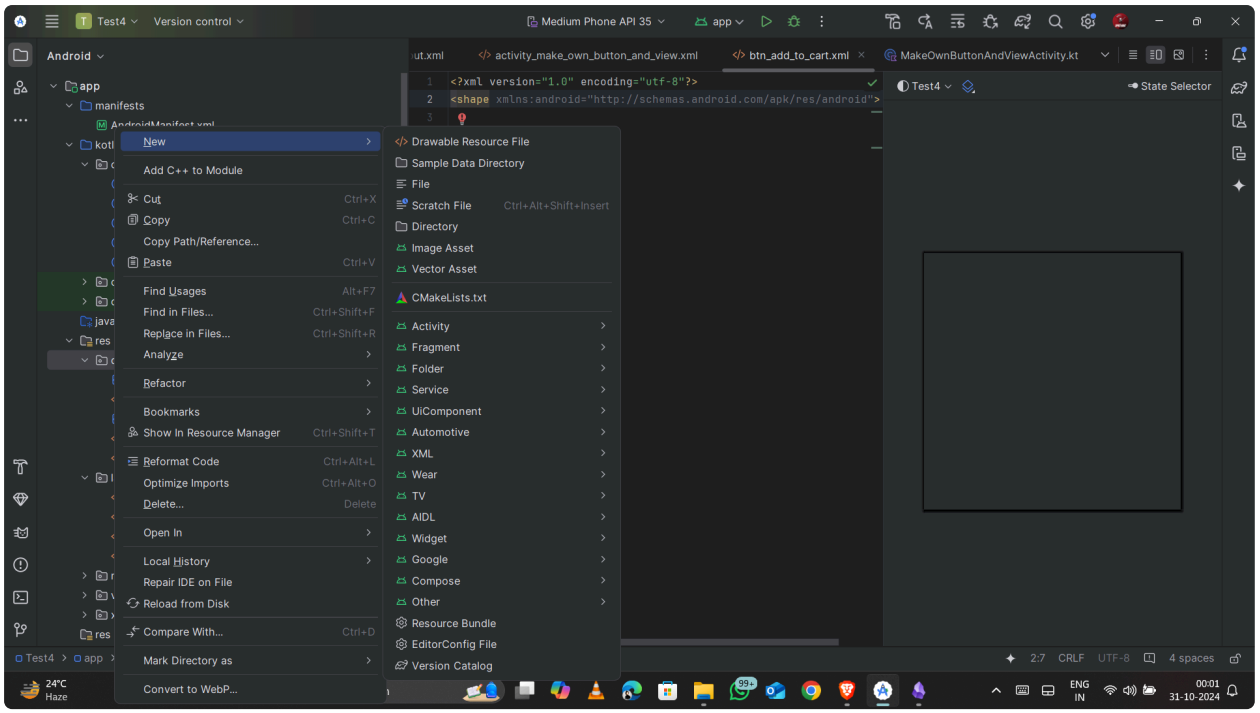


Note - height = wrapcontant === height=0dp

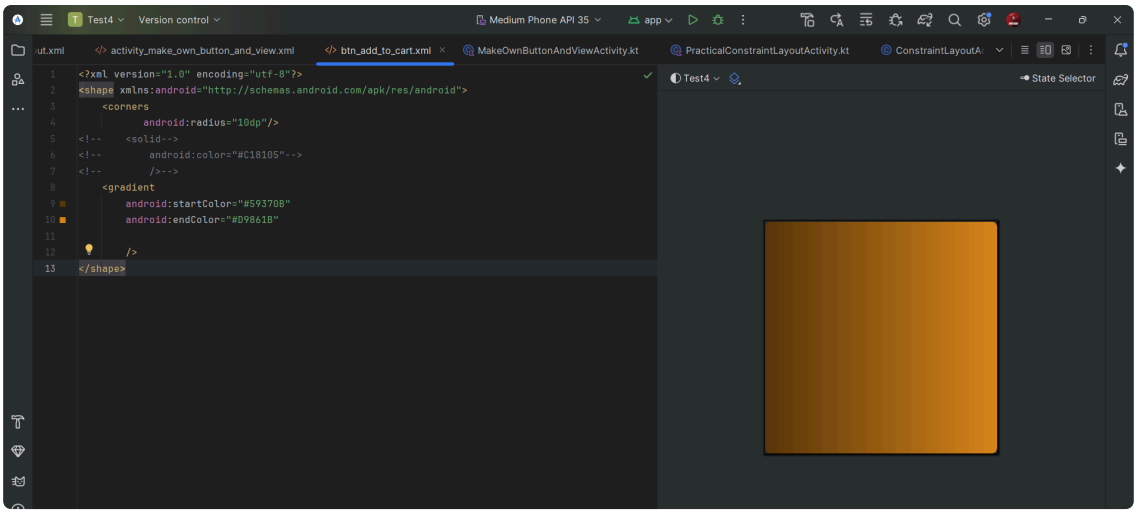
Constraint layout

This layout is faster than other layouts in loading time.

Make your own button and view



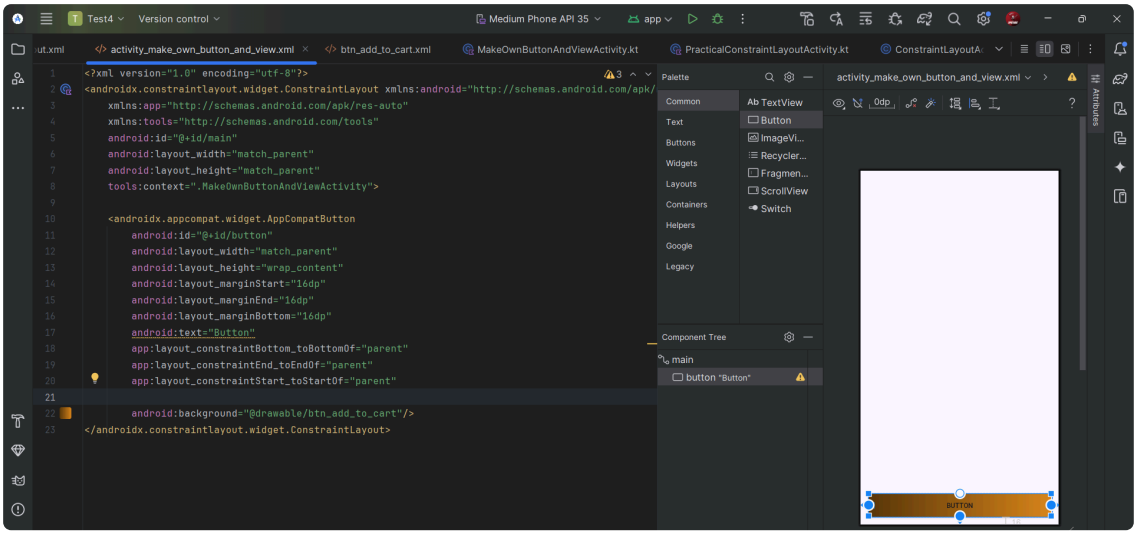
- Go to drawable
- select drawable resource file
- name it and change from selector to shape
  code it in the new drawble file
  ex for button -



to usi this just go to activity and for apply this as background

```
android:background="@drawable/btn_add_to_cart"/>
```

like this



if the normal button doesnt work just change the attribute to this

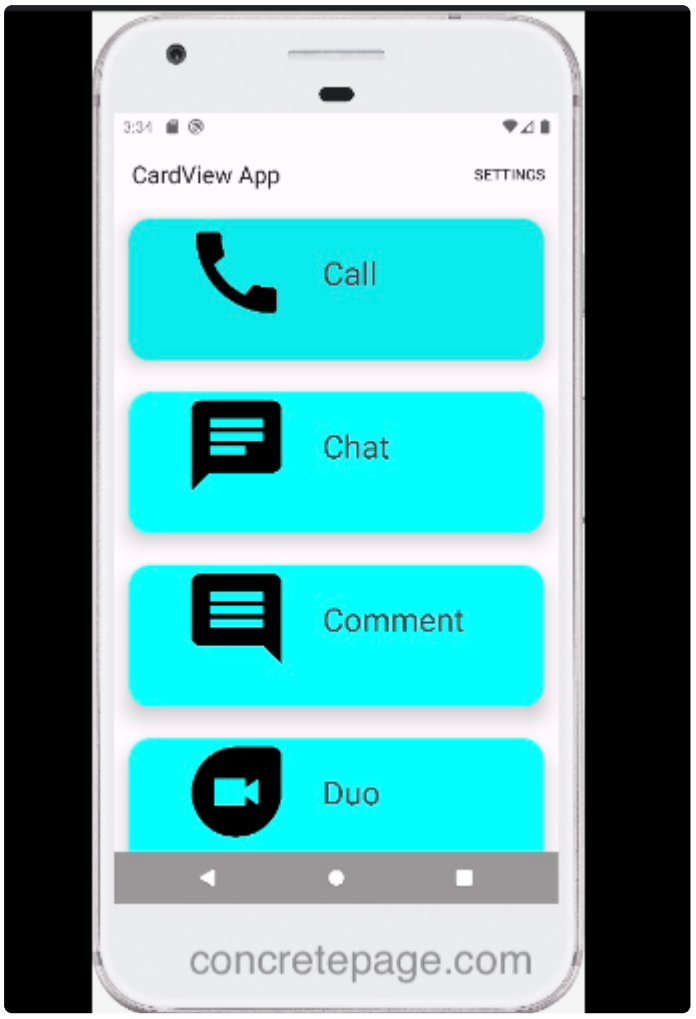
```
<androidx.appcompat.widget.AppCompatButton
```

It will work as same as button

- angle bhi de skte hai magr wo 90 se divide hona chahiye.

- agr hm custom component wale fine me changing krenge t0 direct activity me bhi changing ho jayegi

TO make custom cardView



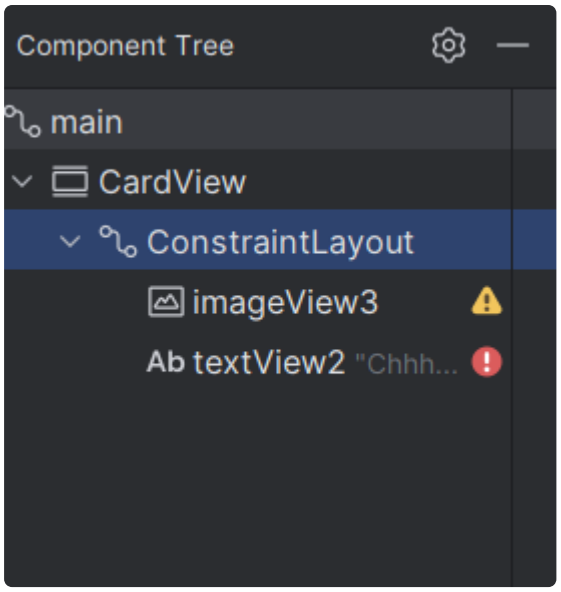
this call , chat , comment, duo are cards

Cardview ek view group hai jo ki views ko contains krta hai aur ye predefined hota hai studio me

Important elements of card view

- cardElevation is an attribute jo define krta hai ki card apne backgroud se kitna utha hoga
- 
- ```
  app:cardElevation="30dp"
  app:cardCornerRadius="50dp"
  ```

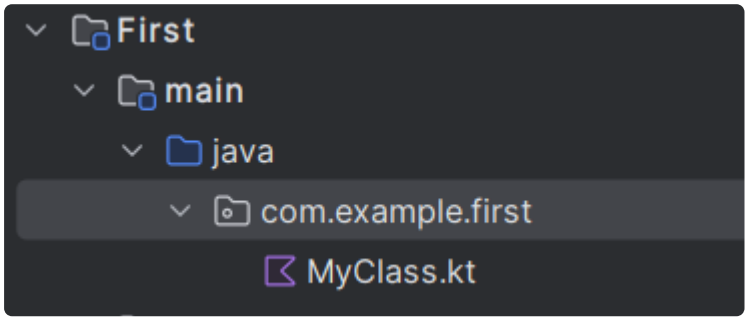
  -- Card View is created on heirarcy constraint layout>cardview>constraint layout> other elemets
  ```



aise

Kotlin

```
JAVA and kotlin are interoperable
kotlin bhi JVM ka use kr ke interprate
hoti hai
```



Dekho kotlin java ke andr hai

Variables in Kotlin

val and var

```
fun main(){
    println("Hello World");
    var name="akhilesh"
    print("hello $name")

}
```

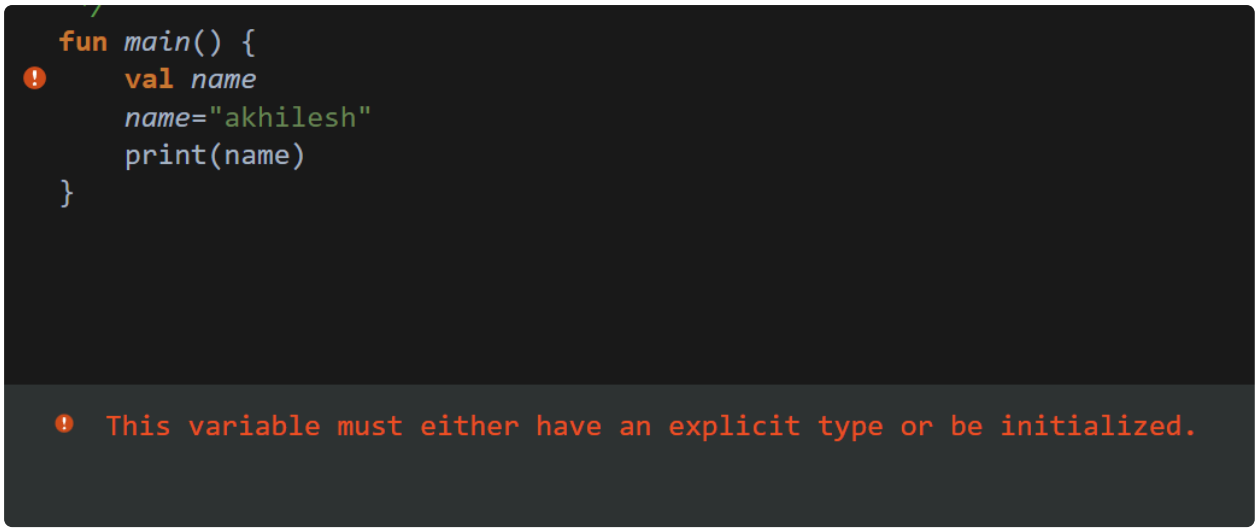
to use a value we have to use template littrels like in JS

var can be reassigned but val can not be reassigned

```
*/
fun main() {
    val name = "akhilesh"
    print(name)
}
```

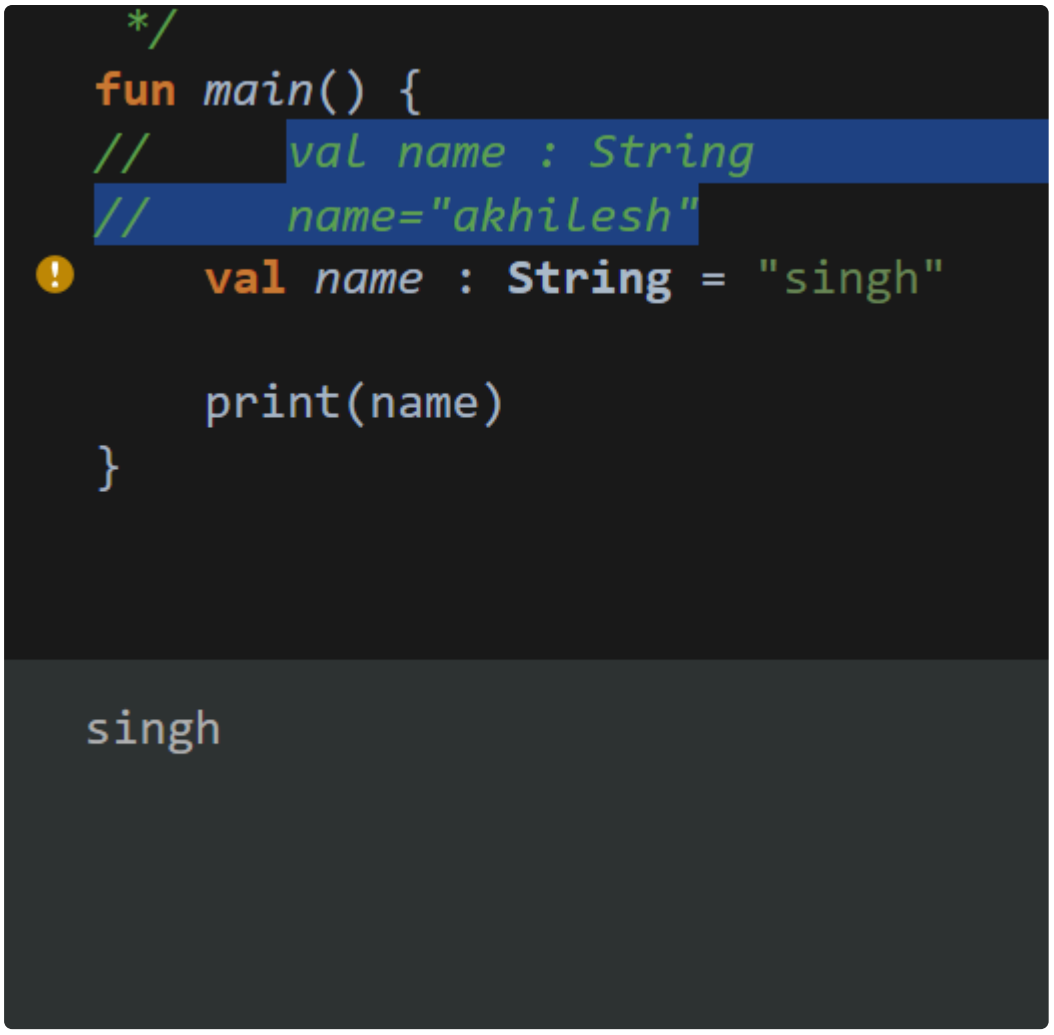
```
akhilesh
```

yaha tk thik hai mtlb aise data type define na bhi kre to chalega jaise ki JS me hota hai



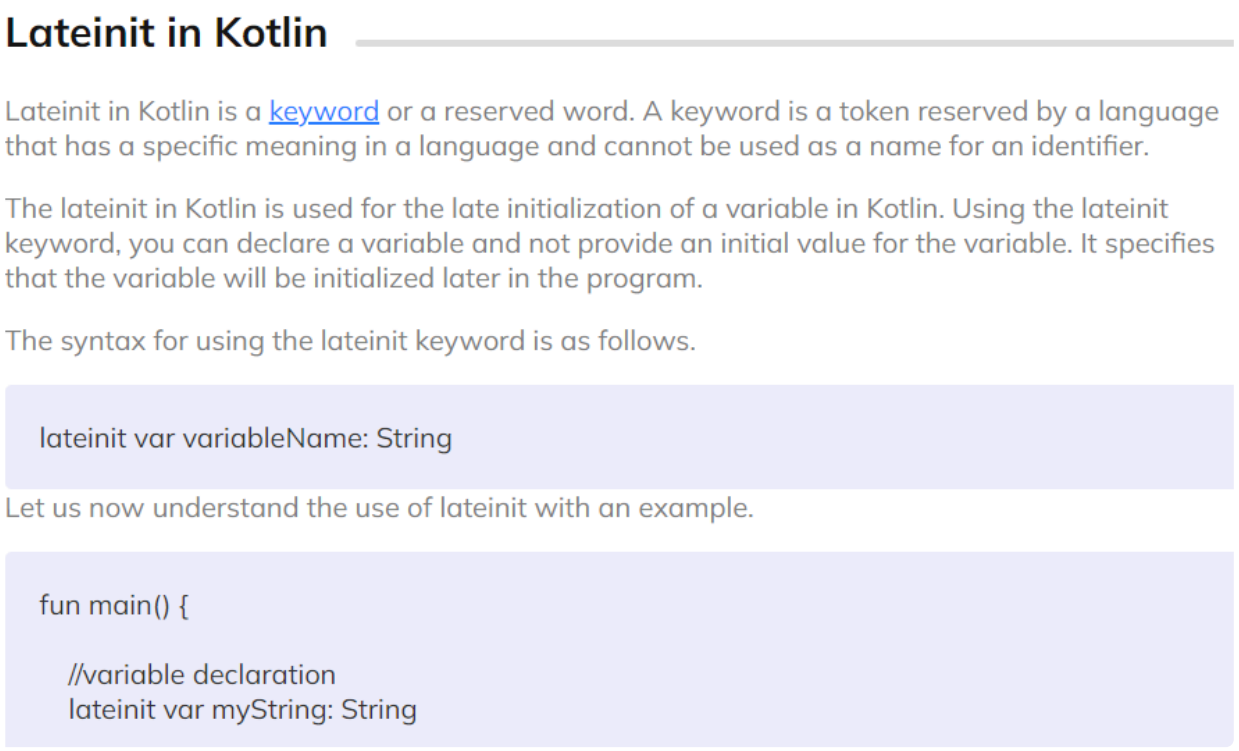
lekin aise aise kiye to ni hoga jbki JS me possible hai

define kr diye to shi chalega



Note - Ek lateinit kr ke bhi type hota hai jaise ki var val ko turant initialize krna pdta hai agr hm chah rhe ki baad me kre to lateinit ka use kr

skte hai



Lateinit in Kotlin

Lateinit in Kotlin is a keyword or a reserved word. A keyword is a token reserved by a language that has a specific meaning in a language and cannot be used as a name for an identifier.

The lateinit in Kotlin is used for the late initialization of a variable in Kotlin. Using the lateinit keyword, you can declare a variable and not provide an initial value for the variable. It specifies that the variable will be initialized later in the program.

The syntax for using the lateinit keyword is as follows.

```
lateinit var variableName: String
```

Let us now understand the use of lateinit with an example.

```
fun main() {

    //variable declaration
    lateinit var myString: String
```

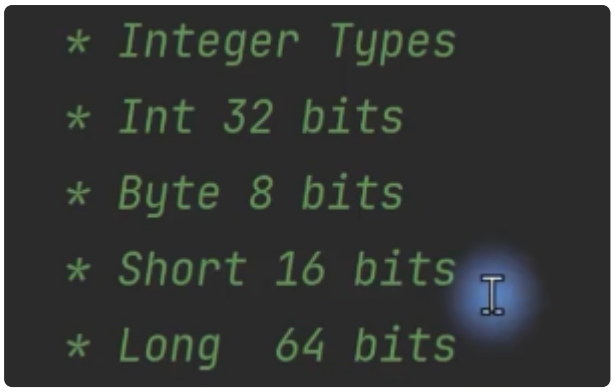
Data types

```
val myNum: Int = 5                  // Int
val myDoubleNum: Double = 5.99      // Double
val myLetter: Char = 'D'            // Char
val myBoolean: Boolean = true       // Boolean
val myText: String = "Hello"        // String
```

Integer data type

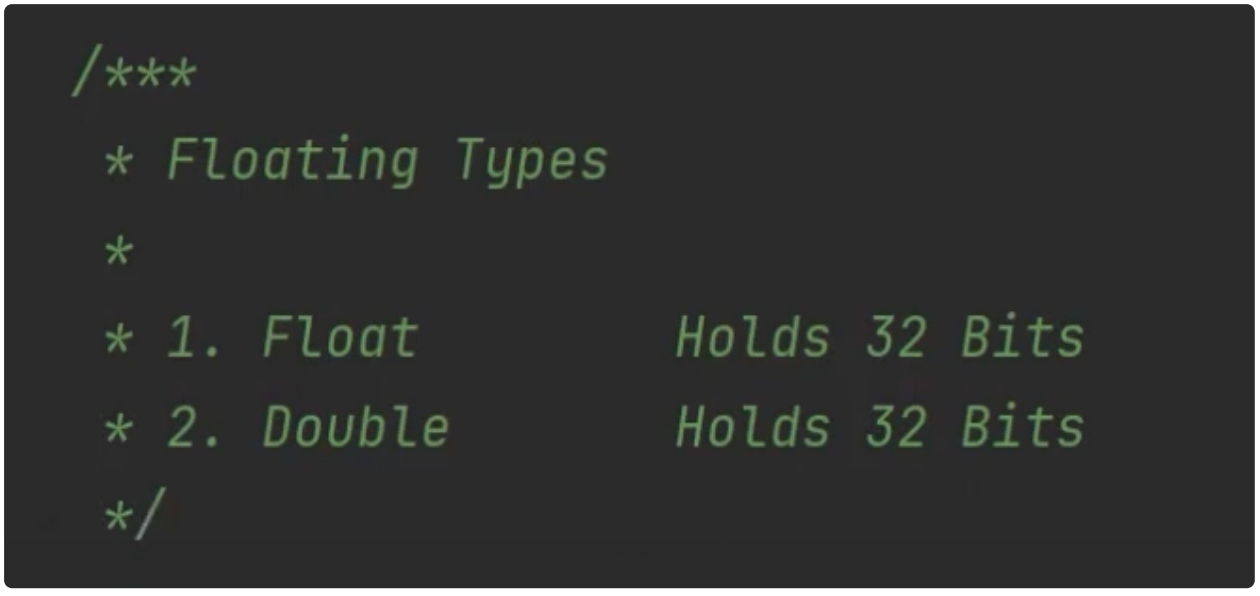
dekho hm app bna rhe hai to app ka size bhi bhut matter krta hai to uske liye hme shi data

types use krne hinge

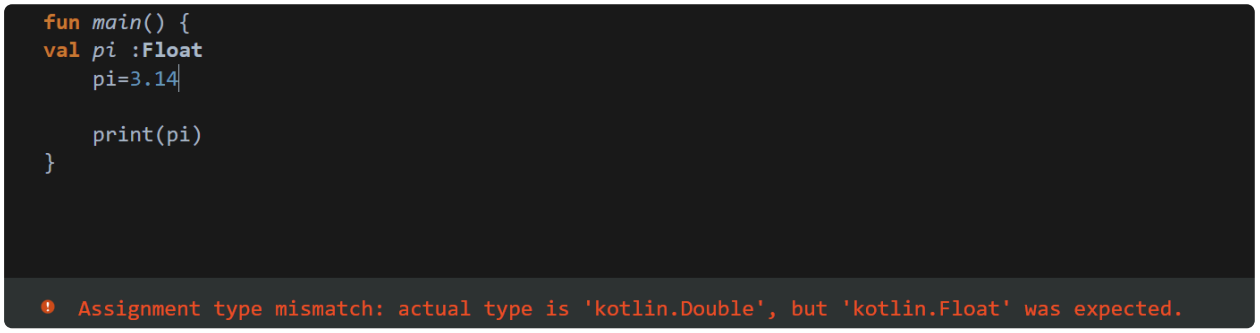


By default agr hmne data type ni batya kisi number ka to wo by default int leta hai

Floating data type

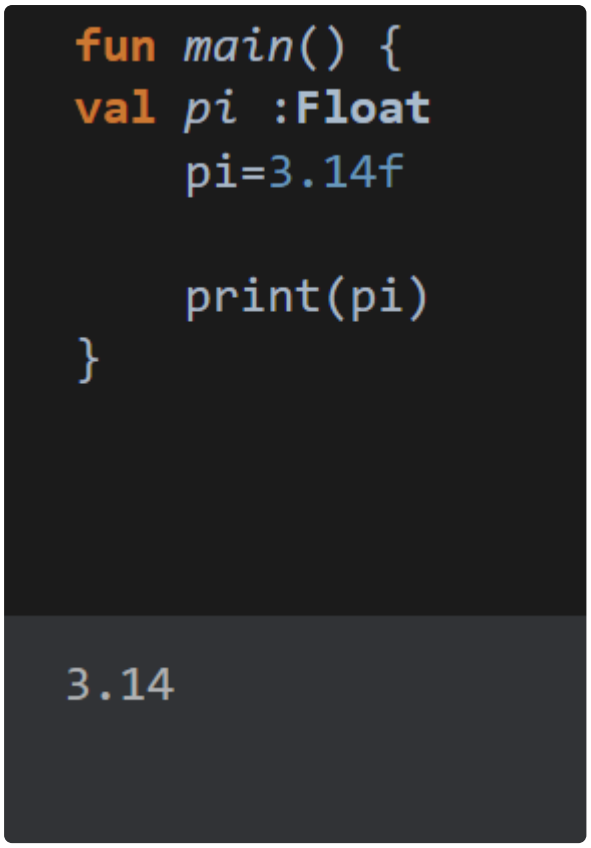


by Default Double



dekho float le hi ni rha wo isko explicitly

batyenge ki bhai tum float ho



F ya f lga ke

Operators

Operator	Name	Description	Example
+	Addition	Adds together two values	x + y
-	Subtraction	Subtracts one value from another	x - y
*	Multiplication	Multiplies two values	x * y
/	Division	Divides one value from another	x / y
%	Modulus	Returns the division remainder	x % y
++	Increment	Increases the value by 1	++x
--	Decrement	Decreases the value by 1	--x

Kotlin Comparison Operators

Comparison operators are used to compare two values, and returns a `Boolean` value: either `true` or `false`.

Operator	Name	Example
==	Equal to	x == y
!=	Not equal	x != y
>	Greater than	x > y
<	Less than	x < y
>=	Greater than or equal to	x >= y
<=	Less than or equal to	x <= y

Kotlin Logical Operators

Logical operators are used to determine the logic between variables or values:

Operator	Name	Description	Example
&&	Logical and	Returns true if both statements are true	x < 5 && x < 10
||	Logical or	Returns true if one of the statements is true	x < 5 || x < 4
!	Logical not	Reverse the result, returns false if the result is true	

Conditional Statement

Syntax

```
if (condition) {
  // block of code to be executed if the condition is true
} else {
  // block of code to be executed if the condition is false
}
```

Kotlin else if

Use `else if` to specify a new condition if the first condition is `false`.

Syntax

```
if (condition1) {
  // block of code to be executed if condition1 is true
} else if (condition2) {
  // block of code to be executed if the condition1 is false and condition2 is true
} else {
  // block of code to be executed if the condition1 is false and condition2 is false
}
```

When (same as switch case)

Important arrow operator use krna hoga

Kotlin when

Instead of writing many `if..else` expressions, you can use the `when` expression, which is much easier to read.

It is used to select one of many code blocks to be executed:

Example

Use the weekday number to calculate the weekday name:

```
val day = 4

val result = when (day) {
  1 -> "Monday"
  2 -> "Tuesday"
  3 -> "Wednesday"
  4 -> "Thursday"
  5 -> "Friday"
  6 -> "Saturday"
  7 -> "Sunday"
  else -> "Invalid day."
}
println(result)
```

ab yaha pe dekho sbke liye ek ek condition to use ni kr skte to iske liye hm range function ka use krte hi jaise is age<60 etc

Range operator

range..operator

Kotlin program of character range using (..) operator –

```
fun main(args : Array<String>){

    println("Character range:")
    // creating character range
    for(ch in 'a'..'e'){
        println(ch)
    }
}
```

Output:

```
Character range:
a
b
c
d
e
```

rangeTo()

rangeTo() function

It is similar to (..) operator. It will create a range upto the value passed as an argument. It is also used to create range for integers as well as characters.

Kotlin program of integer range using rangeTo() function –

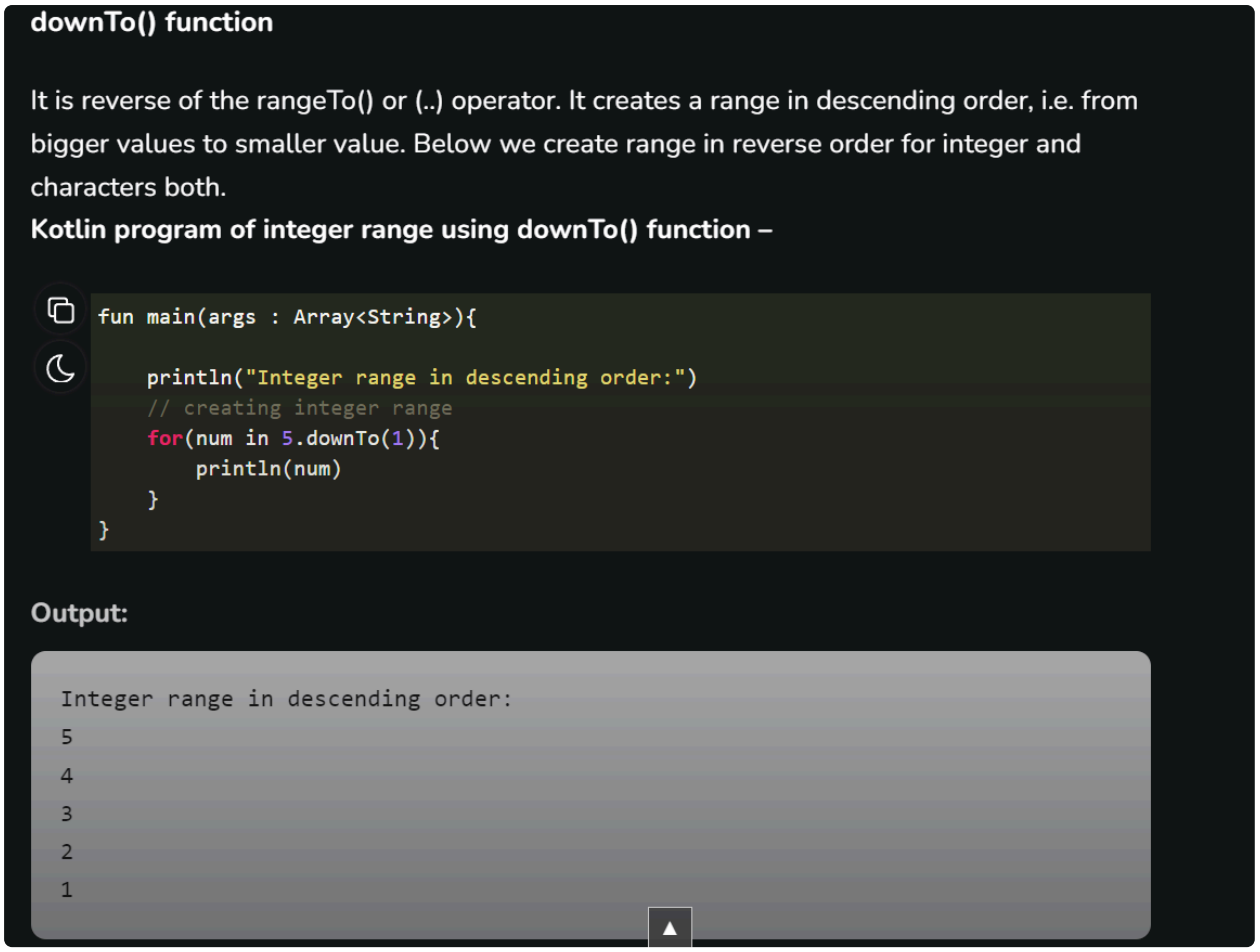
```
fun main(args : Array<String>){

    println("Integer range:")
    // creating integer range
    for(num in 1.rangeTo(5)){
        println(num)
    }
}
```

Output:

```
Integer range:
1
2
3
4
5
```

downTo()



ulta hota hai rang ka descending order me hota hai

Why no null pointer error in kotlin

Kotlin null ko support hi ni krta hme kisi bhi variable to initialize krna bhut jaruri hai
ex

```
fun main() {
val name:String


    print(name)
}
```

```
Variable 'name' must be initialized.
```

aur hm null se init bhi ni kr skte

```
fun main() {
val name:String
    name=null

    print(name)
}
```

```
Null cannot be a value of a non-null type 'kotlin.String'.
```

agr hmko null se init hi krna hai to dataType declaration ke baad "?"
ka use krenge

```kotlin
fun main() {
val name:String?
    name=null

    print(name)
}
```

```
null
```

Known as safe call

Note - ? known as elvis operator in kotlin

```kotlin
fun main() {
val name:String?
    name=null

    print(name.length)
}
```

```
Only safe (?.) or non-null asserted (!!.) calls are allowed on a nullable receiver of type 'kotlin.Nothing?'
```

AGR is se deal krna hai to wahi elvis operator

ka use krenge

```
fun main() {
val name:String?
    name=null

    print(name?.length)
}
```

```
null
```

Functions in kotlin

with parameter

```
fun myFunction(fname: String, age: Int) {
  println(fname + " is " + age)
}

fun main() {
  myFunction("John", 35)
  myFunction("Jane", 32)
  myFunction("George", 15)
}

// John is 35
// Jane is 32
// George is 15
```

Function with return values

Example

A function with two `Int` parameters, and `Int` return type:

```
fun myFunction(x: Int, y: Int): Int {
  return (x + y)
}

fun main() {
  var result = myFunction(3, 5)
  println(result)
}

// 8 (3 + 5)
```

`fun myFun(x:Int):Int{ this last is return type }`

OOPS

Create a Class

To create a class, use the `class` keyword, and specify the name of the class:

Example

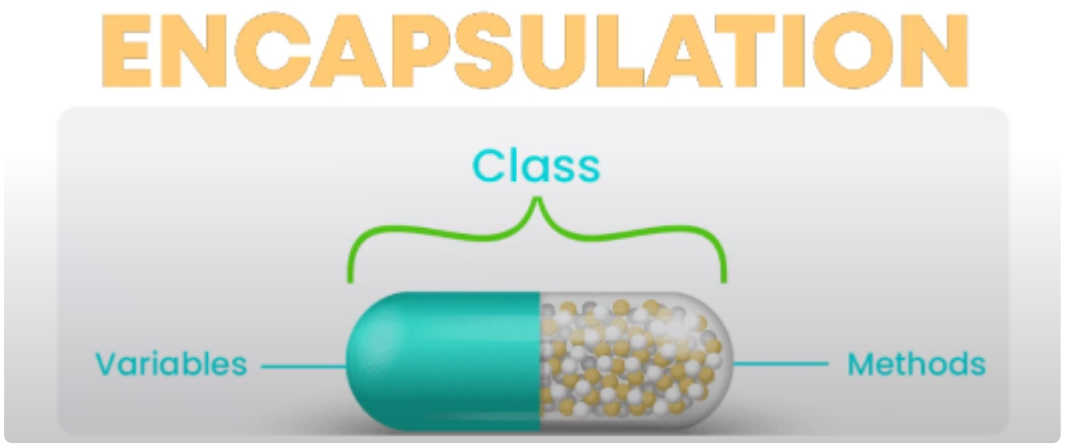
Create a **Car** class along with some **properties** (brand, model and year)

```
class Car {
  var brand = ""
  var model = ""
  var year = 0
}
```

Example

```
// Create a c1 object of the Car class
val c1 = Car()

// Access the properties and add some values to it
c1.brand = "Ford"
c1.model = "Mustang"
c1.year = 1969

println(c1.brand)   // Outputs Ford
println(c1.model)   // Outputs Mustang
println(c1.year)    // Outputs 1969
```

Encapsulation



Access modefiers in kotlin

- If you don't use a visibility modifier, `public` is used by default, which means that your declarations will be visible everywhere.
- If you mark a declaration as `private`, it will only be visible inside the file that contains the declaration.
- If you mark it as `internal`, it will be visible everywhere in the same module.
- The `protected` modifier is not available for top-level declarations.

- encapsulation is achieved through access modifiers such as `private`, `protected`, `internal` and public.

Constructor in kotlin

1. Default constructor (primary constructor)- ye khud se call ho jata hai isko call ni krte

```
class Car {
    var brand = ""
    var model = ""
    var year = 0
}
```

jaise ye yaaha carr ko car() aise ni likkha hai

2. Paramatrize constructor (secondary constructor)

   agr parameter pass krna ho to

```
class cal(var a :Int , var b:Int){

    fun add(){
        println(a+b)
    }
}
fun main(){
    var op = cal(5,7)

    op.add()
}
```

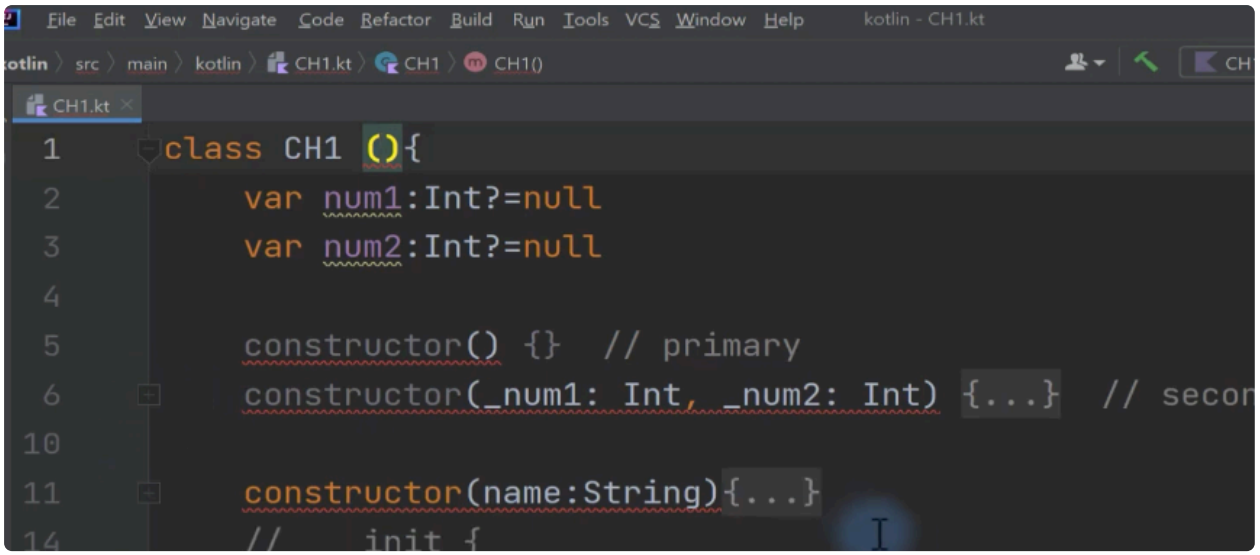
yaha dekho parameter pass kiya hai to round braket ka use kr rhe hai

-- Ye krna ek achi practice ni hai iske liye hm construstor ka use krte hei

```
class cal{
    var a :Int? =null
    var b:Int?=null
   constructor(){}; // Primary
constructor ye tb call hoga jb hmko
nomral object banan hoga jaise var
a1=cal();
    constructor(y : Int =5 , z:Int=8){
        this.a = y
        this.b=z

    }//Secondary constructor agar
value deke pass krenge t0 ye wala call
hoga
    fun add(){
        println(a!!+b!!) // ye to null
value handle krne ke liye hai
    }
}
fun main(){
    var op = cal()
    op.add()

}
```

We can make many constructors like this

Important



agr upr hi CH1() aise define kr diya to tum niche constructor ko define ni kr skte

Inheritance

Are ye wahi hai jo ki parent class ki properties le leta hai
Syntax

```
open class baseClass (x:Int ) {
      ..........
}
class derivedClass(x:Int) : baseClass(x) {
     ...........
}
```

java me hm simply

```
class Employee{
 float salary=40000;
}
class Programmer extends Employee{
 int bonus=10000;
```

ye krte the but yaaha hmko

- Extends ki jgh : lgana hoga
- aur parent class ho open krna hoga joki java me ni krna hota tha

```
open class Father{
    var car = " BMW"
```

```
}
class Son : Father(){

    fun carName(){
        print(car)
    }

}
fun main(){
    var obj = Son();
    obj.carName()
    // yaha dekho father ki car son
use kr pa rha hai
}
```

Types of Inheritance

- Single level Inheritance

```
open class Father{
    var car = " BMW"


}
class Son : Father(){

    fun carName(){
        print(car)
    }
```

- MultiLevel Inhertance

```
open class Father{
    var car = " BMW"


}
open class Son : Father(){
    var bike = "hero"
    fun carName(){
        print(car)
    }

}
class Grand : Son(){
    fun bikeName(){
        print(bike)
    }
}
fun main(){
    var obj = Grand();
    obj.bikeName()
}
```

- Hierarchy Inheritance - ek father ke do bachee

```
open class Father{
    var car = " BMW"


}
open class Son : Father(){
    var bike = "hero"
    fun carName(){
        println(car)
    }

}
class Son2 : Father(){
    fun carName2(){
        println(car)
    }
}
fun main(){
    var obj = Son();
     obj.carName()
    var obj2 = Son2();
       obj2.carName2()
}
```

dono papa ki hi gadi use kr rhe

- Hybrid inheritance - ye dono ka mixup hota hai heirarchy aur multiple dono
- multiple inheritance(does not support by kotlin and java)

Poly-Morphism

The word polymorphism means having many forms. In simple words, we can define polymorphism as the ability of a message to be displayed in more than one form.

1. Compile-time Polymorphism
   Function overLoading parameters change kr ke

```
open class Father{
    var car = " BMW"
     fun car(){
print("hello")
    }
     fun car( a : Int){
print("hi")
    }

}
```

2. Runtime Polymorphism

```
open class Sup{
    open fun method1(){
        println("printing method 1 from inside Sup")
    }
    fun method2(){
        println("printing method 2 from inside Sup")
    }
}

class Sum:Sup(){
    override fun method1(){
        println("printing method 1 from inside Sum")
    }
}
```

yaha override function ka use krte hai
important - jis bhi function ko over ride krna hai usko open krna hoga

Abstraction

- Agar class abstract hai to minimum uska ek function abstract hona chahiye
- agr koi function abstract hai to class bhi abstract honi chahiye

```
class Father{
    var car = " BMW"
     abstract fun car(){
print("hello")
    }
     fun car( a : Int){
print("hi")
    }

}

fun main(){
    var obj = Father()
     obj.car(5)
```

```
Abstract function 'car' in non-abstract class 'Father'.
Function 'car' with a body cannot be abstract.
```

- Abstract class ka object ni bana skte
- jb object bana ni skte to uske function ya memebers ko use krne ke liye child class unko inherit krwate hai

```
abstract class Father{
    var car = " BMW"
     abstract fun car()
     fun car1(){
print("hi")
    }

}

 class Son : Father(){
     override fun car(){
         print("hello from abstract")
     }
 }
```

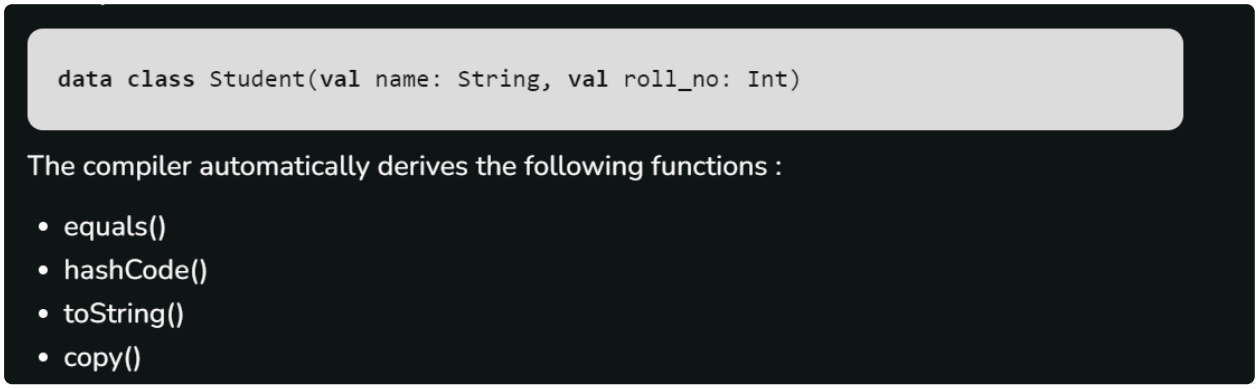
```
hello from abstract
```

- agr kisi bhi function ki body hai to usko abstract ni kr skte

```
abstract function car(){} xxxxxxxx
Aisa ni kr skte
abstract function car() ✅✅✅✅✅✅
aise krna hogaq without body
```

- agar function abstract hai to wo age jake inherit hoga ye default nature hai to oepn lagao na lagao jyada anatar ni hai

Data classes(Java ❌ Kotlin ✅)



agr hm kisi bhi class ko data class define kr dete hia to kotlin already smjh jata hai ki ye bs data ke kaam ayegi aur wo kuch function apne taraf se hi provide kr deti hia

- equals()
- hashCode()
- toString()
- copy()
  ye function java me khud se define krne pdte the
  important
  - The primary constructor needs to have at least one parameter.
- All primary constructor parameters need to be marked as *val* or *var*.
- Data classes cannot be abstract, open, sealed or inner.
- Data classes may only implement interfaces.

Sealed and Enum class in Kotlin

In programming, sometimes there arises a need for a type to have only certain values. To accomplish this, the concept of enumeration was introduced. Enumeration is a named list of

constants. In Kotlin, like many other programming languages, an ****enum**** has its own specialized type, indicating that something has a number of possible values. Unlike Java enums, Kotlin enums are ****classes****.

```
enum class DAYS{
   SUNDAY,
   MONDAY,
   TUESDAY,
   WEDNESDAY,
   THURSDAY,
   FRIDAY,
   SATURDAY
}
```

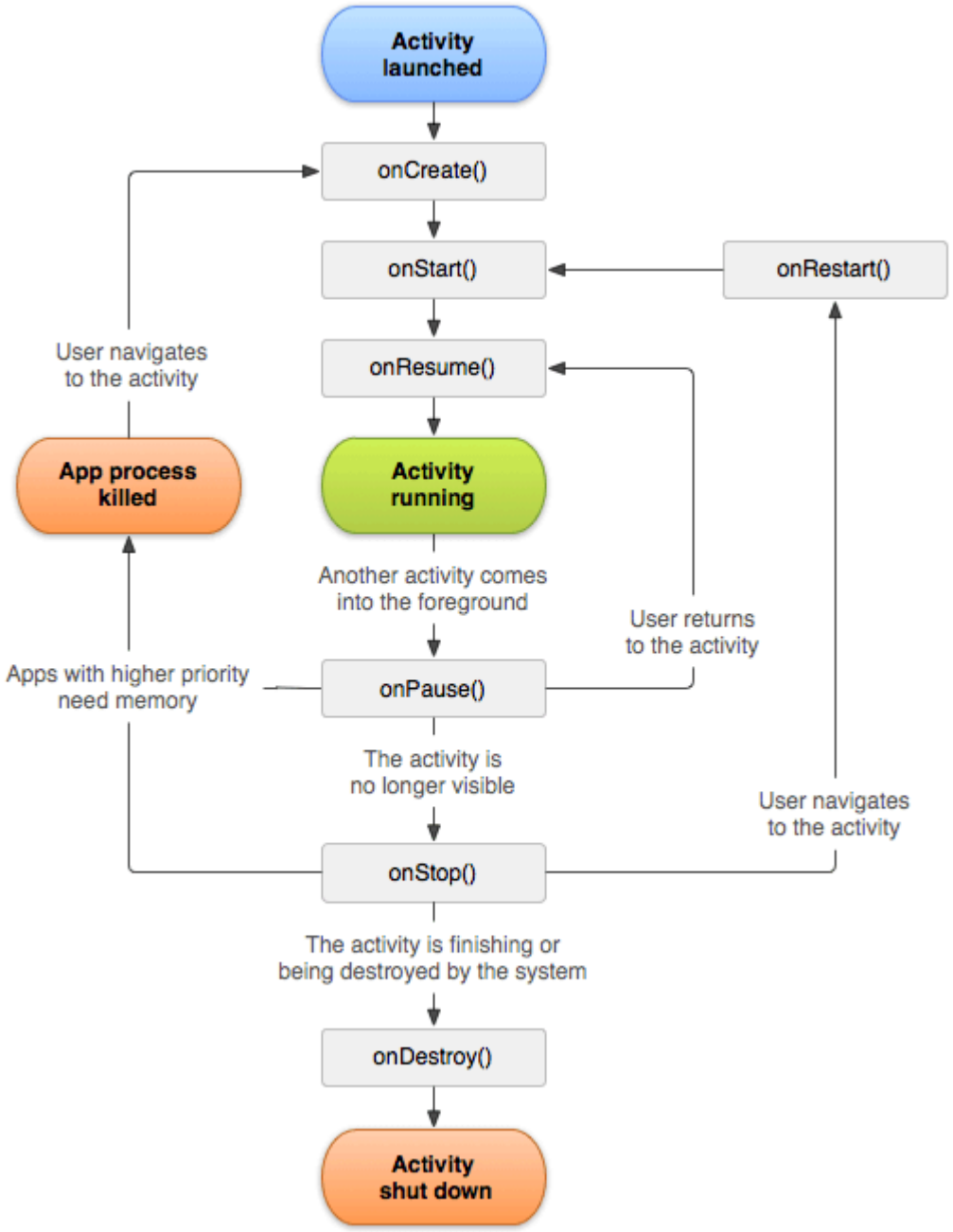
Properties –

1. **ordinal:** This property stores the ordinal value of the constant, which is usually a zero-based index.
2. **name:** This property stores the name of the constant.

Methods –

1. **values**: This method returns a list of all the constants defined within the enum class.
2. **valueOf**: This methods returns the enum constant defined in enum, matching the input string. If the constant, is not present in the enum, then an IllegalArgumentException is thrown.

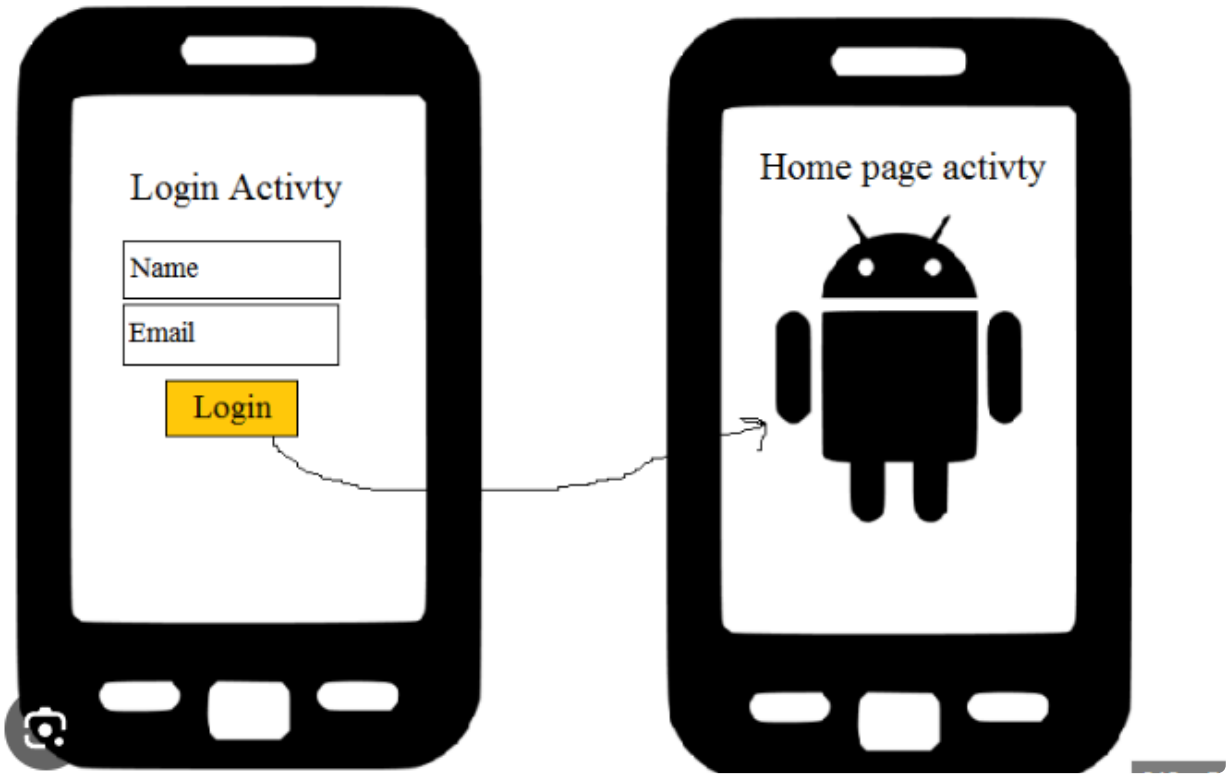
Android Activity Life Cycle



there are 7 methods in Activity life cycle

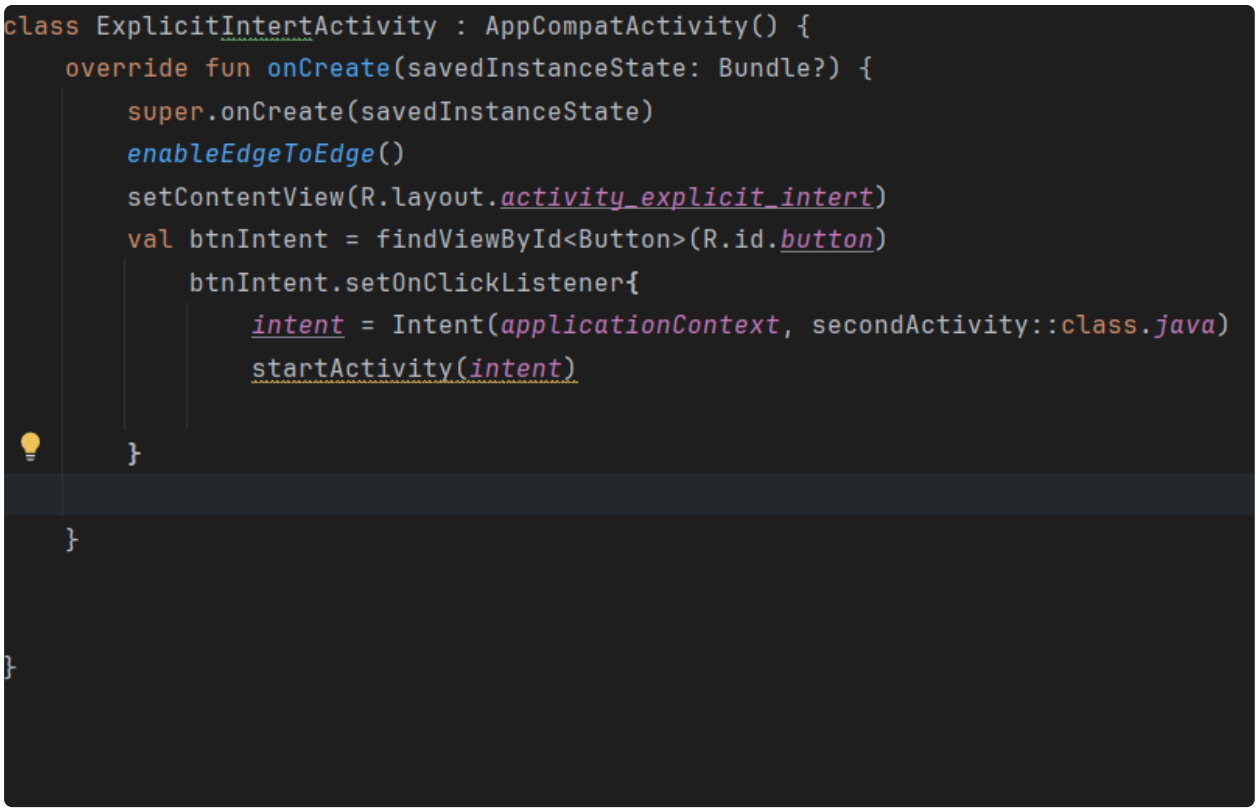
Intent

agr ek activity se dure activity me jana hai to intent ka use krte hai



Types of intent

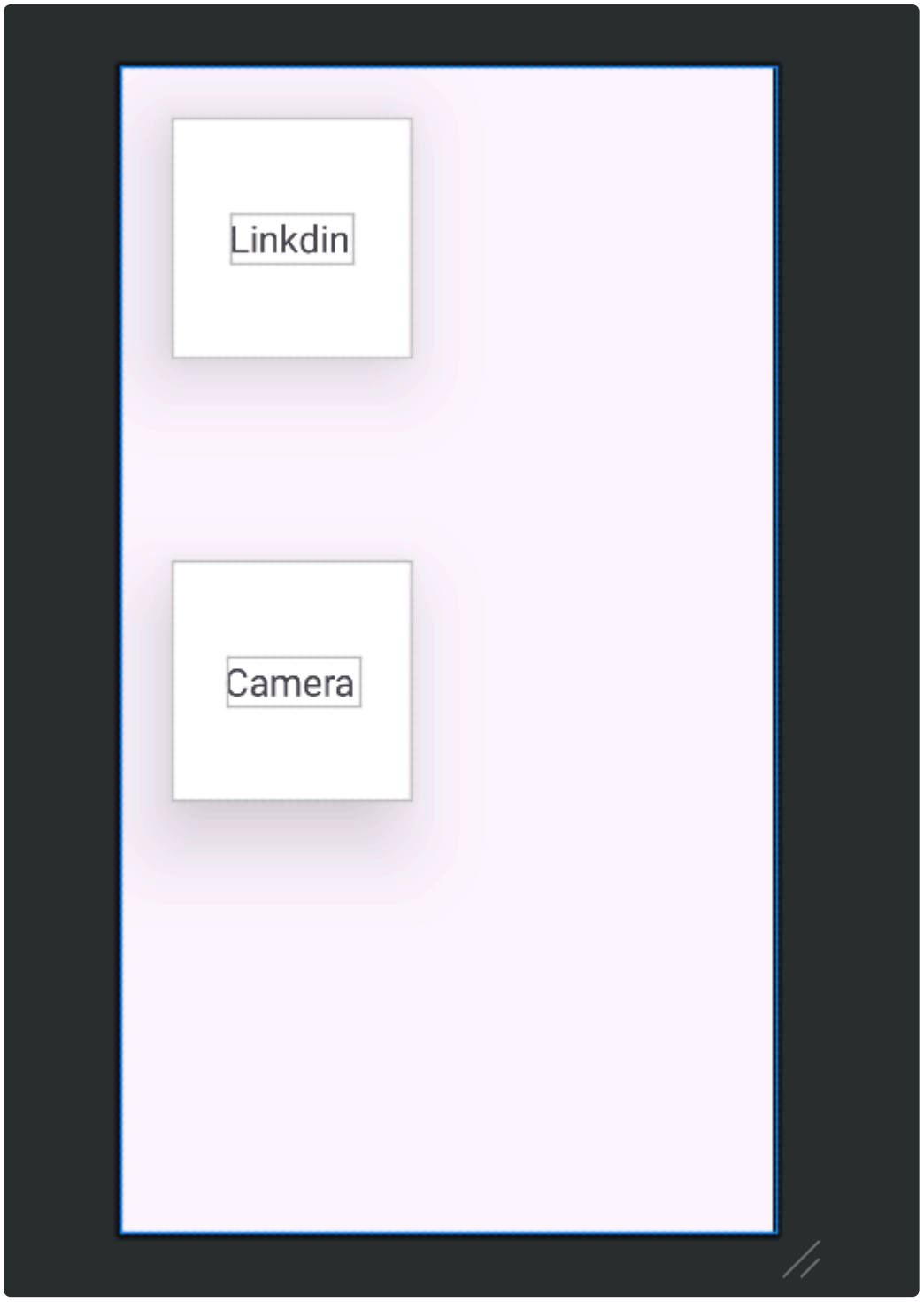
1. Implicit Intent - jo dusre app pe redirect kre
2. Explicit intent - same app me activity change



```
class ExplicitIntertActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_explicit_intert)
        val btnIntent = findViewById<Button>(R.id.button)
            btnIntent.setOnClickListener{
                intent = Intent(applicationContext, secondActivity::class.java)
                startActivity(intent)

        }

    }


}
```

ek activity se dure me jane ka code kisi button ko click kr ke

ImplicitIntent (dusre app me jana)



Backend

```kotlin
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)

        val instaButton = findViewById<CardView>(R.id.Insta)
        val cameraButton = findViewById<CardView>(R.id.camera)
        // yaha dono buttons ko access kra

        instaButton.setOnClickListener {
            val intent = Intent(Intent.ACTION_VIEW) ////  koi action dekhne ke liye ACTION_VIEW use kra
            intent.data = Uri.parse( uriString: "https://www.linkedin.com/in/akhilesh-singh-maurya-500b91257/") // Kis web page pe jana uski link
            startActivity(intent) // ye intent ko start kra
        }
        cameraButton.setOnClickListener {
            val intent = Intent(
                MediaStore.ACTION_IMAGE_CAPTURE // camera ko open krne ke liye ACTION_IMAGE_CAPTURE use kra mediaStore me

            )

            startActivity(intent) // ye intent ko start kra
        }

    }
}
```

WebView

```kotlin
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
        val webViewVariable = findViewById<WebView>(R.id.webView)
        webViewSetup(webViewVariable);

    }
    private fun webViewSetup(webView: WebView){
        webView.webViewClient = WebViewClient() // syntax hai aise hi likhna hoga webview ko
        webView.apply {

            settings.javaScriptEnabled = true
            settings.safeBrowsingEnabled=true
            loadUrl( url: "https://www.manparth.com/")
        }

    }


}
```

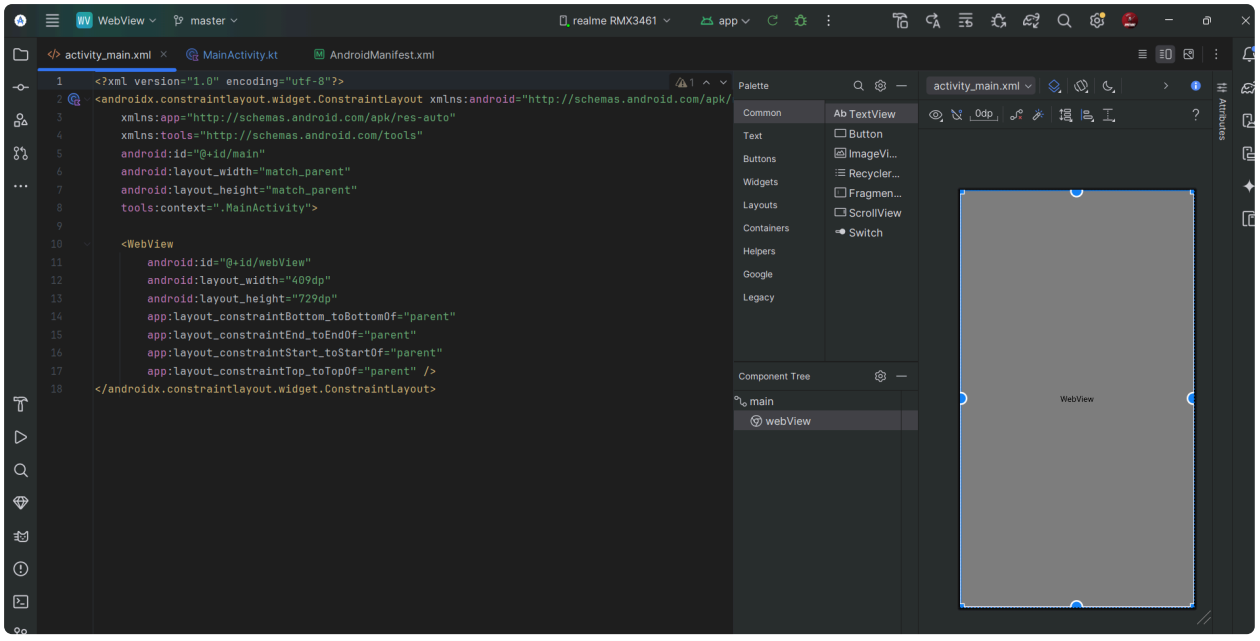
Web view use krne ke liye ek important kaam hota hai user ko internet ki permision deni hoti hai menifest se

activity_main.xml MainActivity.kt AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <uses-permission android:name="android.permission.INTERNET" />
```

YE WALI

activity file



Passing data form one screen to other

Dekho kuch bhi idhar se udhar pass krne ke liye hmko key ki jruruat lgti hai jo ki global ki mtlb har activity me access ho jaye to iske liye hmko global ki bnanani pdegi

java me to static keyword use kr ke bna dete the global mgr kotlin me static ni hota

Static variables

When a variable is declared as static, then a single copy of the variable is created and shared among all objects at the class level. Static variables are, essentially, global variables. All instances of the class share the same static variable.

Important points for static variables:

- We can create static variables at the class level only. See here
- static block and static variables are executed in the order they are present in a program.

Companion Object in Kotlin

- note -Static ki jgh Companion object ka use krte hai kotlin me

To pass data from ONE Activity

```
    companion object {
        // Members of the companion object
        val KEY ="com.example.makeyourordernow.MainActivity.KEY" // this done so ki unique naam rhe ye bas naam hai koi syntax ni hai
        }
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)


        val btnOrder = findViewById<Button>(R.id.btnOrder)
        val et1=findViewById<EditText>(R.id.et1)
        val et2=findViewById<EditText>(R.id.et2)
        val et3=findViewById<EditText>(R.id.et3)
        val et4=findViewById<EditText>(R.id.et4)

        btnOrder.setOnClickListener{
            val orderList = et1.text.toString() + " " + et2.text.toString() + " " + et3.text.toString() + " " + et4.text.toString()
        intent = Intent( packageContext: this,OrderActivity::class.java)
            //putExtra data us side le ke jayega but key aur kya le janan hai wo batana hoga
        intent.putExtra(KEY,orderList)
            startActivity(intent)
        }

    }
}
```

mainly data pass krne ke lie hm putExtra ka use krte hai

```
intent.putExtra(KEY,orderList)
```

isko ek key dete hai jo ki unique hoti hai aur jo data pass krna hai wo dete hai is case me hmne order list diya hai jo editText se extract kiya hai
jaise react e.value kr ke get krte the na yaha bhi kuch aise hi krte hai

```
val orderList = et1.text.toString() + " " + et2.text.toString() + " " + et3.text.toString() + " " + et4.text.toString()
```

yaha ek variable bnaya hai et1(editText ki id).text kr ke extract lr liya hai aur put extra me pass kr diya hai

TO get the passed data in other activity

waha ek text view bnaya hai aur ek variable leke
"intent.getStringExtra(Jis bhi activity se aa rha ho uska naam.key)"
aise leke access kr de

```
val odersOfCoustomer=intent.getStringExtra(MainActivity.KEY)
```

Aise odersOfCoustmer me value li aur fir

```
tvOrder.text= "Coustmer order is : \n ${odersOfCoustomer.toString()}"
```

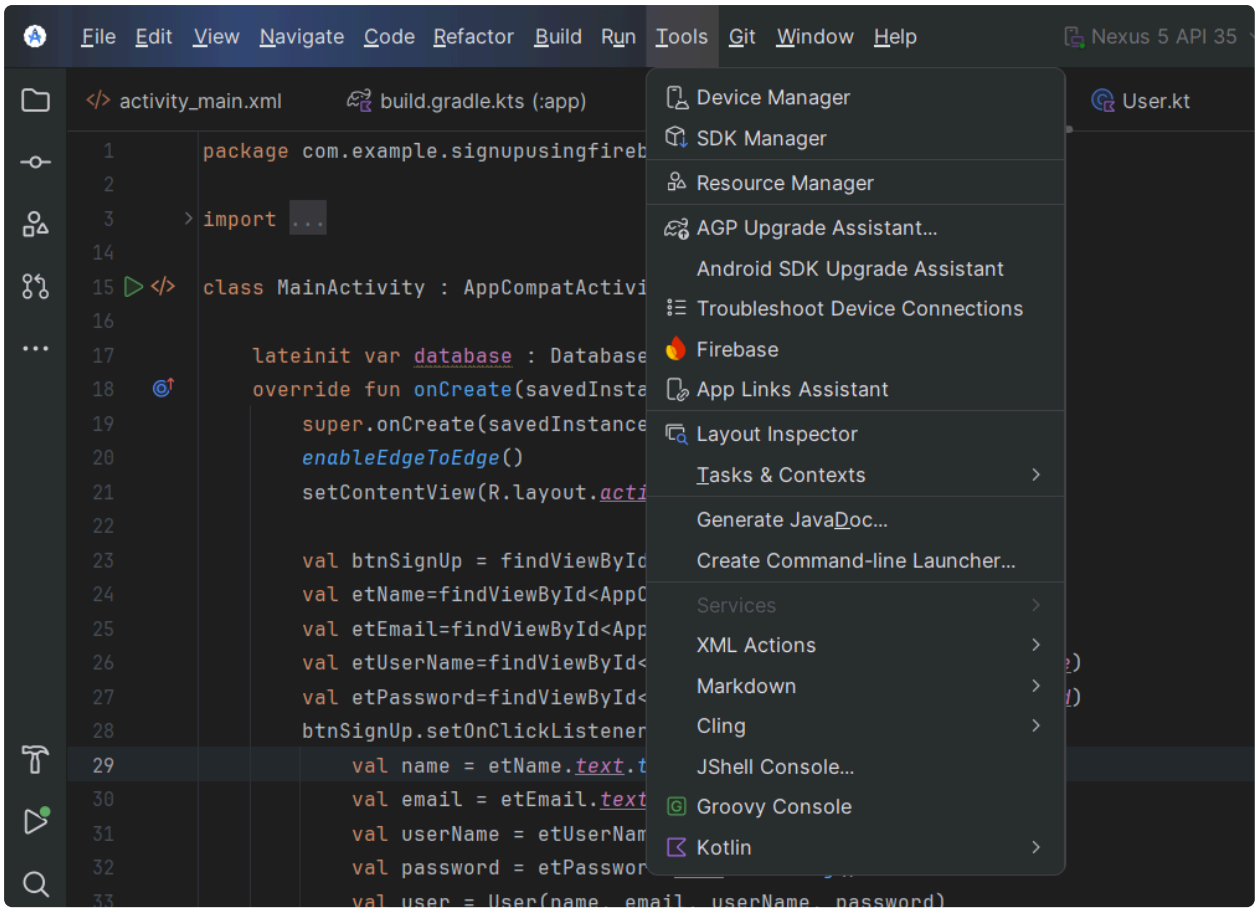
jo text view tha usme daal diya .text property hoti hai change kr ne ki

That's It

Setting data in fire base

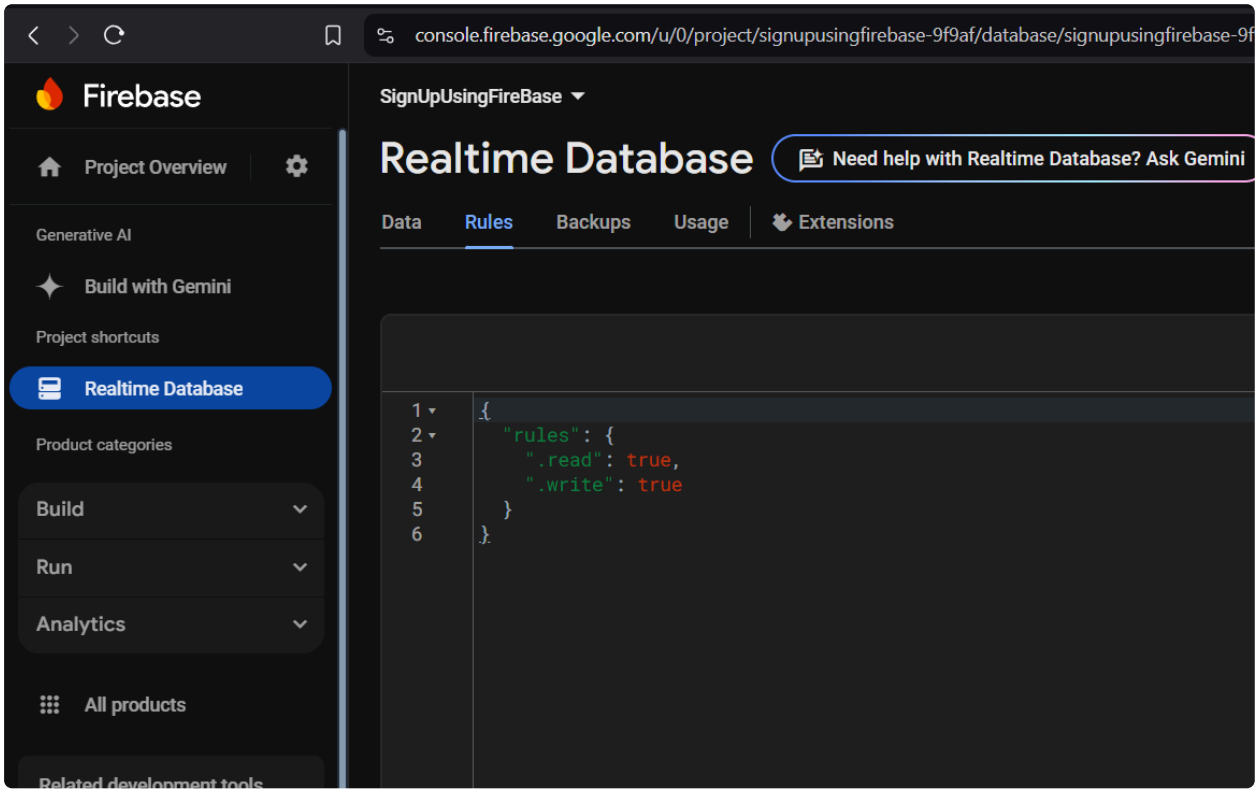
Step 1. Fire base pe account bnaye

Step 2. Android studio ko link kre Fire base se



yaha se link krenge connect kr ke

step 3 .



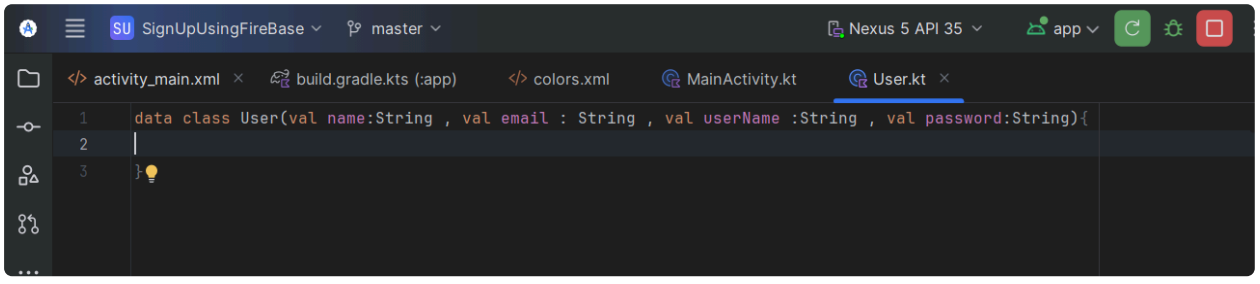
fir fire base me jake real time database me jake rules ko true kr denge

Step 4 . User ka sara data le lenge TextInput se

```
val btnSignUp = findViewById<AppCompatButton>(R.id.btnSignUp)
val etName=findViewById<AppCompatEditText>(R.id.etName)
val etEmail=findViewById<AppCompatEditText>(R.id.etMail)
val etUserName=findViewById<AppCompatEditText>(R.id.etUserName)
val etPassword=findViewById<AppCompatEditText>(R.id.etPassword)
btnSignUp.setOnClickListener{
    val name = etName.text.toString()
    val email = etEmail.text.toString()
    val userName = etUserName.text.toString()
    val password = etPassword.text.toString()
    val user = User(name, email, userName, password)
    etName.text?.clear()
    etEmail.text?.clear()
    etPassword.text?.clear()
    etUserName.text?.clear()
```

jaise yaha liya hai

Step 5. Ek class bnaynge jisme hm sara data store krenge



Bs class bna ke chod denge aur jo value enter karani hai usko as a parameter pass kr denge

Step 6. Ek lateinit variable define krenge jiska data type hoga DatabaseRefrence

```
lateinit var database : DatabaseReference
```

Step 7.

```
database= FirebaseDatabase.getInstance().getReference( path: "Users")
database.child(name).setValue(user).addOnSuccessListener {
    Toast.makeText( context: this, text: "User Registered", Toast.LENGTH_SHORT).show()
}.addOnFailureListener {
    Toast.makeText( context: this,  text: "Failed", Toast.LENGTH_SHORT).show()
}
```

yehi code hai bs itna sa fire base me data send krne ke liye

Explanation of the code

```
This code snippet is part of an
Android application using Firebase
Realtime Database to store user
information. Here's a step-by-step
explanation:

1.
**FirebaseDatabase.getInstance().getRe
ference("Users")**:

    - This retrieves a reference to
the Firebase Realtime Database and
```

points to the "Users" node. This is where user data will be stored.
2. **database.child(name)**:

    - This creates a child node under "Users" with the key `name`. The `name` is typically a unique identifier for the user, like a username or user ID.
3. **.setValue(user)**:

    - This sets the value of the newly created child node to `user`. `user` is an object containing the user details (like name, email, etc.).
4. **addOnSuccessListener { ... }**:

    - This is a callback that runs when the data is successfully written to the database.
    - Inside the callback:
        - A toast message is shown saying "User Registered".

```
5. **addOnFailureListener { ... }**:

    - This is a callback that runs if there is an error while writing the data to the database.
    - Inside the callback:
        - A toast message is shown saying "Failed".

### In short:

- **On success**: The user is registered, and a success message is displayed.
- **On failure**: An error message is displayed.
```

Complete code

```
package com.example.signupusingfirebase
```

```
import User
import android.os.Bundle
import android.widget.Toast
import
androidx.activity.enableEdgeToEdge
import
androidx.appcompat.app.AppCompatActivi
ty
import
androidx.appcompat.widget.AppCompatBut
ton
import
androidx.appcompat.widget.AppCompatEdi
tText
import androidx.core.view.ViewCompat
import
androidx.core.view.WindowInsetsCompat
import
com.google.firebase.database.DatabaseR
eference
import
com.google.firebase.database.FirebaseD
atabase

class MainActivity :
AppCompatActivity() {
```

```
    lateinit var database :
DatabaseReference
    override fun
onCreate(savedInstanceState: Bundle?)
{

super.onCreate(savedInstanceState)
        enableEdgeToEdge()

setContentView(R.layout.activity_main)

        val btnSignUp =
findViewById<AppCompatButton>
(R.id.btnSignUp)
        val
etName=findViewById<AppCompatEditText>
(R.id.etName)
        val
etEmail=findViewById<AppCompatEditText
>(R.id.etMail)
        val
etUserName=findViewById<AppCompatEditT
ext>(R.id.etUserName)
        val
etPassword=findViewById<AppCompatEditT
```

```
ext>(R.id.etPassword)
        btnSignUp.setOnClickListener{
            val name =
etName.text.toString()
            val email =
etEmail.text.toString()
            val userName =
etUserName.text.toString()
            val password =
etPassword.text.toString()
            val user = User(name,
email, userName, password)
            etName.text?.clear()
            etEmail.text?.clear()
            etPassword.text?.clear()
            etUserName.text?.clear()
            database=
FirebaseDatabase.getInstance().getRefe
rence("Users")

database.child(name).setValue(user).ad
dOnSuccessListener {

Toast.makeText(this,"User Registered",
Toast.LENGTH_SHORT).show()
            }.addOnFailureListener {
```

```
                    Toast.makeText(this,
"Failed", Toast.LENGTH_SHORT).show()
                }


            }
        }
}
```

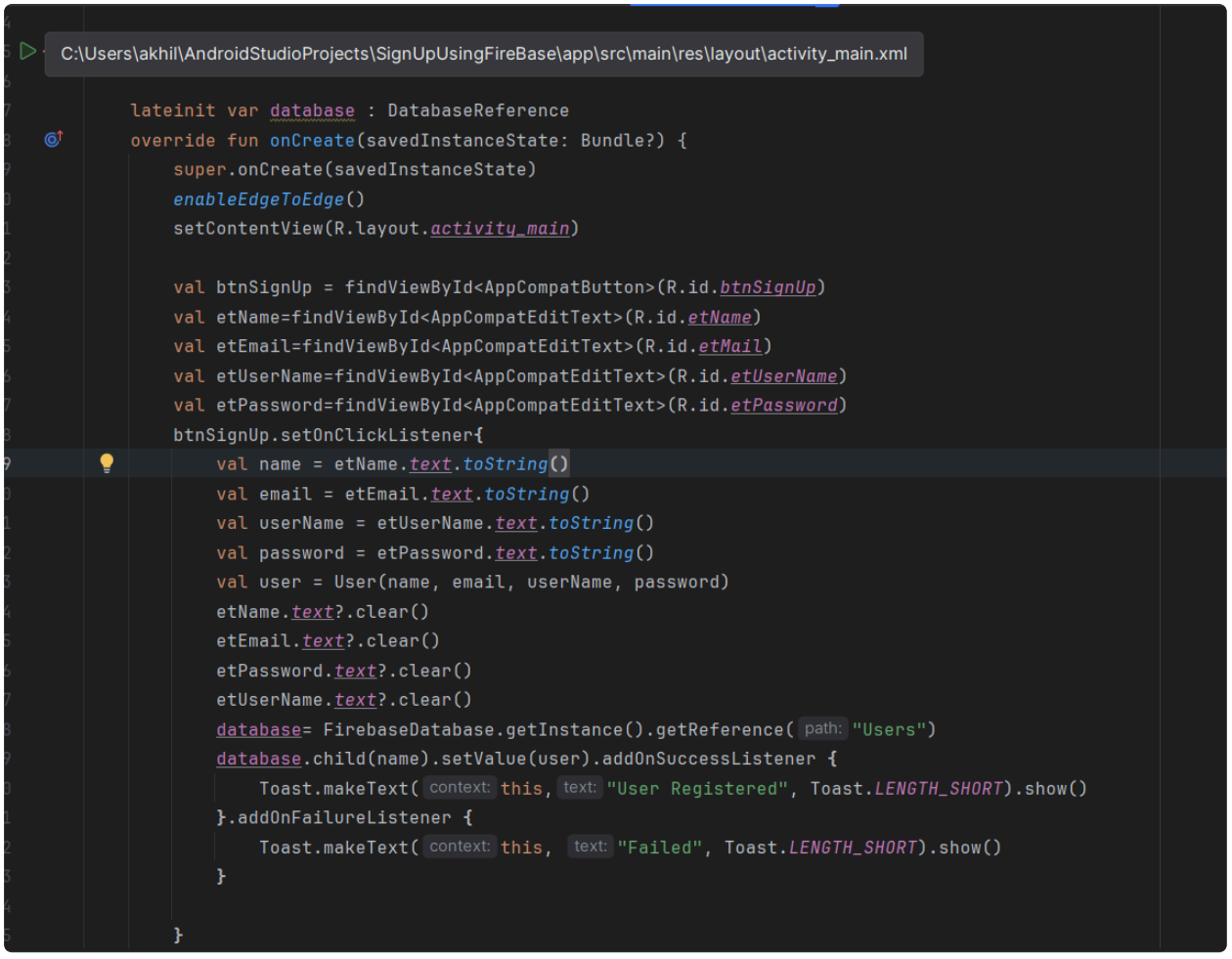
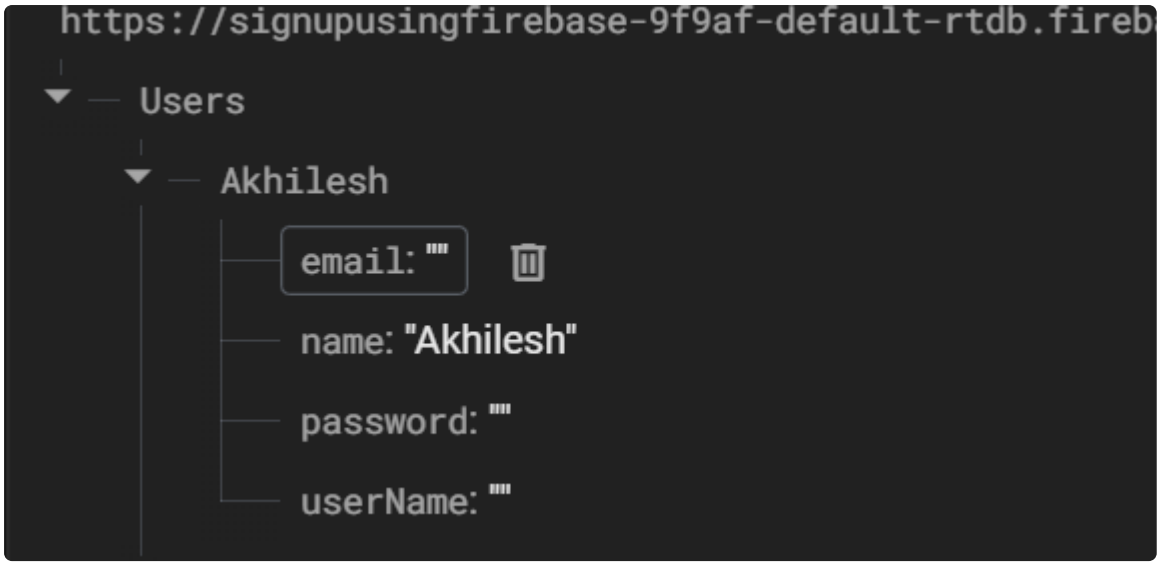


Image Format

Data in FireBase



Reading Data And Using Data From FireBase Data base

Read Data

```
databaseRefrence=FirebaseDatabase.getInstance().getReference( path: "Users")
 databaseRefrence.child(uniqueId).get().addOnSuccessListener {
```

same jaise waha set kra yha get krenge

```
class LoginActivity : AppCompatActivity() {
    companion object{
       const val KEY1 ="com.example.signupusingfirebase.LoginActivity.name"
        const val KEY2 ="com.example.signupusingfirebase.LoginActivity.mail"

        const val KEY3 ="com.example.signupusingfirebase.LoginActivity.userName"    }
    lateinit var databaseRefrence :DatabaseReference
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_login)
       val btnSignIn = findViewById<Button>(R.id.btnSignIn)
        val etSignInUserName = findViewById<AppCompatEditText>(R.id.etSignInUserName)
        val tvDirectSignIN = findViewById<TextView>(R.id.tvDirectSignIN)

        btnSignIn.setOnClickListener{

            // User tk ka address lena hai firebase me
            val uniqueId = etSignInUserName.text.toString()
            if(uniqueId.isNotEmpty()){
                readData(uniqueId)
            }else{
                Toast.makeText( context: this,  text: "Please Enter UserName", Toast.LENGTH_SHORT).show()

            }
        }
        tvDirectSignIN.setOnClickListener{
            val intent = Intent( packageContext: this, MainActivity::class.java)
            startActivity(intent)
        }
```

```
    fun readData(uniqueId: String) {

       databaseRefrence=FirebaseDatabase.getInstance().getReference( path: "Users")
        databaseRefrence.child(uniqueId).get().addOnSuccessListener {
            if(it.exists()){
                // user ko next screen pe bhej denge
                // it is itterator
                val email=it.child( path: "email").value
                val name = it.child( path: "name").value
                val password = it.child( path: "password").value
                val userName = it.child( path: "userName").value
                 val intent = Intent( packageContext: this, WelcomeActivity::class.java)
                intent.putExtra(KEY2,email.toString())
                intent.putExtra(KEY1,name.toString())
                intent.putExtra(KEY3,userName.toString())
                startActivity(intent)

            }else{
                Toast.makeText( context: this,  text: "User Doesn't Exist", Toast.LENGTH_SHORT).show()
            }
        }.addOnFailureListener {
            Toast.makeText( context: this,  text: "User Doesn't Exist, Please register", Toast.LENGTH_SHORT).show()
        }
```

data ko read kiya hai match kiya

Use data ke liye

```
val email=it.child( path: "email").value
val name = it.child( path: "name").value
val password = it.child( path: "password").value
val userName = it.child( path: "userName").value
```

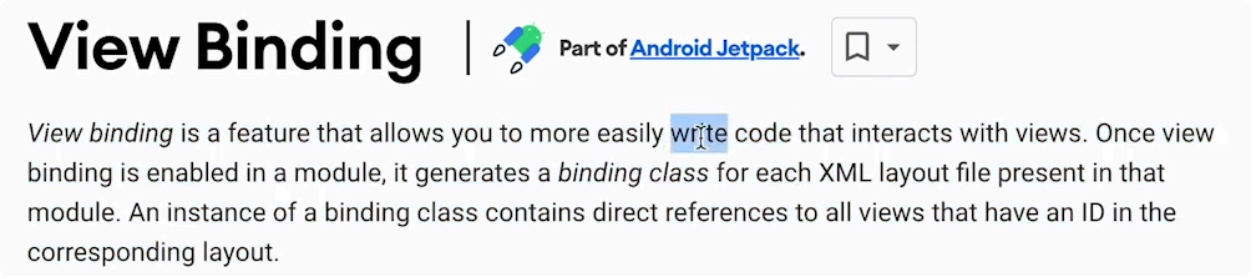
Yaha se data base se value nikal li

fir jha use krna tha waha bhej diya

```
val intent = Intent( packageContext: this, WelcomeActivity::class.java)
intent.putExtra(KEY2,email.toString())
intent.putExtra(KEY1,name.toString())
intent.putExtra(KEY3,userName.toString())
startActivity(intent)
```

View Binding

Find view by id baar baar likhna pdta hai isi ko overcome krne ke liye view binding ka use krte hai



Steps to use

Step 1 - Gradle build me add krna

```
buildFeatures {
    viewBinding = true
}
```

Step 2 -

```
class MainActivity : AppCompatActivity() {
    lateinit var binding: ActivityMainBinding //Step 1 Declare a binding variable
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding=ActivityMainBinding.inflate(layoutInflater)     //Step 2 Initialize the binding variable)
        enableEdgeToEdge()
        setContentView(binding.root) //Step 3 seeting binding root
        binding.tvDefault.setOnClickListener{ // step 4 bs binding ka use kr dive sare id ko le skte hai find view by id ka use ni hoga

        }
}
```

Dialogue Box

1-Alert Dialogue box

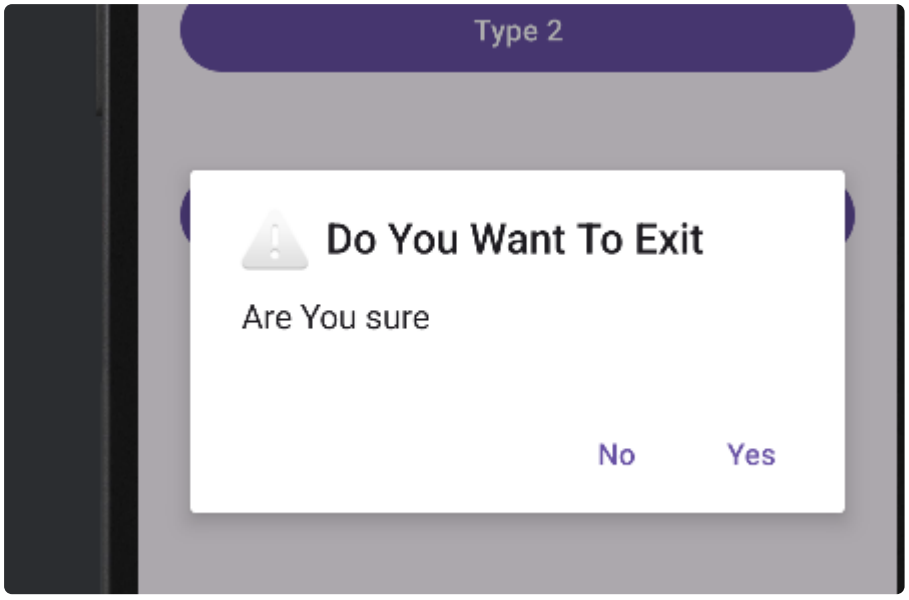
2-Date Picker Dialogue box

3-Time Picker Dialogue box

4-Progress Dialogue box

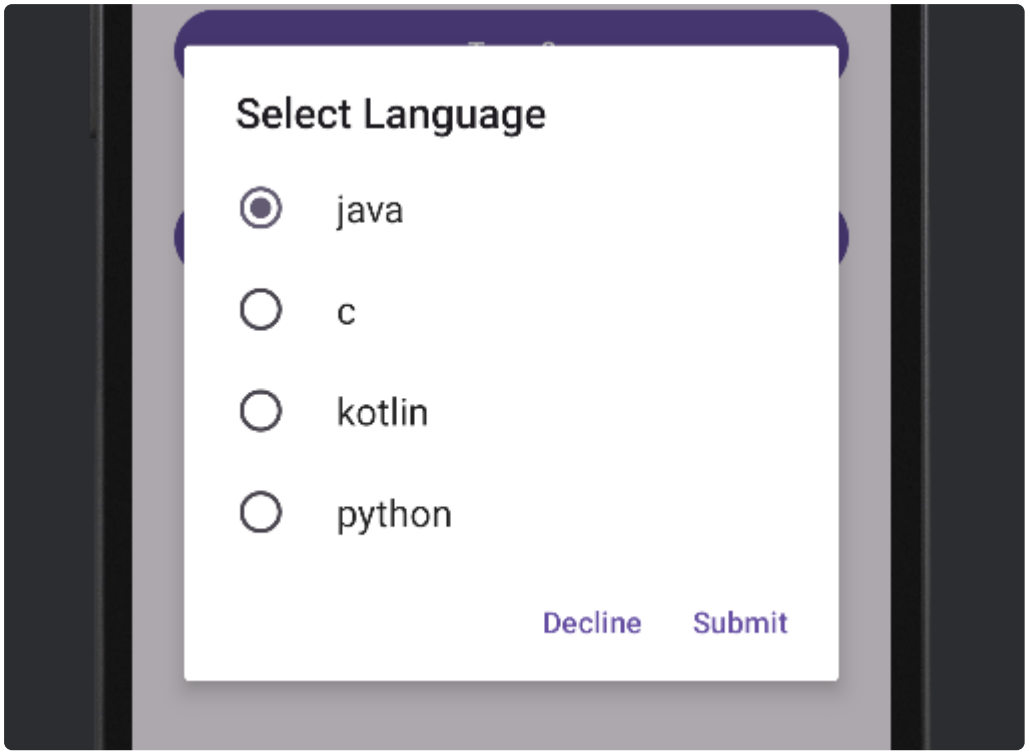
Alert dialogue box

```kotlin
// Simple dialogue box
binding.btn1.setOnClickListener {
    val builder1 = AlertDialog.Builder( context: this)
    builder1.setTitle("Do You Want To Exit")
    builder1.setMessage("Are You sure")
    builder1.setIcon(android.R.drawable.ic_dialog_alert)

    builder1.setPositiveButton( text: "Yes", { dialogInterface, which ->
        // Yes pe click kra to kya hona chahiye
        finish()
    })
    builder1.setNegativeButton( text: "No", { dialogInterface, which ->
        // No ke click kra to kya hona chahiye
    })
    builder1.show()
}
```



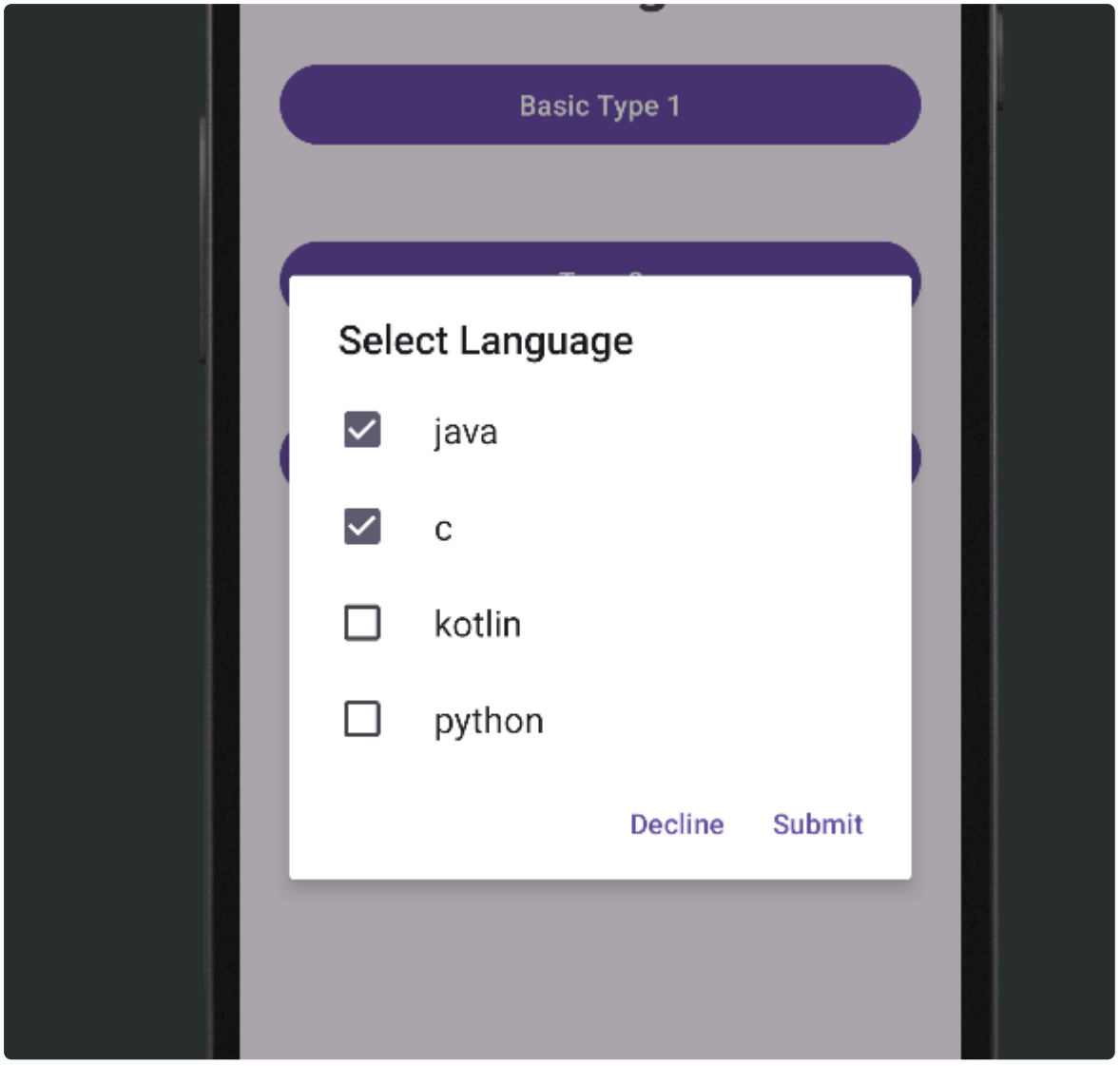
Alert box Option type

```kotlin
binding.btn2.setOnClickListener {
    val options = arrayOf("java", "c", "kotlin", "python")
    val builder2 = AlertDialog.Builder( context: this)
    builder2.setTitle("Select Language")
    builder2.setSingleChoiceItems(options,  checkedItem: 0, { dialogInterface, which ->
        // Kisi option pe click hone pe kya option hona chahiye
        Toast.makeText( context: this,  text: "Clicked on ${options[which]}", Toast.LENGTH_SHORT).show()
        // which is a itterator
    })
    builder2.setPositiveButton( text: "Submit", { dialogInterface, which ->
        // Yes pe click kra to kya hona chahiye
        finish()
    })
    builder2.setNegativeButton( text: "Decline", { dialogInterface, which ->
        // No ke click kra to kya hona chahiye
    })
    builder2.show()
}
```

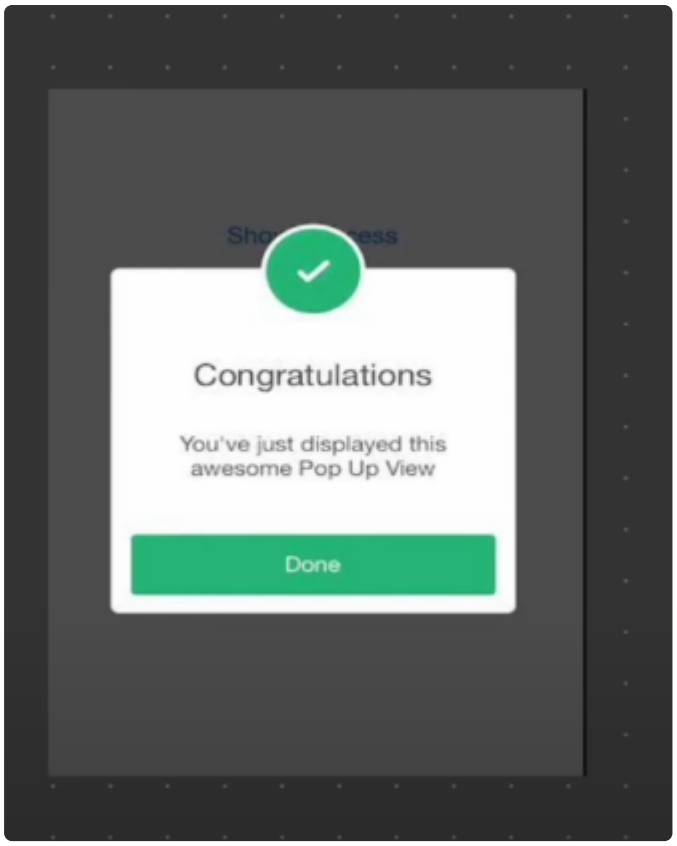


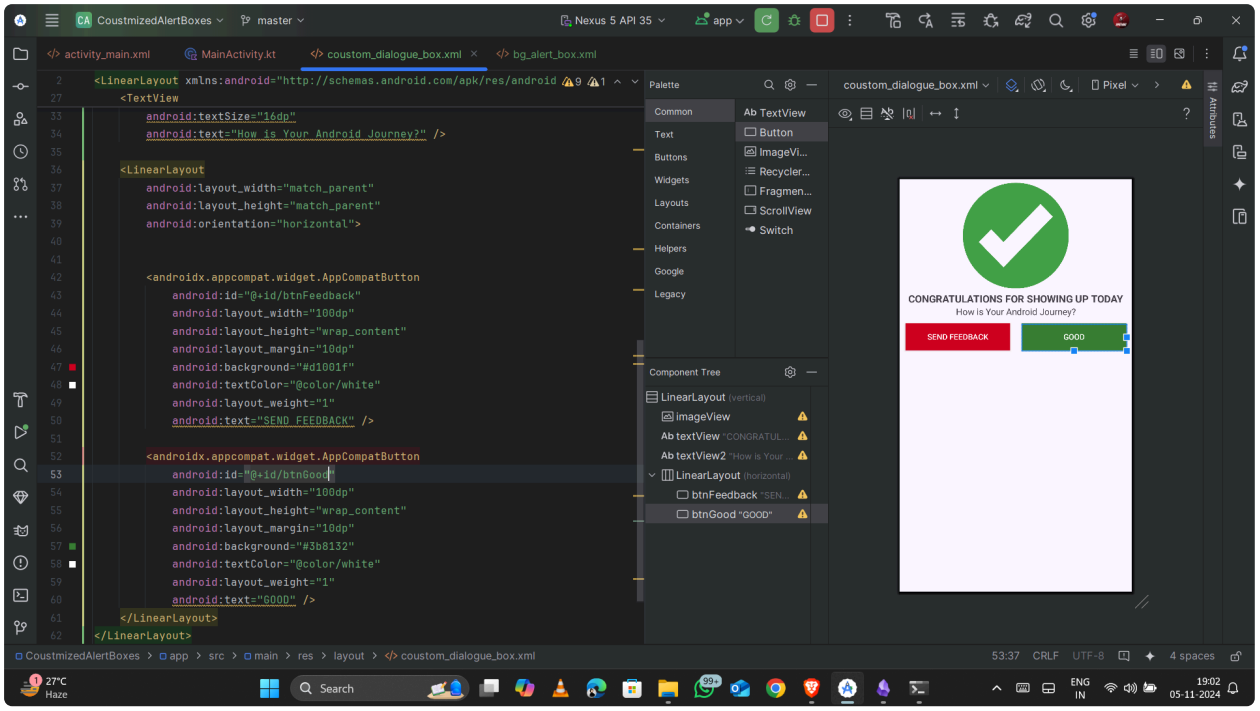
Alert Box Type 3 - MultiOption

```
binding.btn3.setOnClickListener {
    val options = arrayOf("java", "c", "kotlin", "python")
    val builder3 = AlertDialog.Builder( context: this)
    builder3.setTitle("Select Language")
    builder3.setTitle("Select Language")
    builder3.setMultiChoiceItems(options, checkedItems: null , { dialogInterface, which, isChecked ->

        Toast.makeText( context: this,  text: "Clicked on ${options[which]}", Toast.LENGTH_SHORT).show()
    })
    builder3.setPositiveButton( text: "Submit", { dialogInterface, which ->
        // Yes pe click kra to kya hona chahiye
        finish()
    })
    builder3.setNegativeButton( text: "Decline", { dialogInterface, which ->
        // No ke click kra to kya hona chahiye
    })
    builder3.show()

}
```



Coustomized Alert boxes





Step 1 custom xml crete kiya jaha alert box khud se bnaya

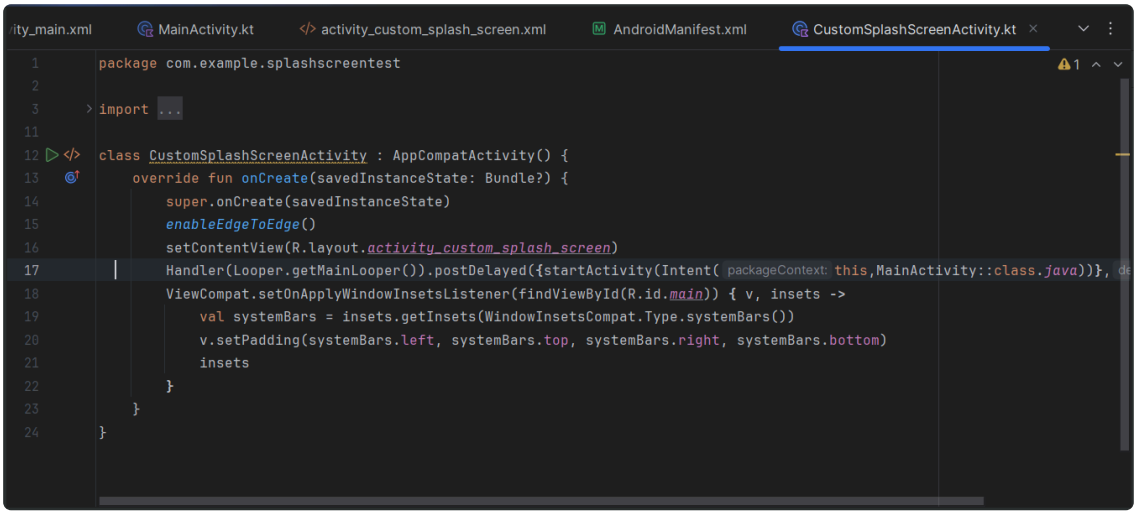
baki steps image me hai

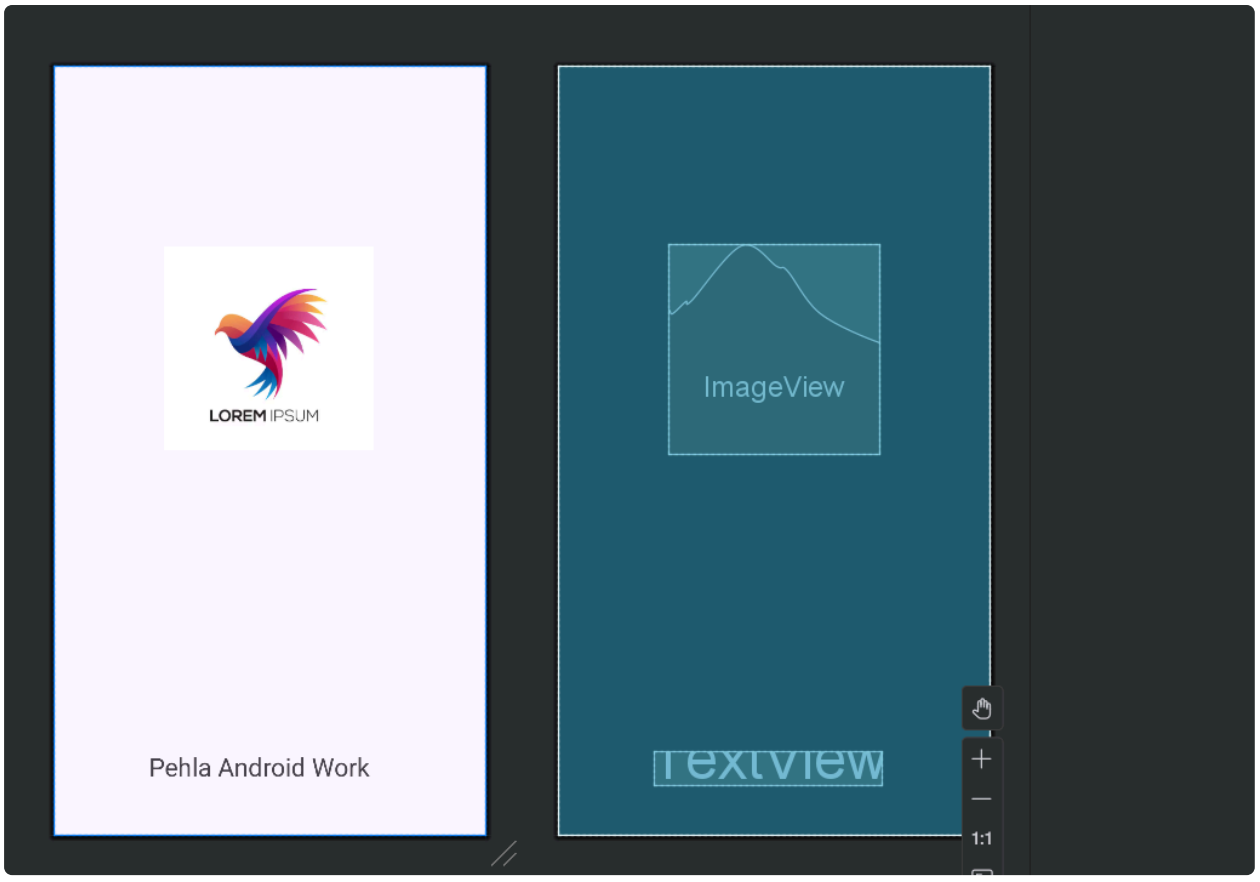
```kotlin
class MainActivity : AppCompatActivity() {
    // Create a variable of type Dialogue
    private lateinit var dialog: Dialog // Step 1 to crate a variable of type Dialogue
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
        dialog = Dialog( context: this) // Step 2 Intialize the dialog variable
        dialog.setContentView(R.layout.coustom_dialogue_box) // Step 3 varaible se xml link krenge

        var myBtn = findViewById<Button>(R.id.btnClickMe)
        // Creating variables from another activity
        var btnGood = dialog.findViewById<Button>(R.id.btnGood)
        var btnFeedback = dialog.findViewById<Button>(R.id.btnFeedback)
        // step 4 us varaible se kuch uthana hoga to aise uthayenge

        btnGood.setOnClickListener {
            dialog.dismiss()
        }
        btnFeedback.setOnClickListener {
            // Implicit intent
            val intent = Intent(Intent.ACTION_SENDTO)
            intent.data = android.net.Uri.parse( uriString: "mailto:akhileshbltr2002@gmail.com")
            startActivity(intent)
        }
        myBtn.setOnClickListener {
            dialog.show() // Step 5 Dialogue show krdiya
        }

    }
}
```

Projects

1. Splash screen

Backend





2. Splash screen when loading
   it is a dynamic splash screen mtlb ki agr background me kuch load ho rha to splash screen chl jaye

```
activity
<ImageView
    android:id="@+id/imageView2"
    android:layout_width="200dp"
    android:layout_height="200dp"
    android:layout_marginStart="16dp"
```

```
    android:layout_marginTop="212dp"

android:layout_marginBottom="211dp"

app:layout_constraintBottom_toBottomOf
="parent"

app:layout_constraintEnd_toEndOf="pare
nt"

app:layout_constraintHorizontal_bias="
0.456"

app:layout_constraintStart_toStartOf="
parent"

app:layout_constraintTop_toTopOf="pare
nt"

app:layout_constraintVertical_bias="0.
203"
    app:srcCompat="@drawable/img" />

<TextView
    android:id="@+id/textView2"
```

```
android:layout_width="wrap_content"

android:layout_height="wrap_content"
    android:layout_marginTop="200dp"
    android:layout_marginBottom="34dp"
    android:padding="10dp"
    android:text="Ye screen tb chlegi
jb kuch load hoga"
    android:textSize="25sp"
    android:textStyle="bold"

app:layout_constraintBottom_toBottomOf
="parent"

app:layout_constraintEnd_toEndOf="pare
nt"

app:layout_constraintStart_toStartOf="
parent"

app:layout_constraintTop_toBottomOf="@
+id/imageView2" />
```

```
bACKEND

import android.content.Intent
```

```
import android.os.AsyncTask
import android.os.Bundle
import
androidx.activity.enableEdgeToEdge
import
androidx.appcompat.app.AppCompatActivi
ty
import androidx.core.view.ViewCompat
import
androidx.core.view.WindowInsetsCompat

class SplashScreenLoading :
AppCompatActivity() {
    override fun
onCreate(savedInstanceState: Bundle?)
{

super.onCreate(savedInstanceState)
        enableEdgeToEdge()

setContentView(R.layout.activity_splas
h_screen_loading)
        tillLoading();

ViewCompat.setOnApplyWindowInsetsListe
ner(findViewById(R.id.main)) { v,
```

```
insets ->
            val systemBars =
insets.getInsets(WindowInsetsCompat.Ty
pe.systemBars())

v.setPadding(systemBars.left,
systemBars.top, systemBars.right,
systemBars.bottom)
            insets


        }
    }
    private fun tillLoading() {
        loadingOperation().execute();
    }
    private open inner class
loadingOperation :
AsyncTask<String?,Void?,String?>(){
        override fun
doInBackground(vararg params:
String?): String? {
            for(i in 0..6){
                try{
                    Thread.sleep(1000)
                }
```

```
                catch(e:Exception){

Thread.interrupted()

                }
            }
            return "Result"
        }

        override fun
onPostExecute(result: String?) {

super.onPostExecute(result)
            val intent =
Intent(this@SplashScreenLoading,MainAc
tivity::class.java);
            startActivity(intent)
        }

    }

}
```

```
class SplashScreenLoading : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_splash_screen_loading)
        tillLoading();
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->
            val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)
            insets


        }
    }
    private fun tillLoading() {
        loadingOperation().execute();
    }
    private open inner class loadingOperation : AsyncTask<String?,Void?,String?>(){
        override fun doInBackground(vararg params: String?): String? {...}

        override fun onPostExecute(result: String?) {
            super.onPostExecute(result)
            val intent = Intent( packageContext: this@SplashScreenLoading,MainActivity::class.java);
            startActivity(intent)
        }

    }

}
```