


Logcat & ConstraintLayout

**Mobile App Programming
Fall, 2024**



Today's Contents

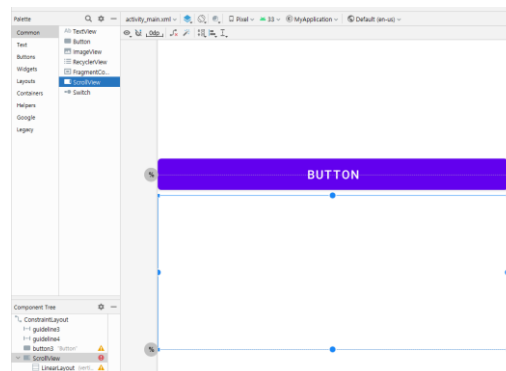
- Review of Previous Lecture
- Logcat
- ConstraintLayout & Guideline
- Lab practice



Review

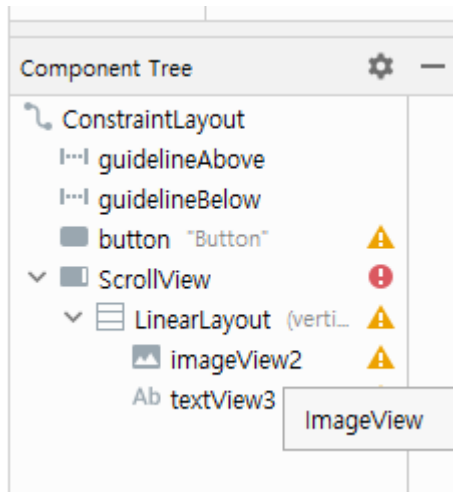
ScrollView

- You remember about (Horizontal) ScrollView
 - It's a scrollable View
 - **Only one component can be in ScrollView**
 - Let's try to make **vertically scrollable** one in this time.
- Put the ScrollView – Fill it below button

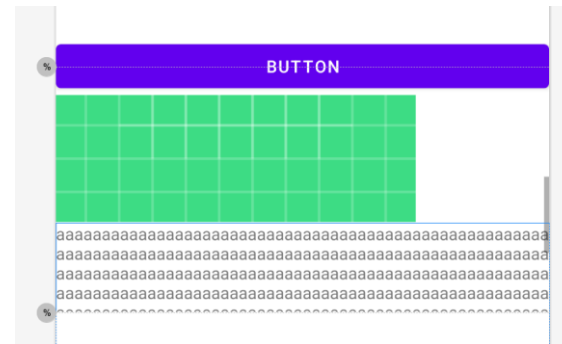


ScrollView

- Put ImageView and TextView inside the ScrollView
 - Directly put two views is impossible
 - Puts LinearLayout first and then insert them.

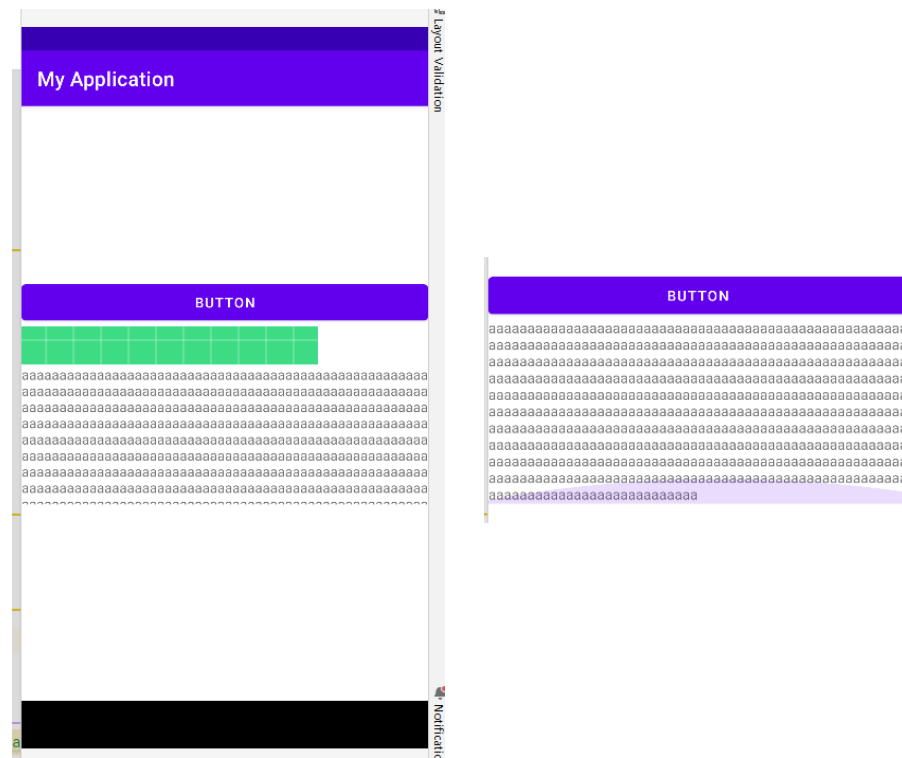


- Set ImageView height/width to 300dp
- Set TextView text to something long



ScrollView

- You can scroll vertically in one part of the screen

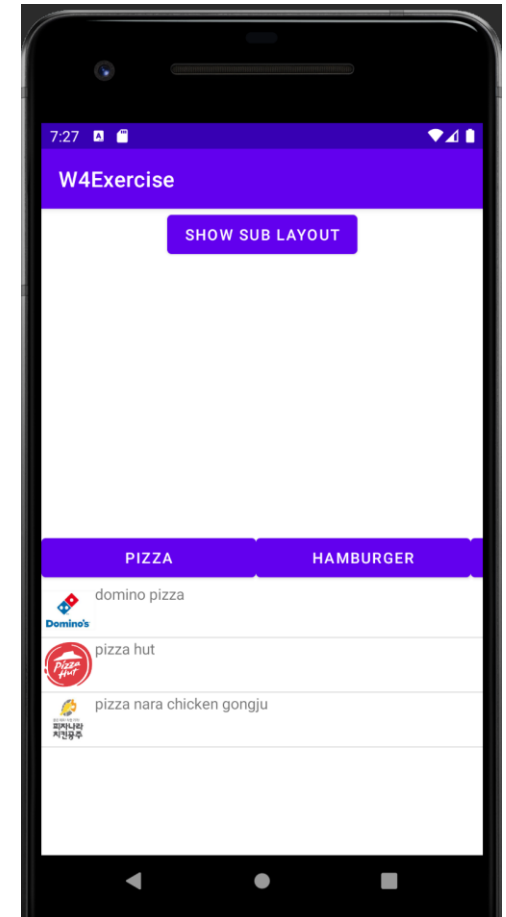


ListView

- Think about you are making ListView manually
 - You need to iterate with the number of items
 - and construct the view by objectifying each element in each item
- With Inflater, you can make use of layout XML instead of making and adjusting all element
- With Adapter, you don't need to take care of iteration
 - Just `listView.adapter = myAdapterInstance` do all!
 - It is calling setter, `listView.setAdapter(...)`

ListView

- Think about the previous lecture...
- ListView data has two type,
 - Restaurant name
 - Restaurant image ID
- To store all types of data in one object, we've made the **class** Restaurant
 - Which have String and Int

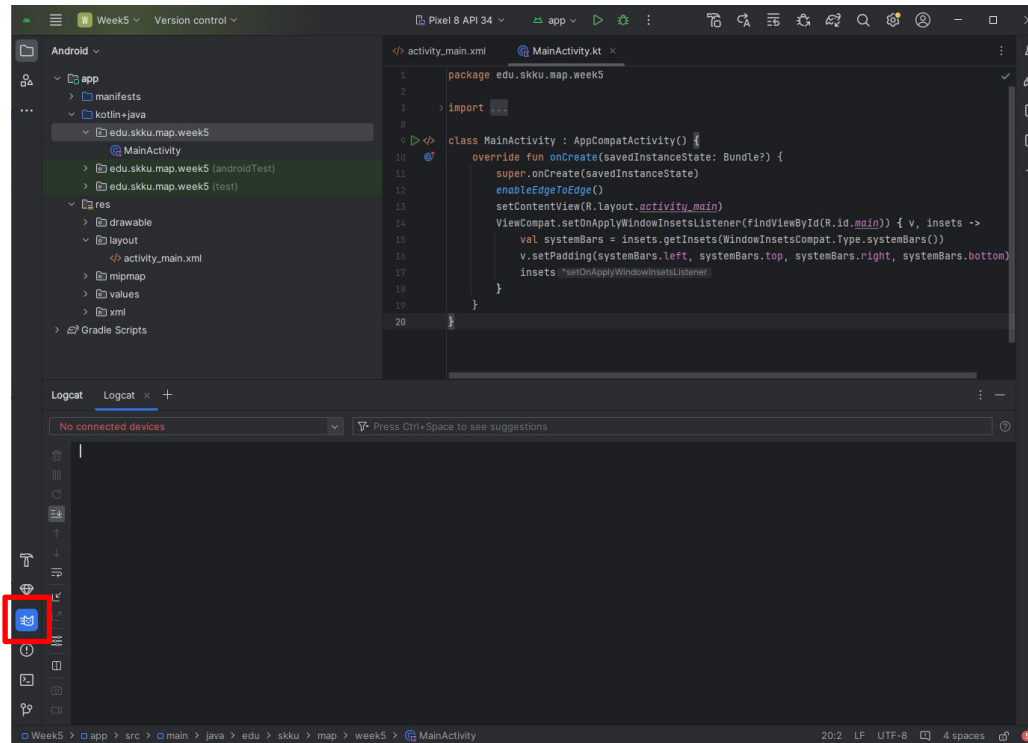




Logcat

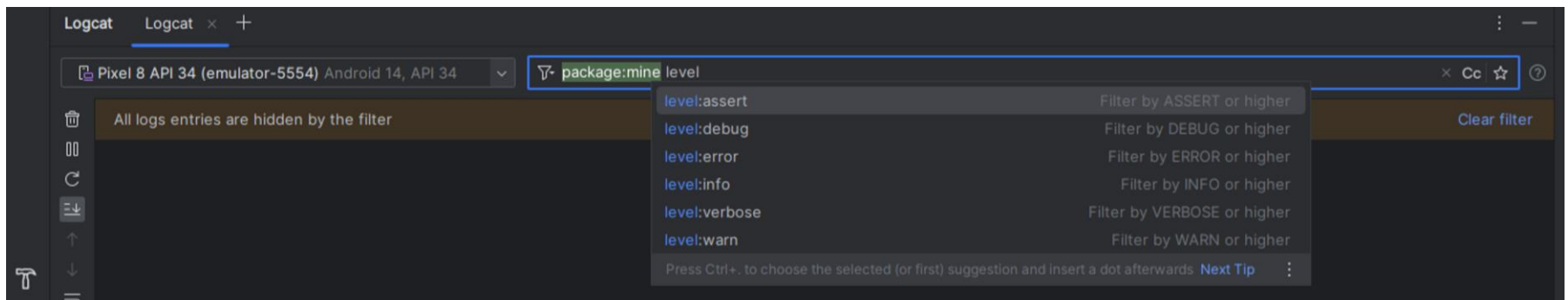
Logcat

- Logcat
 - Bottom
 - See what happened to your android device



Logcat

- Logcat
 - All logs are shown here.
 - You can filter logs by searching keyword, or debug level
 - Verbose < Debug < Info < Warn < Error < Assert
 - Selecting debug level: show logs with same or higher levels
 - by typing **level:<level>** in search window



Logcat

The screenshot shows the Logcat window in Android Studio. The top bar indicates the device is 'Pixel 2 API 30 (emulator-5554) Android 11, API 30' and the log level is set to 'package:mime level:debug'. The log entries show the application's startup sequence, including loading system libraries and initializing the Android runtime. A fatal exception occurs, with the message 'FATAL EXCEPTION: main' and 'Process: edu.skku.cs.myapplication, PID: 7342'. The exception is a 'java.lang.RuntimeException: Unable to start activity ComponentInfo{edu.skku.cs.myapplication/edu.skku.cs.myapplication.MainActivity}: java.lang.ClassCastException: com.google.android.material.textview.MaterialTextView cannot be cast to edu.skku.cs.myapplication.MainActivity.onCreate(MainActivity.kt:11)'. The stack trace shows the exception was thrown from the MainActivity.onCreate method at line 11, which called performLaunchActivity. The bottom of the window shows the exception was caused by a ClassCastException where a MaterialTextView was cast to MainActivity.

```
Logcat: Logcat x +
Pixel 2 API 30 (emulator-5554) Android 11, API 30 package:mime level:debug
2023-03-26 20:19:13.027 7342-7342 NetworkSecurityConfig edu.skku.cs.myapplication D No Network Security Config specified, using platform default
2023-03-26 20:19:13.028 7342-7342 NetworkSecurityConfig edu.skku.cs.myapplication D No Network Security Config specified, using platform default
2023-03-26 20:19:13.075 7342-7367 libEGL edu.skku.cs.myapplication D loaded /vendor/lib/egl/libEGL_emulation.so
2023-03-26 20:19:13.080 7342-7367 libEGL edu.skku.cs.myapplication D loaded /vendor/lib/egl/libGLESv1_CM_emulation.so
2023-03-26 20:19:13.082 7342-7367 libEGL edu.skku.cs.myapplication D loaded /vendor/lib/egl/libGLESv2_emulation.so
2023-03-26 20:19:13.087 7342-7342 AppCompatDelegate edu.skku.cs.myapplication D Checking for metadata for AppLocalesMetadataHolderService : Service not found
2023-03-26 20:19:13.140 7342-7342 s.myapplication edu.skku.cs.myapplication W Accessing hidden method Landroid/view/View; ->computeFitSystemWindows(Landroid/graphics/Rect;Landroid/graphics,
2023-03-26 20:19:13.140 7342-7342 s.myapplication edu.skku.cs.myapplication W Accessing hidden method Landroid/view/ViewGroup; ->makeOptionalFitsSystemWindows()V (greylist, reflection, all
2023-03-26 20:19:13.155 7342-7342 AndroidRuntime edu.skku.cs.myapplication D Shutting down VM
2023-03-26 20:19:13.158 7342-7342 AndroidRuntime edu.skku.cs.myapplication E FATAL EXCEPTION: main
Process: edu.skku.cs.myapplication, PID: 7342
java.lang.RuntimeException: Unable to start activity ComponentInfo{edu.skku.cs.myapplication/edu.skku.cs.myapplication.MainActivity}: java.lang.ClassCastException: com.google.android.material.textview.MaterialTextView cannot be cast to edu.skku.cs.myapplication.MainActivity.onCreate(MainActivity.kt:11)
    at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:3449)
    at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:3601)
    at android.app.servertransaction.LaunchActivityItem.execute(LaunchActivityItem.java:85)
    at android.app.servertransaction.TransactionExecutor.executeCallbacks(TransactionExecutor.java:135)
    at android.app.servertransaction.TransactionExecutor.execute(TransactionExecutor.java:95)
    at android.app.ActivityThread$H.handleMessage(ActivityThread.java:2066)
    at android.os.Handler.dispatchMessage(Handler.java:106)
    at android.os.Looper.loop(Looper.java:223)
    at android.app.ActivityThread.main(ActivityThread.java:7656) <1 internal line>
    at com.android.internal.os.RuntimeInit$MethodAndArgsCaller.run(RuntimeInit.java:592)
    at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:947)
Caused by: java.lang.ClassCastException: com.google.android.material.textview.MaterialTextView cannot be cast to edu.skku.cs.myapplication.MainActivity.onCreate(MainActivity.kt:11)
    at android.app.Activity.performCreate(Activity.java:8000)
    at android.app.Activity.performCreate(Activity.java:7984)
    at android.app.Instrumentation.callActivityOnCreate(Instrumentation.java:1309)
    at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:3422) <11 more...>
Sending signal SIGKILL to PID 7342 etc. 0
```

- You can also check error (or exception) message.
 - Even if you did not manually print logs.

Logcat

```
E FATAL EXCEPTION: main
Process: edu.skku.cs.myapplication, PID: 7342
java.lang.RuntimeException: Unable to start activity ComponentInfo{edu.skku.cs.myapplication/edu.skku.cs.myapplication.MainActivity}: java.lang.ClassCastException: c
    at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:3449)
    at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:3601)
    at android.app.servertransaction.LaunchActivityItem.execute(LaunchActivityItem.java:85)
    at android.app.servertransaction.TransactionExecutor.executeCallbacks(TransactionExecutor.java:135)
    at android.app.servertransaction.TransactionExecutor.execute(TransactionExecutor.java:95)
    at android.app.ActivityThread$H.handleMessage(ActivityThread.java:2066)
    at android.os.Handler.dispatchMessage(Handler.java:106)
    at android.os.Looper.loop(Looper.java:223)
    at android.app.ActivityThread.main(ActivityThread.java:7656) <1 internal line>
    at com.android.internal.os.RuntimeInit$MethodAndArgsCaller.run(RuntimeInit.java:592)
    at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:947)
Caused by: java.lang.ClassCastException: com.google.android.material.textview.MaterialTextView cannot be cast to android.widget.Button
    at edu.skku.cs.myapplication.MainActivity.onCreate(MainActivity.kt:11)
    at android.app.Activity.performCreate(Activity.java:8000)
    at android.app.Activity.performCreate(Activity.java:7984)
    at android.app.Instrumentation.callActivityOnCreate(Instrumentation.java:1309)
    at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:3422) <11 more...>
```

- First, find the **blue text**
 - It shows you which part of your code is in error.
 - Gray text is android API code (not recommend to go that deep)
 - Here, MainActivity.kt:11

Logcat

```
E FATAL EXCEPTION: main
Process: edu.skku.cs.myapplication, PID: 7342
java.lang.RuntimeException: Unable to start activity ComponentInfo{edu.skku.cs.myapplication/edu.skku.cs.myapplication.MainActivity}: java.lang.ClassCastException: c
    at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:3449)
    at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:3601)
    at android.app.servertransaction.LaunchActivityItem.execute(LaunchActivityItem.java:85)
    at android.app.servertransaction.TransactionExecutor.executeCallbacks(TransactionExecutor.java:135)
    at android.app.servertransaction.TransactionExecutor.execute(TransactionExecutor.java:95)
    at android.app.ActivityThread$H.handleMessage(ActivityThread.java:2066)
    at android.os.Handler.dispatchMessage(Handler.java:106)
    at android.os.Looper.loop(Looper.java:223)
    at android.app.ActivityThread.main(ActivityThread.java:7656) <1 internal line>
    at com.android.internal.os.RuntimeInit$MethodAndArgsCaller.run(RuntimeInit.java:592)
    at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:947)
Caused by: java.lang.ClassCastException: com.google.android.material.textview.MaterialTextView cannot be cast to android.widget.Button
    at edu.skku.cs.myapplication.MainActivity.onCreate(MainActivity.kt:11)
    at android.app.Activity.performCreate(Activity.java:8000)
    at android.app.Activity.performCreate(Activity.java:7984)
    at android.app.Instrumentation.callActivityOnCreate(Instrumentation.java:1309)
    at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:3422) <11 more...>
```

- Then, check **what Exception is occurred.**
 - Above that blue text, find nearest **Caused by:**
 - There, it'll explain why the error occurred
 - Here, java.lang.ClassCastException
 - and its description says **TextView cannot be cast to Button**

Logcat

```
FATAL EXCEPTION: main
Process: edu.skku.cs.myapplication, PID: 7342
java.lang.RuntimeException: Unable to start activity ComponentInfo{edu.skku.cs.myapplication/edu.skku.cs.myapplication.MainActivity}: java.lang.ClassCastException: com.google.android.material.textview.MaterialTextView cannot be cast to edu.skku.cs.myapplication.MainActivity.onCreate(MainActivity.kt:11)
    at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2910)
    at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:3041)
    at android.app.servertransaction.LaunchActivityItem.execute(LaunchActivityItem.java:81)
    at android.app.servertransaction.TransactionExecutor.executeLifecycleTransaction(TransactionExecutor.java:176)
    at android.app.servertransaction.TransactionExecutor.execute(TransactionExecutor.java:95)
    at android.app.ActivityThread$H.handleMessage(ActivityThread.java:1797)
    at android.os.Handler.dispatchMessage(Handler.java:106)
    at android.os.Looper.loop(Looper.java:223)
    at android.app.ActivityThread.main(ActivityThread.java:753)
    at com.android.internal.os.RuntimeInit$MethodAndArgsCaller.run(RuntimeInit.java:522)
    at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:985)
Caused by: java.lang.ClassCastException: com.google.android.material.textview.MaterialTextView cannot be cast to edu.skku.cs.myapplication.MainActivity.onCreate(MainActivity.kt:11)
    at android.app.Activity.performCreate(Activity.java:8000)
    at android.app.Activity.performCreate(Activity.java:7984)
    at android.app.Instrumentation.callActivityOnCreate(Instrumentation.java:1309)
    at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:3422) <11 more...>
```

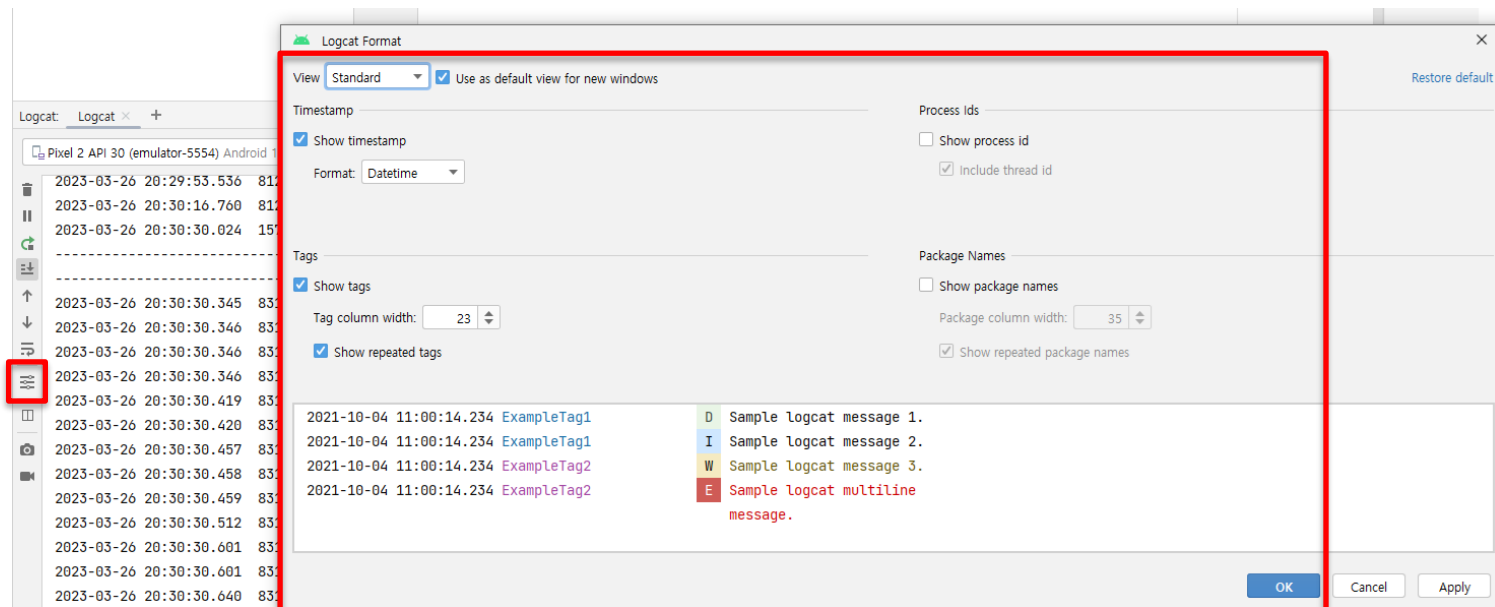
```
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        val btn1 = findViewById<Button>(R.id.tv) as Button
    }
}
```

```
<TextView
    android:id="@+id/tv"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Hello World!"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

- Finally, check your code (not only .kt but also .xml if it is related to UI)
 - Click that blue text to move to your code
 - With the exception and following information, try to find why errors are occurred
 - Here, you just typed wrong code: **cast TextView to Button**
 - If you still can't fix error, try manual logging for details.

Logcat

- You can modify the logcat format



Logcat

- You can manually log
 - **Log.v/d/i/w/e("tag string", "message string")**
 - Each alphabet represents: verbose/debug/info/warn/error
 - **Log.?(localClassName, "debug message")**
 - Automatically set tag to its class name

```
Log.i( tag: "This is tag", msg: "This is message")  
Log.w(localClassName, msg: "Easier tagging with 'localClassName'")
```

```
2023-03-26 20:30:30.640 I/MyApp:12345  
2023-03-26 20:30:30.640 I This is tag  
2023-03-26 20:30:30.640 MainActivity  
2023-03-26 20:30:30.700 W/HostConnection
```

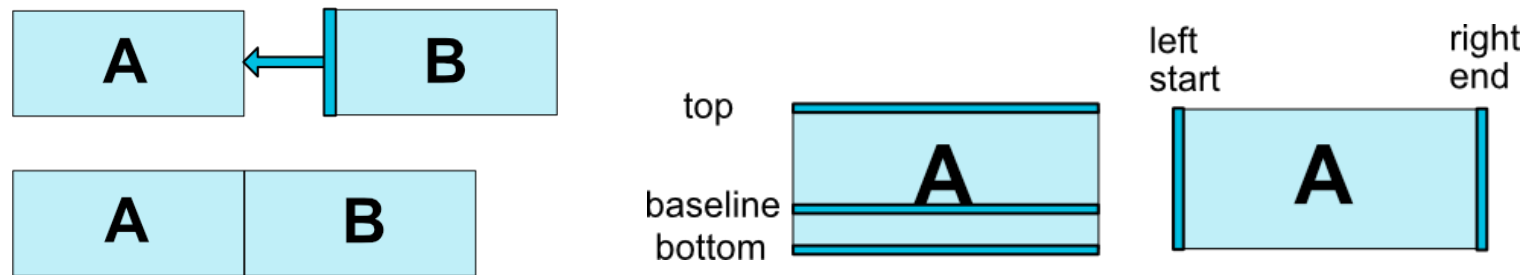
```
W necessary hidden method found 0x0/0x0/  
I This is message  
W Easier tagging with 'localClassName'  
D HostConnection: get() New Host Connect
```



ConstraintLayout

ConstraintLayout

- **ConstraintLayout** is a ViewGroup which allows you to position and size widgets in a flexible way

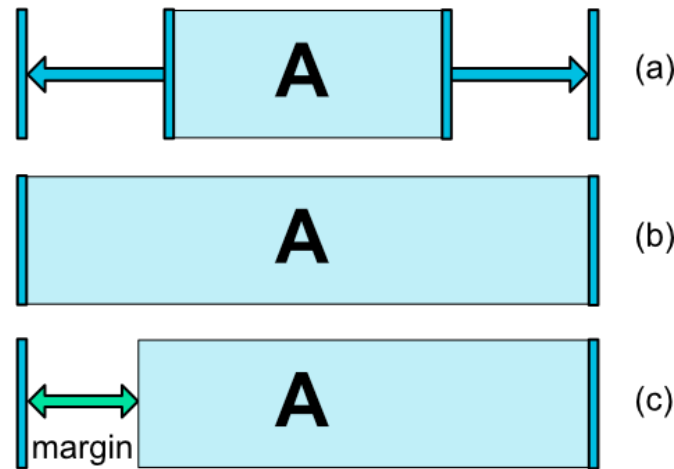


- Constraints allow you to position a given widget relative to another
- You can constrain a widget on the horizontal and vertical axis

<https://developer.android.com/reference/androidx/constraintlayout/widget/ConstraintLayout>

ConstraintLayout

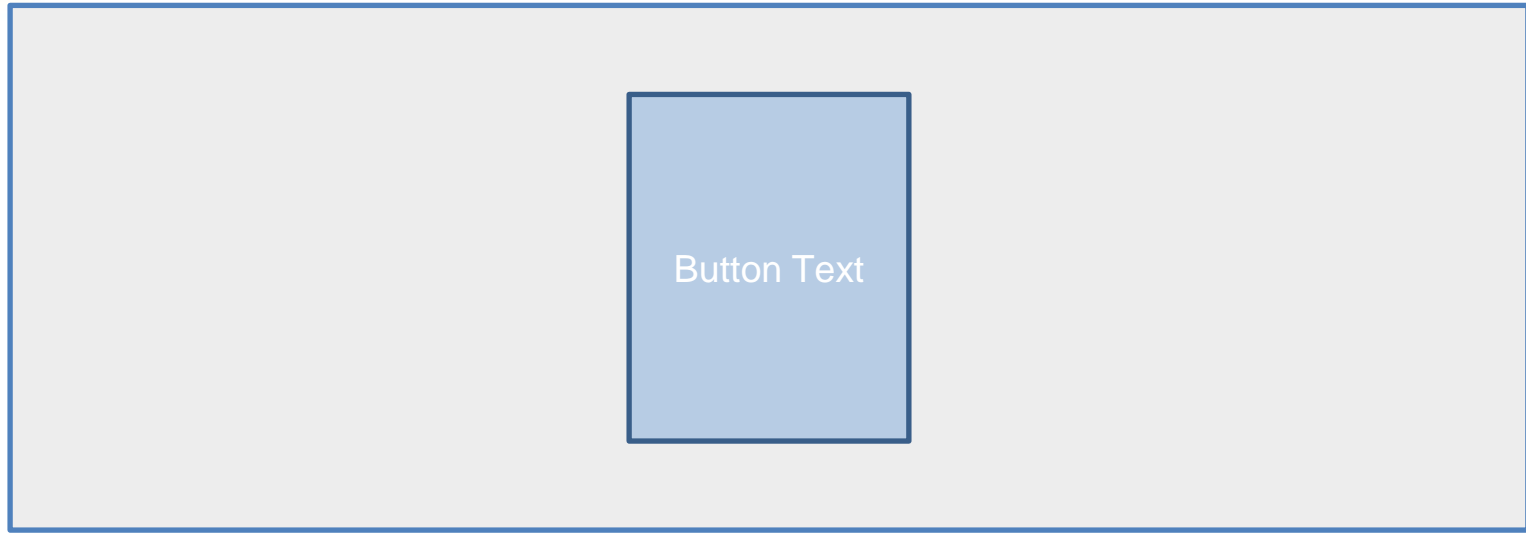
- If some widgets have horizontal constraints, you can define the width in 3 cases:



- (a): width = "**wrap_content**"
- (b): width = "**0dp**" (or "**match_constraint**")
- (c): width = "**0dp**" & has start margin

<https://developer.android.com/reference/androidx/constraintlayout/widget/ConstraintLayout>

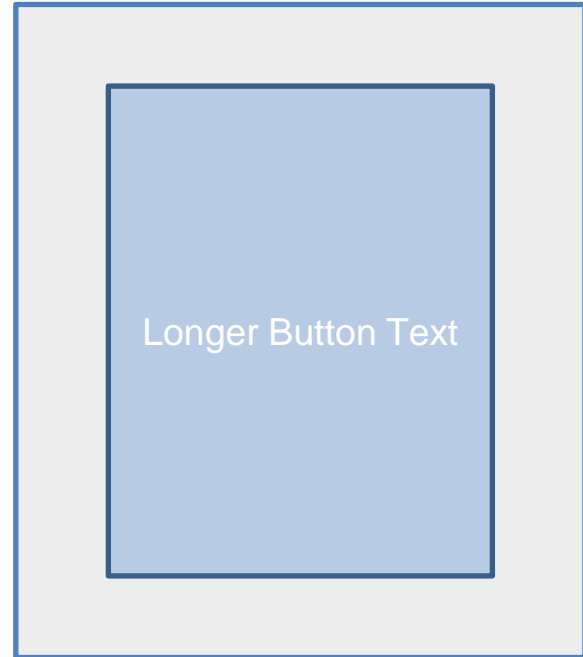
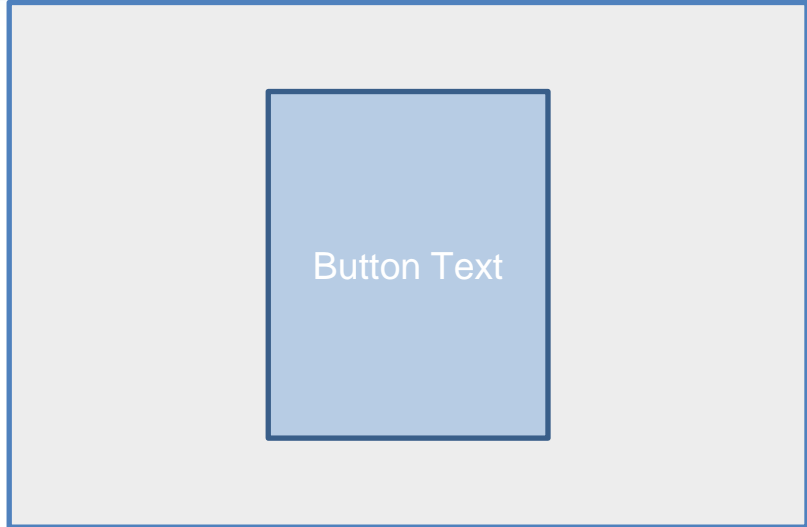
ConstraintLayout



- You want to make button that:
- Fill vertically with 50dp margin(both up and down)
- Width is flexible, according to the length of button text
- Center alignment in horizontal direction

<https://developer.android.com/reference/androidx/constraintlayout/widget/ConstraintLayout>

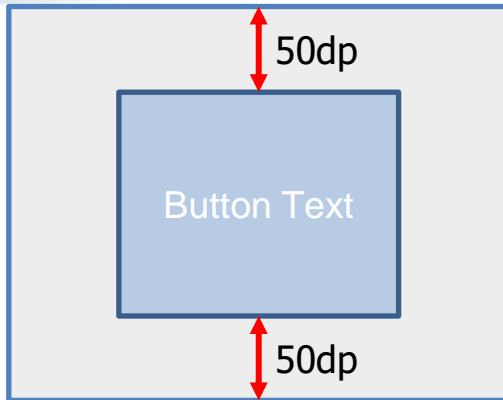
ConstraintLayout



- You want to make button that:
- Fill vertically with 50dp margin(both up and down)
- Width is flexible, according to the length of button text
- Center alignment in horizontal direction

<https://developer.android.com/reference/androidx/constraintlayout/widget/ConstraintLayout>

ConstraintLayout



width: WRAP_CONTENT

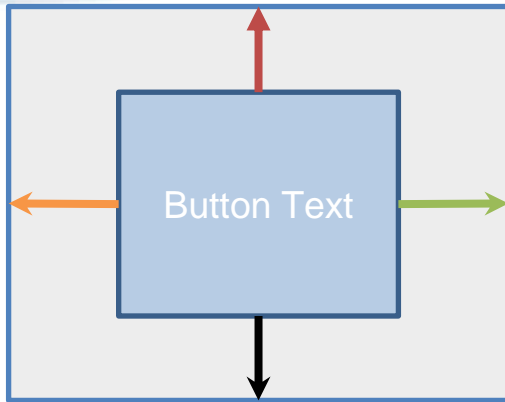
height: 0dp (=MATCH_CONSTRAINT)

marginVertical: 50dp

- You want to make button that:
- Fill vertically with 50dp margin(both up and down)
- Width is flexible, according to the length of button text
- Center alignment in horizontal direction

<https://developer.android.com/reference/androidx/constraintlayout/widget/ConstraintLayout>

ConstraintLayout

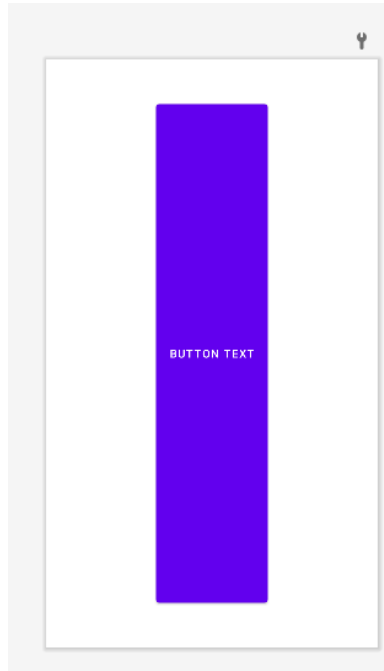


↑ `constraintTop_toTopOf: parent`
↓ `constraintBottom_toBottomOf: parent`
← `constraintStart_toStartOf: parent`
→ `constraintEnd_toEndOf: parent`

- You want to make button that:
- Fill vertically with 50dp margin(both up and down)
- Width is flexible, according to the length of button text
- Center alignment in horizontal direction

<https://developer.android.com/reference/androidx/constraintlayout/widget/ConstraintLayout>

ConstraintLayout

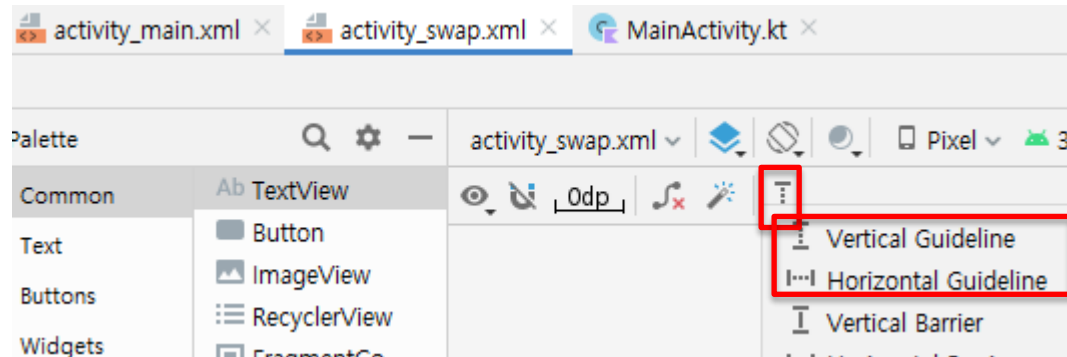


```
<Button
    android:id="@+id/button_flex"
    android:layout_width="wrap_content"
    android:layout_height="0dp"
    android:text="Button Text"
    android:layout_marginVertical="50dp"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
/>
```

- You want to make button that:
- Fill vertically with 50dp margin(both up and down)
- Width is flexible, according to the length of button text
- Center alignment in horizontal direction

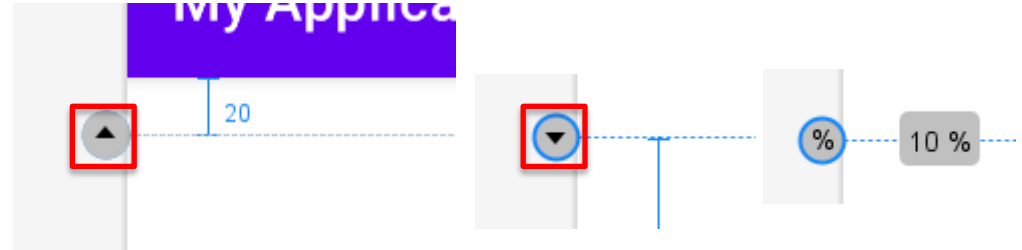
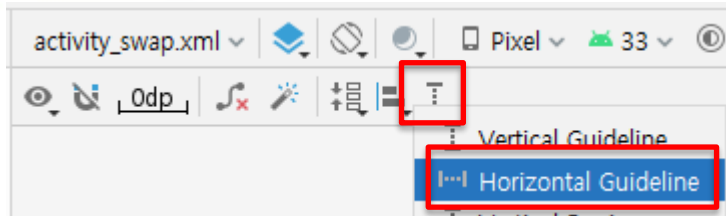
<https://developer.android.com/reference/androidx/constraintlayout/widget/ConstraintLayout>

Guideline



- You can use guidelines to set constraints properly
 - Guideline can be **X dp** from left/right/up/down
 - or **X %** horizontally/vertically
 - You can **divide the screen into sections**

Guideline

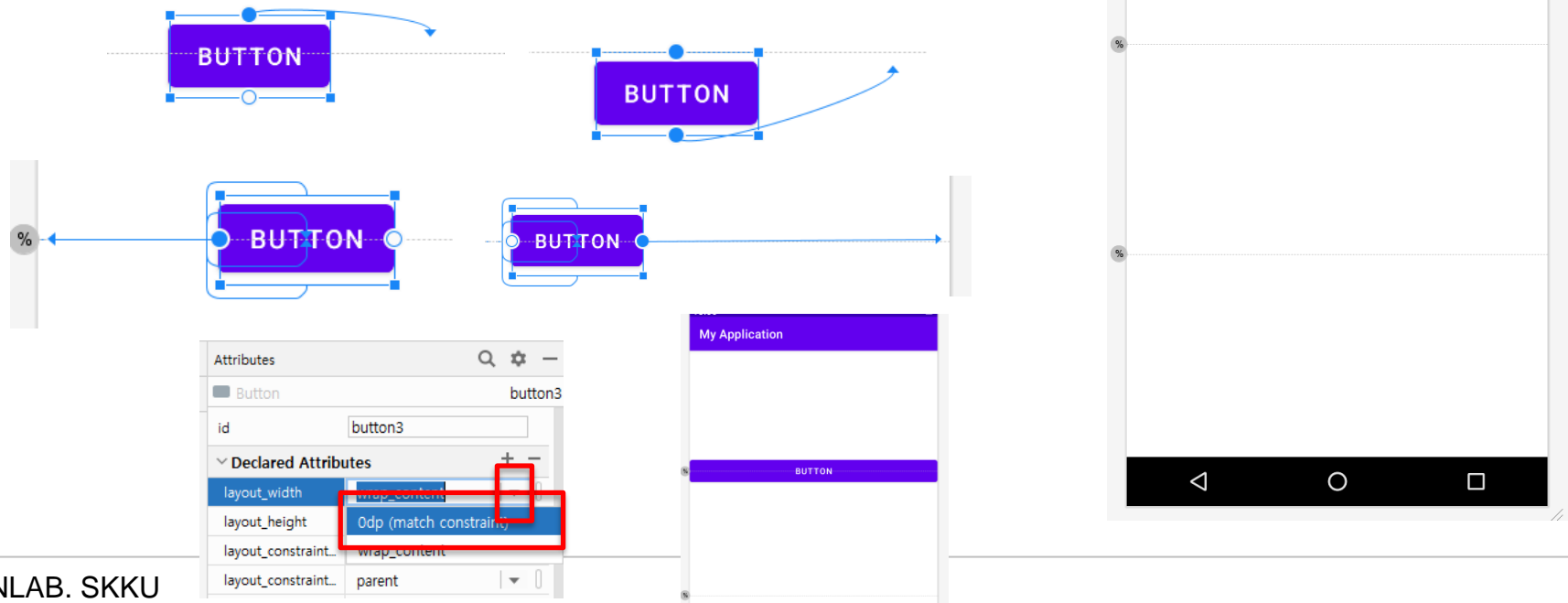


- Add one horizontal guideline and click the circle
 - It will change mode(from top / from bottom / %)
 - Set it to % and put it on 33%
- Add another and put it on 67%

```
<androidx.constraintlayout.widget.Guideline
    android:id="@+id/guideline"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintGuide_percent="0.33"
/>
```

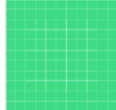
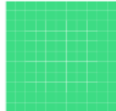
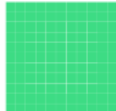
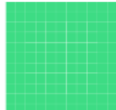
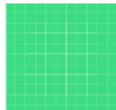

Guideline

- It will divide the screen to three (almost) identical sections
 - You can set constraint to application
- Put the button ON the above guideline



[Lab-Practice #5] Chatroom List

- We are going to make chatroom application
- There are
 - Thumbnail(Profile image)
 - Chatroom name
 - Number of people in chatroom
 - Time of last message
 - Last message preview

	Kakao T 1 Please leave review.	8:24 p.m.
	SKKU Software 512 Anybody taking MAP lecture? This week lab session is too hard. Can anybody give me the	7:15 p.m.
	Brother 1 Hey.	4:21 p.m.
	Family 4 Emoji	4:05 p.m.
	Study group 5 See you tomorrow!	4:01 p.m.
	Morning 1	3:24 p.m.

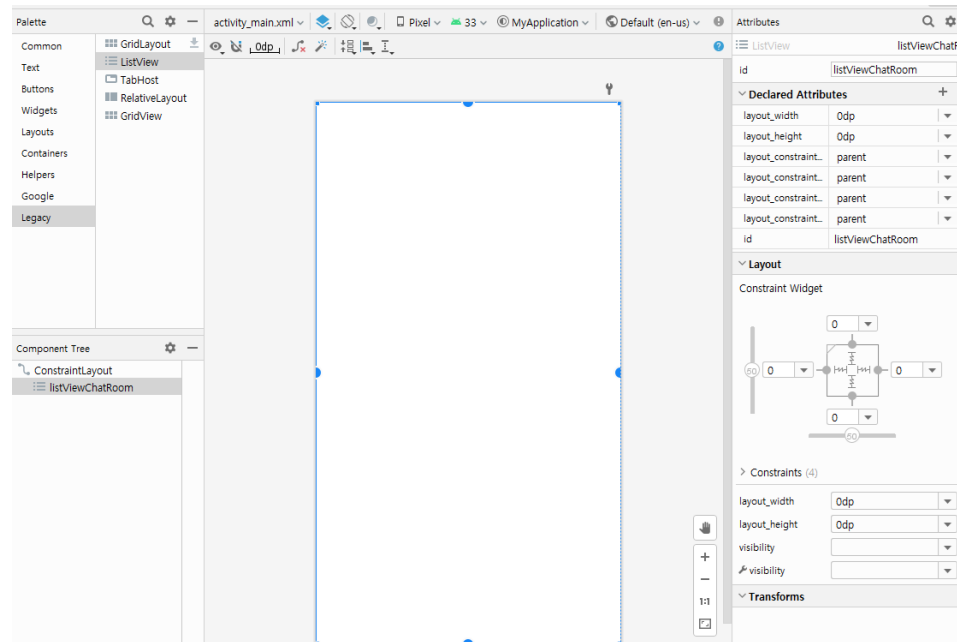
[Lab-Practice #5] Chatroom List

- First, Let's make the class to store chatroom information
 - Thumbnail
 - Chatroom name
 - Number of people in chatroom
 - Time of last message
 - Last message preview

```
class ChatRoom(  
    val name: String,  
    val lastChat: String,  
    val thumbnail: Int,  
    val groupNumber: Int,  
    val lastTime: String  
) {
```

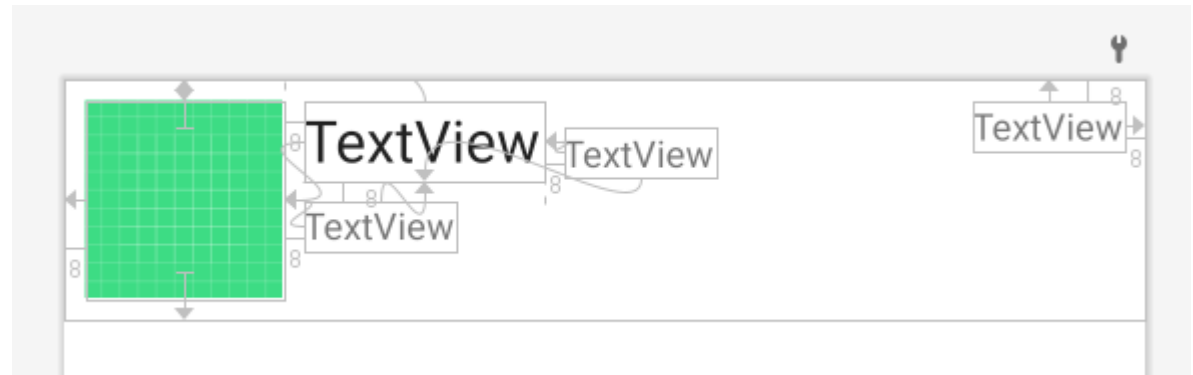
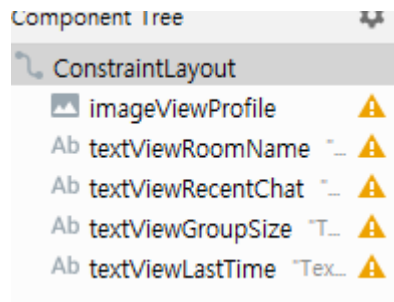
[Lab-Practice #5] Chatroom List

- Next, set MainActivity Layout
 - Just one ListView with proper id
 - Set to fill screen



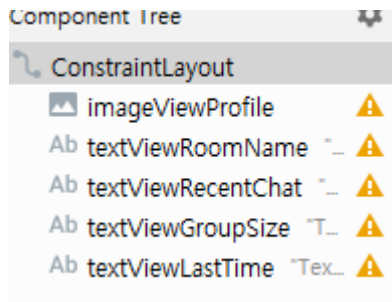
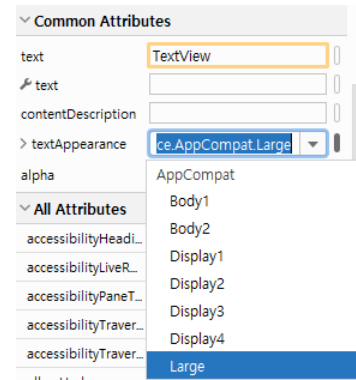
[Lab-Practice #5] Chatroom List

- Next, make layout XML for the item in list
 - Set constraints, width, height properly
 - `ConstraintLayout(parent)`: height wrap_content
 - `Profile`: top/start/bottom 8dp from `parent` top/start/bottom height/width 75dp
 - `LastTime`: top/end 8dp from `parent` top/end



[Lab-Practice #5] Chatroom List

- **RoomName**: start 8dp from **Profile** end
top same with **Profile** top
textAppearance: AppCompatActivity.Large
- **GroupSize**: start 8dp from **RoomName** end
top/bottom to **RoomName** top/bottom
- **RecentChat**: top 8dp from **RoomName** bottom
start 8dp from **Profile** end



[Lab-Practice #5] Chatroom List

- Make my own adapter
 - Same as previous week, but the number of views differs

```
class ChatRoomAdapter(val data: ArrayList<ChatRoom>, val context: Context): BaseAdapter() {  
    override fun getCount(): Int {  
        return data.size  
    }  
  
    override fun getItem(p0: Int): Any {  
        return data[p0]  
    }  
  
    override fun getItemId(p0: Int): Long {  
        return 0  
    }  
  
    override fun getView(p0: Int, p1: View?, p2: ViewGroup?): View {  
        val inflater = context.getSystemService(Context.LAYOUT_INFLATER_SERVICE) as LayoutInflater  
        val generatedView = inflater.inflate(R.layout.item_chatroom, root null)  
  
        val textViewName = generatedView.findViewById<TextView>(R.id.textViewRoomName)  
        val textViewChat = generatedView.findViewById<TextView>(R.id.textViewRecentChat)  
        val textViewGroupName = generatedView.findViewById<TextView>(R.id.textViewGroupSize)  
        val textViewTime = generatedView.findViewById<TextView>(R.id.textViewLastTime)  
        val imageViewThumbnail = generatedView.findViewById<ImageView>(R.id.imageViewProfile)  
  
        textViewName.text = data[p0].name  
        textViewChat.text = data[p0].lastChat  
        textViewTime.text = data[p0].lastTime  
        textViewGroupName.text = "" + data[p0].groupName  
        imageViewThumbnail.setImageResource(data[p0].thumbnail)  
  
        return generatedView  
    }  
}
```

[Lab-Practice #5] Chatroom List

- Finally, make some data and set adapter
 - Go to above gist link and take data
 - and rest is also same as previous week

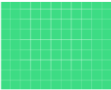





```
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val items = ArrayList<ChatRoom>()
        items.add(ChatRoom( name: "Kakao T", lastChat: "Please leave review.", R.drawable.ic_launcher_backgrou
        items.add(ChatRoom( name: "SKKU Software", lastChat: "Anybody taking MAP lecture? This week lab sessio
        items.add(ChatRoom( name: "Brother", lastChat: "Hey.", R.drawable.ic_launcher_background, groupNumber:
        items.add(ChatRoom( name: "Family", lastChat: "Emoji", R.drawable.ic_launcher_background, groupNumber:
        items.add(ChatRoom( name: "Study group", lastChat: "See you tomorrow!", R.drawable.ic_launcher_backgro
        items.add(ChatRoom( name: "Yogiyo", lastChat: "How was the food?", R.drawable.ic_launcher_background,
        items.add(ChatRoom( name: "Lorem ipsum", lastChat: "dolor", R.drawable.ic_launcher_background, groupN
        items.add(ChatRoom( name: "Placeholder", lastChat: "Placeholder", R.drawable.ic_launcher_background,

        val myAdapter = ChatRoomAdapter(items, applicationContext)
        val listView = findViewById<ListView>(R.id.listViewChatRoom)
        listView.adapter = myAdapter
    }
}
```

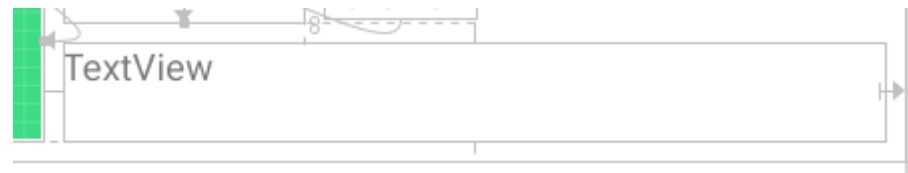
[Lab-Practice #5] Chatroom List

- There is a problem!
 - Long recent chat will break the layout
 - Right side is cutting, and item height is different

	Nakao 1 Please leave review.	
	SKKU Software 512 Anybody taking MAP lecture? This week lab session Can anybody give me the hint? I will be very happy if nobody help me, I will be very sad.	7:15 p.m.
	Brother 1 Hey.	4:21 p.m.
	Family 4 Emoji	4:05 p.m.
	Study group 5 See you tomorrow!	4:01 p.m.
	Yoaivo 1	3:24 p.m.

[Lab-Practice #5] Chatroom List

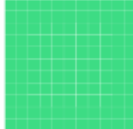
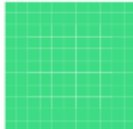
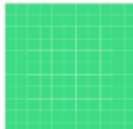
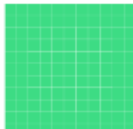
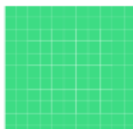
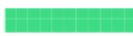
- Add constraint and modify properties of **RecentChat**



- top: 8dp from **RoomName** bottom
 - bottom: same as **Profile** bottom
 - start: 8dp from **Profile** end
 - end: 8dp from **parent** end
 - width & height: match_constraint
- Constraint height not to go below than imageView
 - Constraint width that having 8dp margin to screen

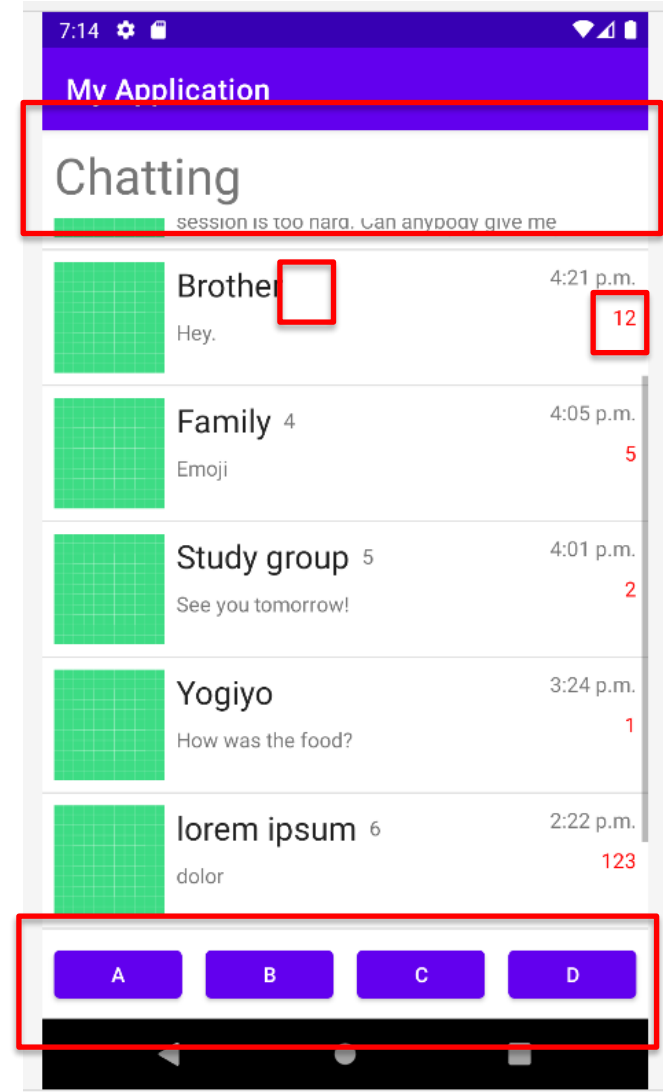
[Lab-Practice #5] Chatroom List

- Now it shows well

	Kakao T 1	8:24 p.m.
	Please leave review.	
	SKKU Software 512	7:15 p.m.
	Anybody taking MAP lecture? This week lab session is too hard. Can anybody give me the	
	Brother 1	4:21 p.m.
	Hey.	
	Family 4	4:05 p.m.
	Emoji	
	Study group 5	4:01 p.m.
	See you tomorrow!	
	Yoonja 1	3:24 p.m.

[Lab-Practice #5] Chatroom List

- Now, you must do
 - Add “Unread message count” view on item
 - Hide number of people in group if it is 1
 - Add header and footer



[Lab-Practice #5] Chatroom List

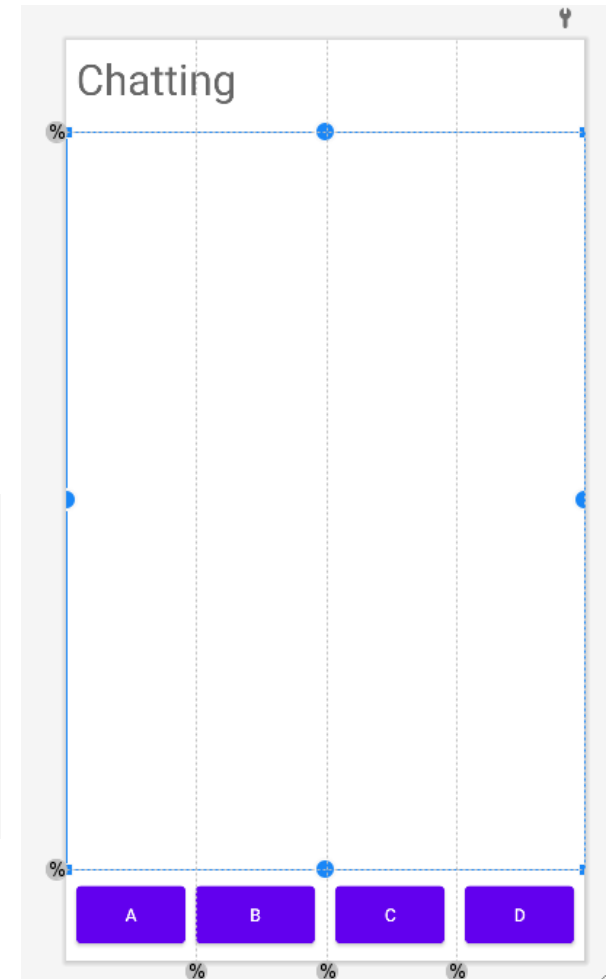
- Now, you must do
 - Add “Unread message count” view on item
 - Red color, 8dp below `LastTime`
 - Adjust end constraint of `RecentChat`
 - Hide number of people in group if it is 1
 - Add header and footer of the screen
 - Header: 10% top of screen, one TextView
 - TextView text is “Chatting”
 - Footer: 10% bottom of screen, 4 Buttons
 - Button width must be equal with 8dp margin
 - Button text is A, B, C, D each

[Lab-Practice #5] Chatroom List

- Hint
 - Make use of above gist link for chatroom data insertion
 - Modify “textColor” attribute for text color
 - Use if statement in adapter, for #people in group
 - `textView.text = if (a==1) “one” else “two or more”`
 - Use guideline for setting up MainActivity layout
 - If you are hard to tagging the view,
 - modify width and height to 50dp
 - and remove all constraints (w/ Ctrl+Click or attribute)
 - Set up constraint, and then set width and height

[Lab-Practice #5] Chatroom List

- Hint
 - You might modify all files
 - 3 Kotlin files, 2 XML files
 - Use Guideline for MainActivity!



[Lab-Practice #5] Chatroom List

- Criteria
 - Set up 4 buttons properly
 - Set up ListView constraint properly
 - Properly hiding number of people in group when it is 1
 - Properly added unread message count
 - position and color(red)
 - Properly adjusted last message position
 - bounded to last message count text
 - NOT care about “Chatting” text detailed position(just left-upper then it is ok) and its size