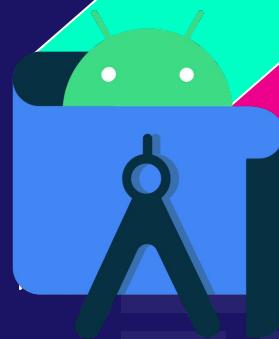


#12

Android Studio



Android Studio

"Jetpack Compose - the future of Android Studio?"

INNEHÅLLSFÖRTECKNING

01

Översikt

03

Modifiers,
Layouts & Coding

02

Why Compose?
First Project

04

Uppgifter
&
Övningar



01

ÖVERSIKT

Jetpack Compose



Compose

Compose löser följande problem

- Boilerplate kod
- Komplexa layouts
- State management (inbyggda viewModel metoder)
- Testande av UI komponenter är enklare





02

Why Compose?

Dåtid & Nutid



Release

2021 släpptes jetpack Compose 1.0 som 'Stable Release'



2021

Release

Tidigt stadie oavsett om det är 1.4 Stable.

Mycket som har ändrats och kommer förändras.

Design är något som många utvecklare måste lära om.

Idag Existerar 1.4

2023

Compose Features



XML Approach

```
<Button  
    android:id="@+id/my_button"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Click me" />
```

```
findViewById<Button>(R.id.my_button).setOnClickListener {  
    // Handle button click  
}
```

Mer kod, svårare att underhålla, förvirrande upplägg

Compose Approach

```
Button(  
    onClick = { /* Handle button click */ },  
    text = "Click me"  
)
```

Mindre veligt, mindre kod, bättre byggnation och deklaration.

Compose Features

- State Management
- Animationer
- Material Design (google guidelines design)
- Preview (coding preview)



Who wins?

Xml används fortfarande, det är inte underlägset.

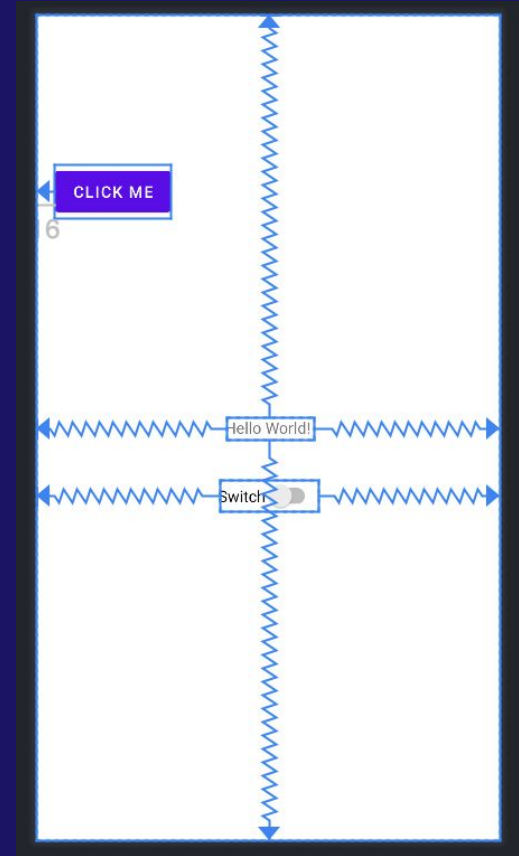
Företag och privatpersoner använder sig fortfarande av XML via standard Android programmering.

Compose

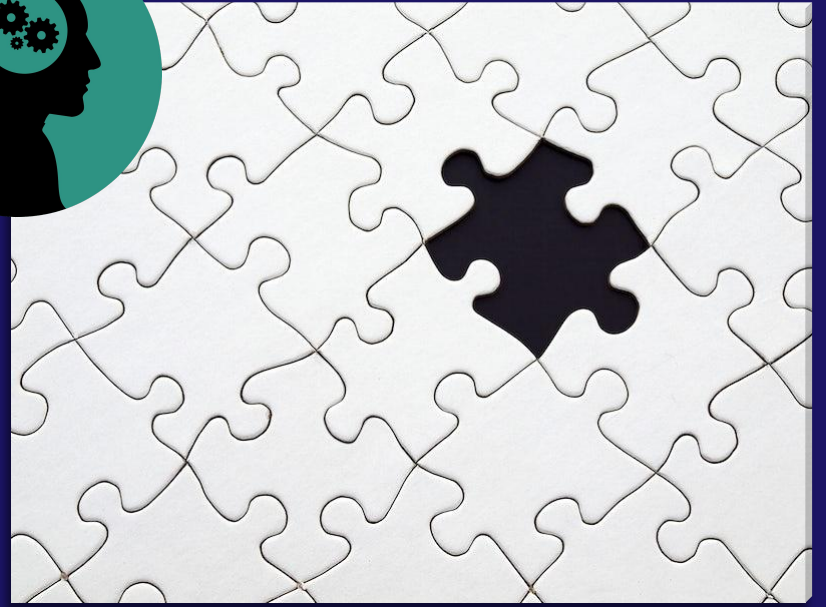
Är nytt och tillkommer med många nya features!

Detta är inte lika använt som XML i dagsläget, men det finns många pionjärer och 'early adopters' som kommer framkalla Compose som det 'framtida verktyget'.

Compose stödjer inte Drag and Drop



Frågor?



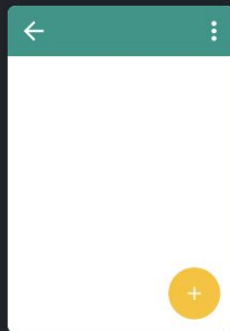
First Project



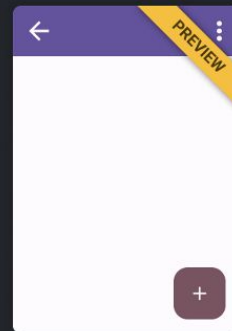
New Project



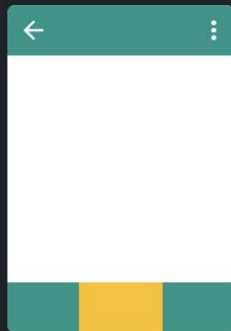
No Activity



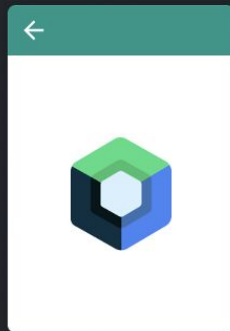
Basic Activity



Basic Activity (Material3)



Bottom Navigation Activity

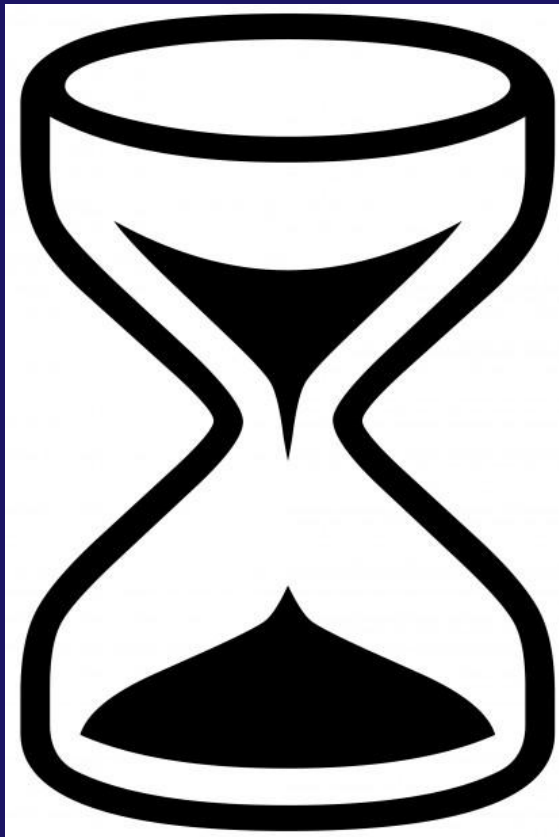


Empty Compose Activity



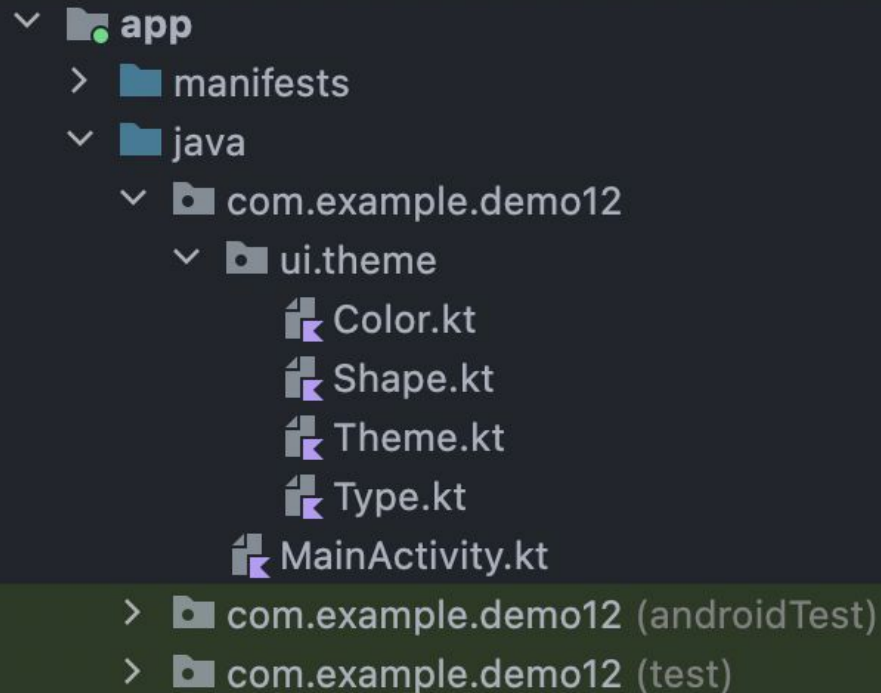
Empty Compose Activity (Material3)

First Time Setup



Kommer ta en stund för den att
ladda ner och hämta senaste!

Project Structure



Project Structure

Appens Färger

▼ **app**
 > manifests
 ▼ java
 ▼ com.example.demo12
 ▼ ui.theme

Color.kt

Shape.kt

Theme.kt

Type.kt

MainActivity.kt

> com.example.demo12 (androidTest)

> com.example.demo12 (test)

```
1  package com.example.demo12.ui.theme
2
3  import androidx.compose.ui.graphics.Color
4
5  val Purple200 = Color(color: 0xFFBB86FC)
6  val Purple500 = Color(color: 0xFF6200EE)
7  val Purple700 = Color(color: 0xFF3700B3)
8  val Teal200 = Color(color: 0xFF03DAC5)
```

Project Structure

Former

▼ app	1	<code>package com.example.demo12.ui.theme</code>
> manifests	2	
▼ java	3	<code>import ...</code>
▼ com.example.demo12	6	
▼ ui.theme	7	<code>val Shapes = Shapes(</code>
Color.kt	8	<code> small = RoundedCornerShape(4.dp),</code>
Shape.kt	9	<code> medium = RoundedCornerShape(4.dp),</code>
Theme.kt	10	<code> large = RoundedCornerShape(0.dp)</code>
Type.kt	11	<code>)</code>
MainActivity.kt		
> com.example.demo12 (androidTest)		
> com.example.demo12 (test)		
▼ res		
> drawable		
> minman		

Project Structure

Ljus / Mörk tema

- ▼ app
 - > manifests
 - ▼ java
 - ▼ com.example.demo12
 - ▼ ui.theme
 - Color.kt
 - Shape.kt
 - Theme.kt
 - Type.kt
 - MainActivity.kt
 - > com.example.demo12 (androidTest)
 - > com.example.demo12 (test)
 - ▼ res
 - > drawable
 - > mipmap
 - > values
 - > xml

```
1 package com.example.demo12.ui.theme
2
3 import ...
4
5
6
7
8
9 private val DarkColorPalette = darkColors(
10     primary = Purple200,
11     primaryVariant = Purple700,
12     secondary = Teal200
13 )
14
15 private val LightColorPalette = lightColors(
16     primary = Purple500,
17     primaryVariant = Purple700,
18     secondary = Teal200
19 )
```

Project Structure

Typsnitt AKA Fonts

▼ app

> manifests

▼ java

▼ com.example.demo12

▼ ui.theme

Color.kt

Shape.kt

Theme.kt

Type.kt

MainActivity.kt

> com.example.demo12 (androidTest)

> com.example.demo12 (test)

▼ res

▼ drawable

1

2

3

8

9

10

11

12

13

14

15

```
package com.example.demo12.ui.theme

import ...

// Set of Material typography styles to start with
val Typography = Typography(
    body1 = TextStyle(
        fontFamily = FontFamily.Default,
        fontWeight = FontWeight.Normal,
        fontSize = 16.sp
    )
)
```


Main

```
class MainActivity : ComponentActivity () {  
  
    override fun onCreate (savedInstanceState : Bundle?) {  
        super.onCreate (savedInstanceState)  
  
        setContent {  
            Demo12Theme {  
                // A surface container using the 'background' color from the theme  
                Surface (  
                    modifier = Modifier.fillMaxSize (),  
                    color = MaterialTheme.colors.background  
                ) {  
                    Greeting ("Android")  
                }  
            }  
        }  
    }  
}
```

Main

```
setContent {  
    Demo12Theme {  
  
        Surface(  
            modifier = Modifier.fillMaxSize(),  
            color = MaterialTheme.colors.background  
        ) {  
            Greeting("Android")  
        }  
    }  
}
```



Denna färg

Main

```
@Composable  
fun Greeting(name: String) {  
    Text(text = "Hello $name!")  
}
```

Main

```
@Composable
fun Greeting(name: String) {
    Text(text = "Hello $name!")
}
```

`@Composable` är byggstenarna inom Compose. Dessa ska främst fokusera på byggnation av UI. Composable funktioner/lambda kan också innehålla enkel funktionalitet och bearbeta data.

Best practice är att försöka hålla dessa till UI och låta dem vara så koncisa och enkla som möjligt.

Main

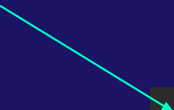
```
@Preview(showBackground = true)
@Composable
fun DefaultPreview() {
    Demo12Theme {
        Greeting("Android")
    }
}
```

Main

```
@Preview(showBackground = true)
@Composable
fun DefaultPreview() {
    Demo12Theme {
        Greeting("Android")
    }
}
```

Vår funktion från tidigare!

Main

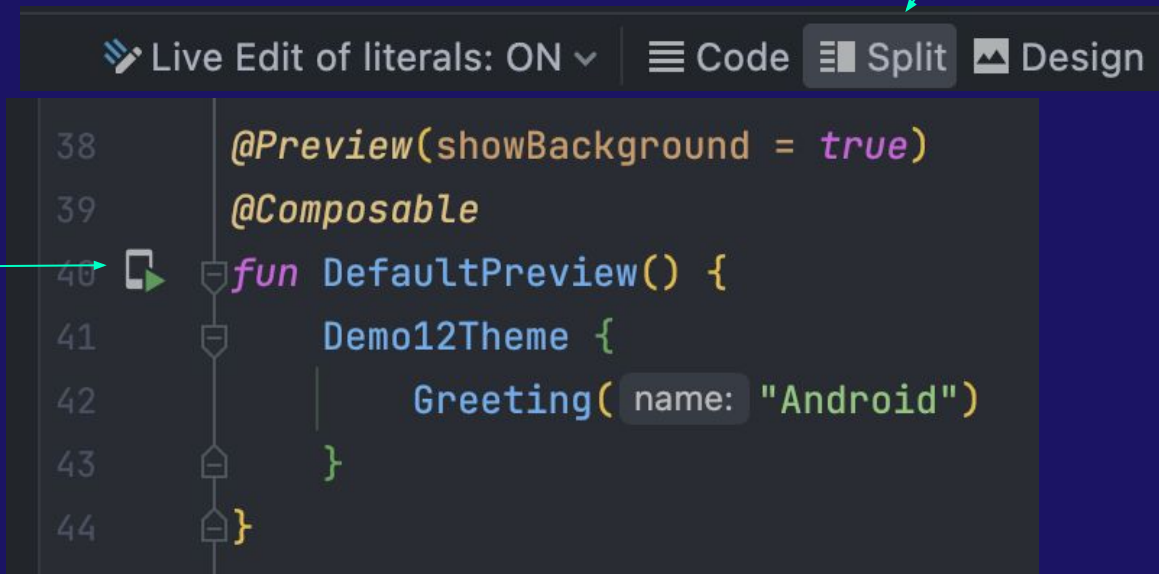


```
@Preview(showBackground = true)
@Composable
fun DefaultPreview() {
    Demo12Theme {
        Greeting("Android")
    }
}
```

Hur ska vi kunna se design om vi inte kan köra drag'n'drop?

`@Preview()` tillåter oss att se detta utan att starta appen!

Preview



```
38     @Preview(showBackground = true)
39     @Composable
40     fun DefaultPreview() {
41         Demo12Theme {
42             Greeting(name: "Android")
43         }
44     }
```

Preview om ni vill visa upp design!

Preview

```
29  @Composable
30  fun Greeting(name: String) {
31      Text(text = "Hello $name!")
32  }
33
34  @Preview(showBackground = true)
35  @Composable
36  fun DefaultPreview() {
37      Demo12Theme {
38          Greeting(name = "Android")
39      }
40  }
```

DefaultPreview

Hello Android!

Koda och få resultat: LIVE

Shortcut!

com

 comp

Create @Composable function

Press ^. to choose the selected (or first) suggestion and insert a dot afterwards


Next Tip



37 @Composable

38  fun  () {

39 ~~~~~

40  }

41

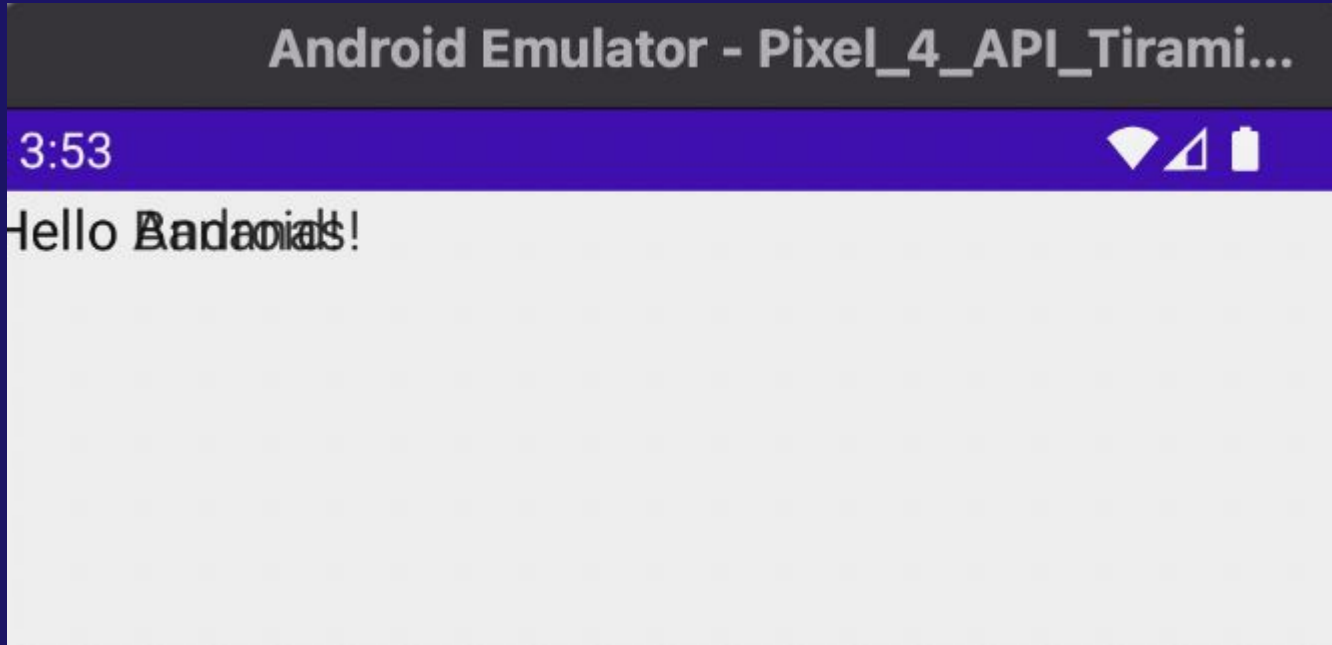
Shortcut!

Två Composable Komponenter skapar problem!

```
34  @Preview(showBackground = true)
35  @Composable
36  fun DefaultPreview() {
37      Demo12Theme {
38          Greeting(name: "Android")
39          Greeting(name: "Bananas")
40      }
41  }
```



Problems...





03

Modifiers, Layouts & Coding

UI Layouts



Image

```
Image(  
    painter = painterResource(id = R.drawable.ic_launcher_background),  
    contentDescription = "Template",  
    contentScale = ContentScale.Crop,  
)
```

painter = Image Location

contentDescription = Beskrivning av bilden

contentScale = Bildförhållande AKA *Aspect Ratio*

More here: <https://developer.android.com/jetpack/compose/graphics/images/customize>

Layouts



Column



Row



Box

Column

```
35     @Preview(showBackground = true)
36     @Composable
37     fun DefaultPreview() {
38         Demo12Theme {
39
40             Column() { this: ColumnScope
41                 Greeting( name: "Android")
42                 Greeting( name: "Bananas")
43             }
44
45         }
46     }
```

DefaultPreview

Hello Android!
Hello Bananas!

Row



```
Row() { this: RowScope  
    Greeting(name = "Android ")  
    Greeting(name = "Android")  
}
```

DefaultPreview



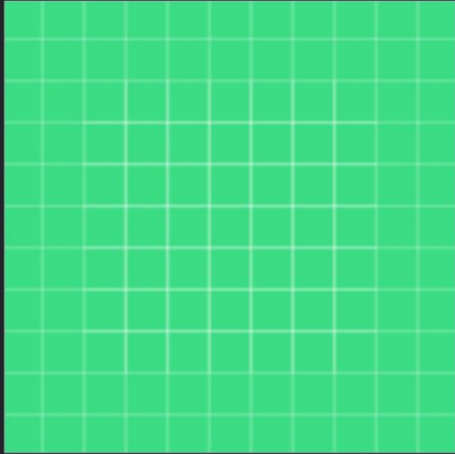
Hello Android !Hello Android!

Box

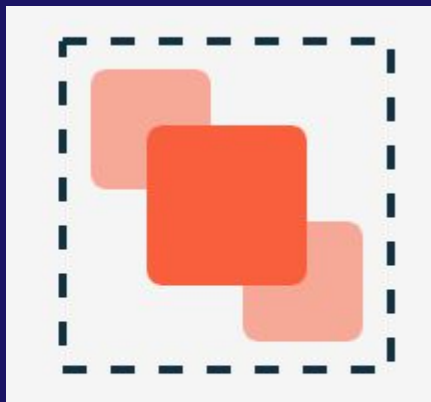
Vart tog texten vägen?

```
46 Box() { this: BoxScope
47   Greeting(name = "Android")
48   Image(
49     painter = painterResource(id = R.drawable.ic_launcher_background),
50     contentDescription = "Template",
51     contentScale = ContentScale.Crop,
52   )
53 }
54
55
56
57
58 }
59 }
60
61
62
```

DefaultPreview



Box



```
Box() { this: BoxScope  
  Image(  
    painter = painterResource(id = R.drawable.ic_launcher_background),  
    contentDescription = "Template",  
    contentScale = ContentScale.Crop,  
  )  
  Greeting(name = "Android")  
}
```

DefaultPreview



Hello Android!

EXAMPLES

```
@Composable
fun ArtistCard(artist: Artist) {
    Row(verticalAlignment = Alignment.CenterVertically) {
        Image(/*...*/)
        Column {
            Text(artist.name)
            Text(artist.lastSeenOnline)
        }
    }
}
```



Alfred Sisley

3 minutes ago

EXAMPLES

```
@Composable
fun ArtistAvatar(artist: Artist) {
    Box {
        Image(/*...*/)
        Icon(/*...*/)
    }
}
```



Modifiers



Modifier.



background(color: Color, shape: Shape = ...) for ...	Modifier
background(brush: Brush, shape: Shape = ..., alph...	Modifier
fillMaxSize(fraction: Float = ...) for Modifier i...	Modifier
size(size: Dp) for Modifier in androidx.compose.f...	Modifier
clip(shape: Shape) for Modifier in androidx.compo...	Modifier
size(size: DpSize) for Modifier in androidx.compo...	Modifier
size(width: Dp, height: Dp) for Modifier in andro...	Modifier
height(height: Dp) for Modifier in androidx.compo...	Modifier
height(intrinsicSize: IntrinsicSize) for Modifier	Modifier
fillMaxWidth(fraction: Float = ...) for Modifier ...	Modifier
absoluteOffset(x: Dp = ..., y: Dp = ...) for Modi...	Modifier

Modifier.

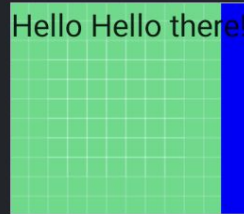
```
Box(  
  Modifier  
    .background(Color.Gray)  
    .size(200.dp)) { this: BoxScope
```

EXAMPLES

```
48 Box(Modifier.background(Color.Blue)) { this: BoxScope
49
50     Image(
51         painter = painterResource(id = R.drawable.ic_launcher_background),
52         contentDescription = "Template",
53         contentScale = ContentScale.Crop
54     )
55
56     Column{ this: ColumnScope
57         Greeting( name: "Hello there")
58     }
59 }
```

DefaultPre...  

Hello Hello there!



EXAMPLES

```
50 Column() { this: ColumnScope
51     Box(
52         Modifier
53             .background(Color.Gray)
54             .size(200.dp)) { this: BoxScope
55
56         Box(Modifier.background(color = Color.Magenta)) { this: BoxScope
57
58             Image(
59                 painter = painterResource(id = R.drawable.ic_launcher_foreground),
60                 contentDescription = "")
61
62             Column() { this: ColumnScope
63
64                 Greeting( name: "lorem lorem lorem lorem lore mlorem")
65             }
66         }
67     }
68 }
69 }
```

DefaultPreview

Hello lorem lorem lorem
lorem lore mlorem!



EXAMPLES

```
@Preview
@Composable
fun Test2() {
    Column(
        modifier = Modifier.fillMaxSize()
            .background(Color.White),
        verticalArrangement = Arrangement.Center,
        horizontalAlignment = Alignment.CenterHorizontally
    ) {
        Text(
            text = "First item",
            modifier = Modifier.padding(16.dp)
        )
        Text(
            text = "Second item",
            modifier = Modifier.padding(16.dp)
        )
        Text(
            text = "Third item",
            modifier = Modifier.padding(16.dp)
        )
    }
}
```

First item

Second item

Third item

ViewModel?



ViewModel?

Remember Approach

<https://developer.android.com/jetpack/compose/state>

ViewModel Approach

<https://farhan-tanvir.medium.com/stateflow-with-jetpack-compose-7d9c9711c2>

86

Navigation?



Navigation...?

<https://developer.android.com/jetpack/compose/navigation>



04

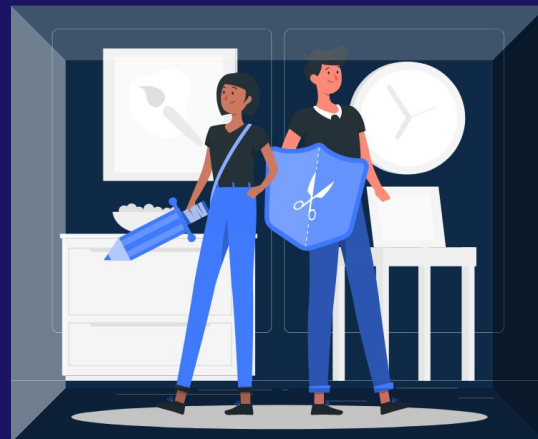
Uppgifter
&
Eget Arbete

Välkommen till första uppgiften!

Uppgifterna är till för att testa dina färdigheter och kunskaper för att både öva och repetera på det vi har arbetat med under föreläsningarna.

Dessa är **INTE** obligatoriska.
Men är starkt rekommenderat att arbeta med.

Uppgifter



MINNS DU?

```
// Vad är Jetpack Compose?
```

```
// När det kommer till 'state handling'  
så rekommenderar de att vi istället  
använder oss utav 'rememberBySaveable'  
för att bevara state istället för  
ViewModel.
```

Varför är detta ett problem?

```
1          // -Uppgift #1- //
2
3      /* INSTRUCTIONS
4
5          Skapa ett nytt projekt!
6
7          Döp projektet till: Lektion_12_uppgifter
8
9
10         -Skapa ett Jetpack Compose projekt-
11
12     */
13
14
15
16
17
18
19
20
21
22
23
```

Uppgift #1

Kom igång enkelt med uppgift #1

```

1      // -Uppgift #2- //
2
3      /* INSTRUCTIONS
4
5          Skapa en 'composable' function.
6          Skriv 'comp'
7
8          Skapa en 'button' med en 'onClick' som visar
9          upp en 'snackbar'
10
11      */
12
13      // HINT & Examples
14      hint("
15
16      @Composable
17      fun DisplaySnack(
18          message: String) {
19          Button(onClick = { println(message) }) {
20              Text(message)
21          }
22      }
23
24      ")

```

Uppgift #2

Övning ger färdighet, börja enkelt med variabler!

```
1          // -Uppgift #3- //
2
3  /* INSTRUCTIONS
4
5      Med dina kunskaper, försök efterlikna bilden
6
7      Glöm inte att tänka på hur saker ligger!
8
9
10
11
12  HINT - Glöm inte...
13      + Column
14      + Box
15      + Row
16
17  */
18
19
20
21
22
23
```

Uppgift #3



Kelvin

*Dedicated Android
Developer, I am now
open for work!*

THANKS!

Do you have any questions?
kristoffer.johansson@sti.se

sti.learning.nu/

CREDITS: This presentation template was created by
Slidesgo, including icons by Flaticon, and
infographics & images by Freepik.

You can also contact me VIA Teams (quicker response)
Du kan också kontakta mig VIA Teams (Snabbare svar)