# Explicit Intent

Mobile App Programming
Fall, 2024

## Today’s Contents

- Review

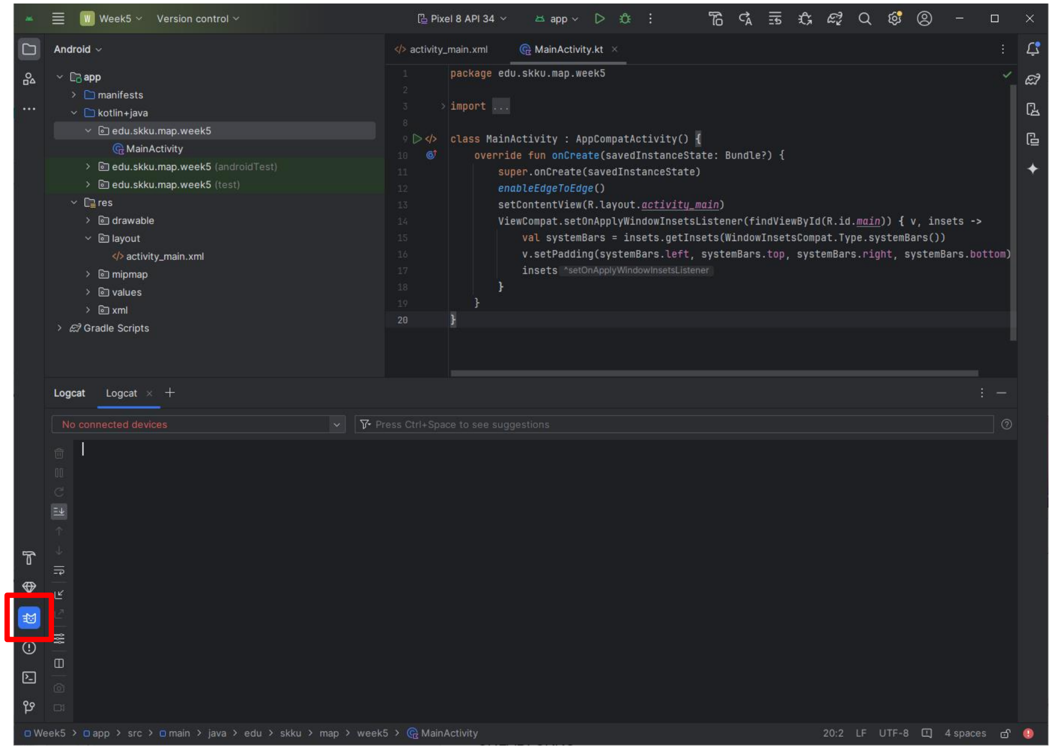
- Intent
  - Intent
  - Explicit intent and Implicit intent

- Lab practice

## Review

# Logcat

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

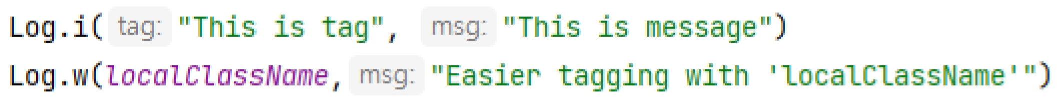


```kotlin
package edu.skku.map.week5

import ...

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

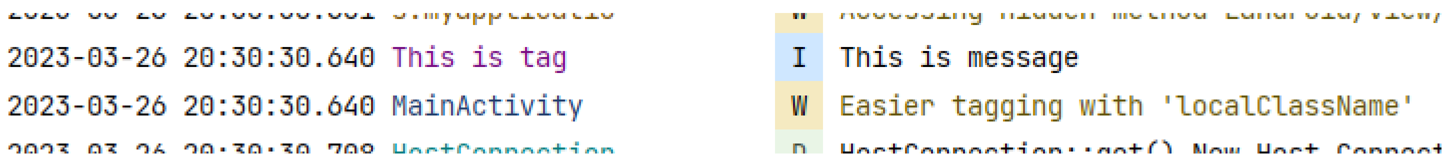
## Logcat

- You can manually log
  - **Log.v/d/i/w/e("tag string", "message string")**
  - Each alphabet represents: <u>verbose/debug/info/warn/error</u>

  - **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'")
```

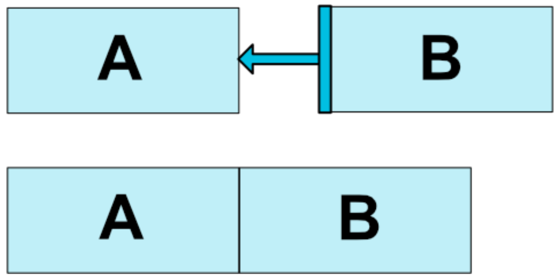


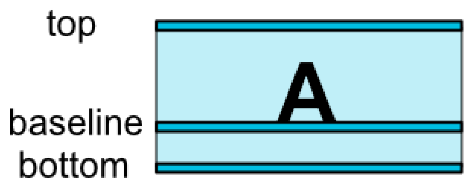
```
2023-03-26 20:30:30.640 This is tag          I  This is message
2023-03-26 20:30:30.640 MainActivity         W  Easier tagging with 'localClassName'
```

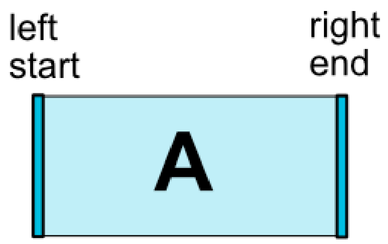
## ConstraintLayout

- Position
  - Define the position relative to other widgets
  - `app:layout_constraint{}_to{}Of`

- Size
  - Define the size with a constraint
  - e.g.) wrap_content / match_parent / 0dp

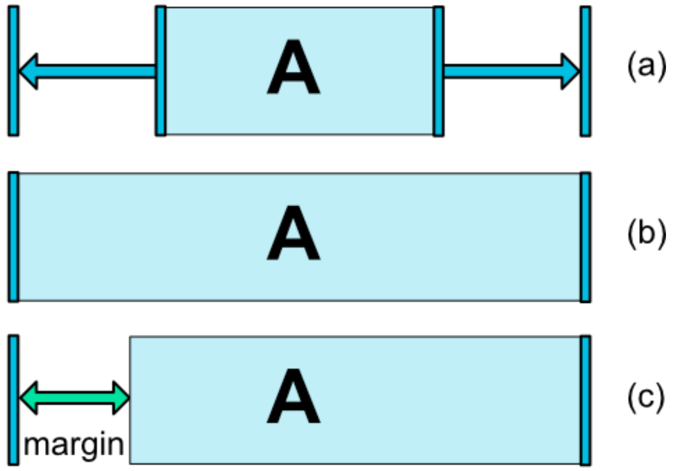






## ConstraintLayout

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



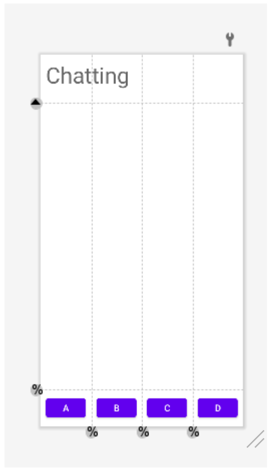
- (a): width = "**wrap_content**"
- (b): width = "**0dp**" (or "match_constraint")
- (c): width = "0dp" & has <u>start margin</u>

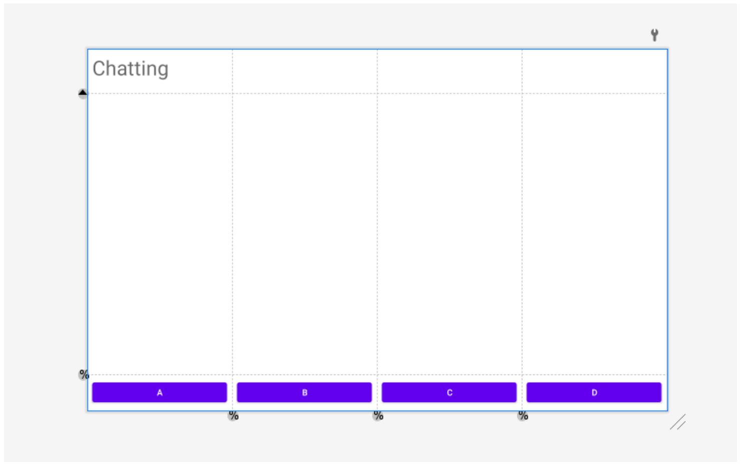
## ConstraintLayout

- Why is a fixed size not recommended?
  - There are various of screen (Different size of Phones, Tablets ...)
  - "X dp" size means, its <u>physical length is fixed</u>
    - Thanks to dp, we don't need to care resolution
    - But we <u>need to care physical size</u> ☹
  - Your screen layout looks well…
    But will it be on other screen size? like tablet?
  - Not saying "X dp" is always bad, but just use in place!

## ConstraintLayout

- If size of large view is fixed, (my screen is 3 inch wide -> 480 dp)
  - It can be cut on the smaller screen (2 inch wide screen -> cannot see right 33%)
  - It can be small on larger screen (12 inch wide screen -> only 33% view)
- That's why we use match_constraint and *% guideline!





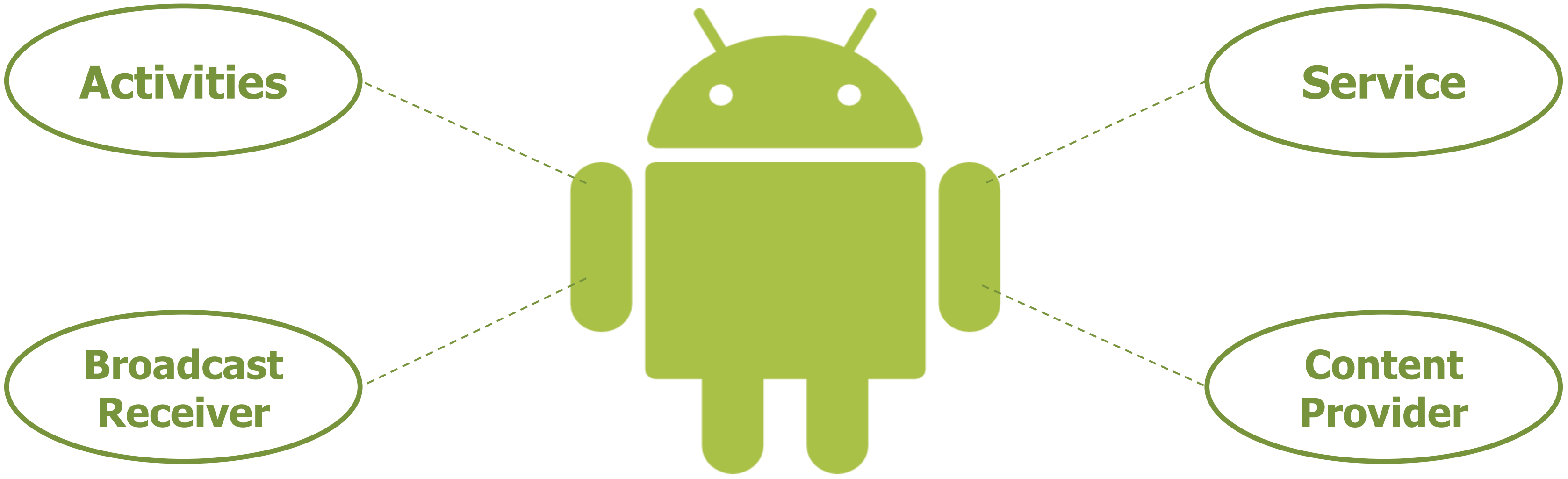
- Not one-size-fits-all, use accordingly!

## Intent

## Android Glossary

- There are four types of main app components:
  → Activity, Service, Broadcast receiver, Content provider

- Whenever you create or use any of them, you must include elements in the **project manifest**.



## Intent

- An Intent is a messaging object you can use to request an action from another app component.
- Three fundamental use cases:
  - Starting an activity
  - Starting a service
  - Delivering a broadcast
- Two types of intents:
  - Explicit intents
  - Implicit intents

https://developer.android.com/guide/components/intents-filters?hl=ko

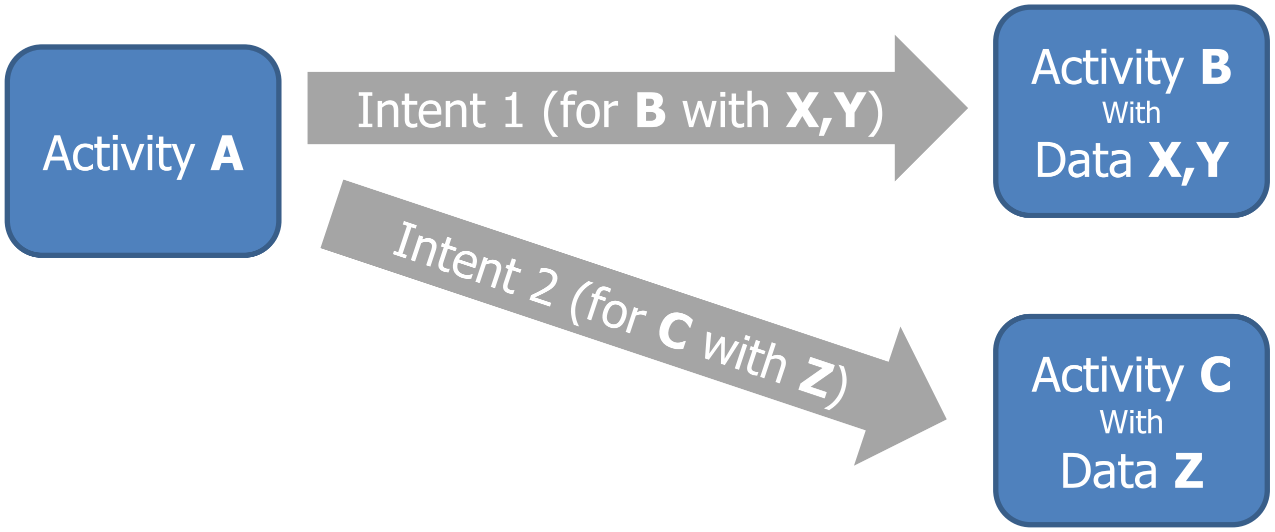
## Implicit/Explicit Intent

- **Explicit** intents
  - Calling by specifying activity name or service to run
  - Use when you know the class name want to execute.

- **Implicit** intents
  - The class name of the activity or service is <u>not specified</u>.
  - Just request general action (e.g. Open Internet, Call, Message...)
  - Android system finds and executes an activity or app through an intent filter.

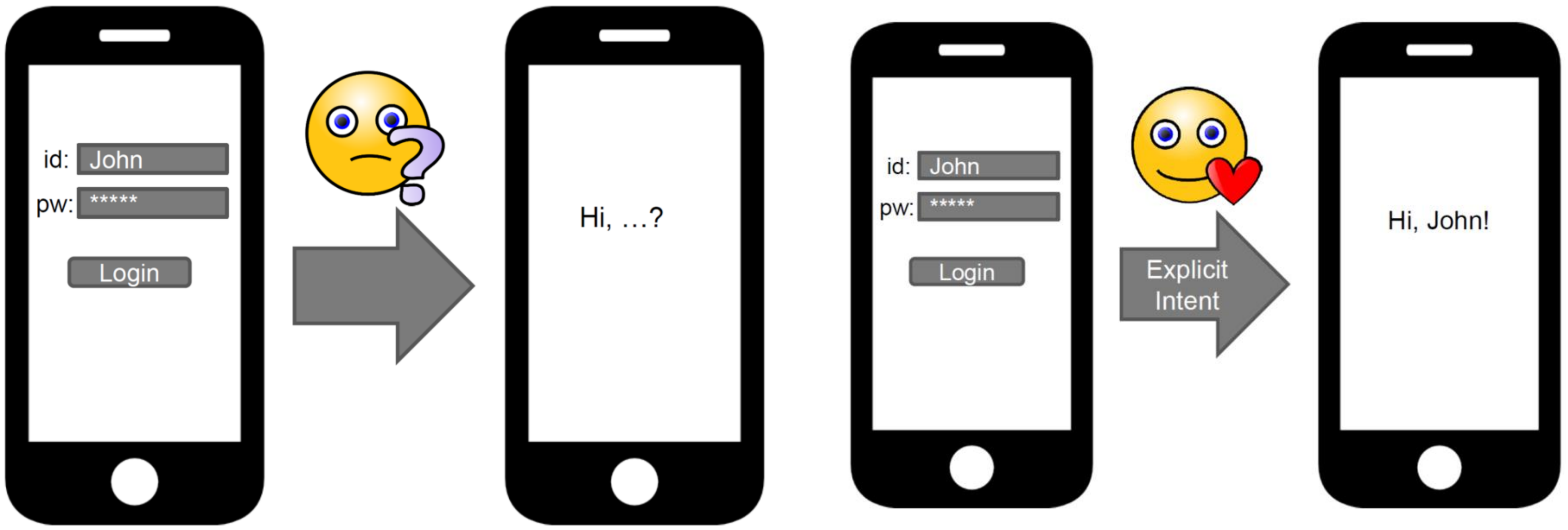
# Explicit Intent

- Explicit intents
  - Specify package name or component class name
  - Start an activity in Intent object
  - Data can be passed via Extras


Activity **A**
Intent 1 (for **B** with **X,Y**)
Activity **B** With Data **X,Y**
Intent 2 (for **C** with **Z**)
Activity **C** With Data **Z**

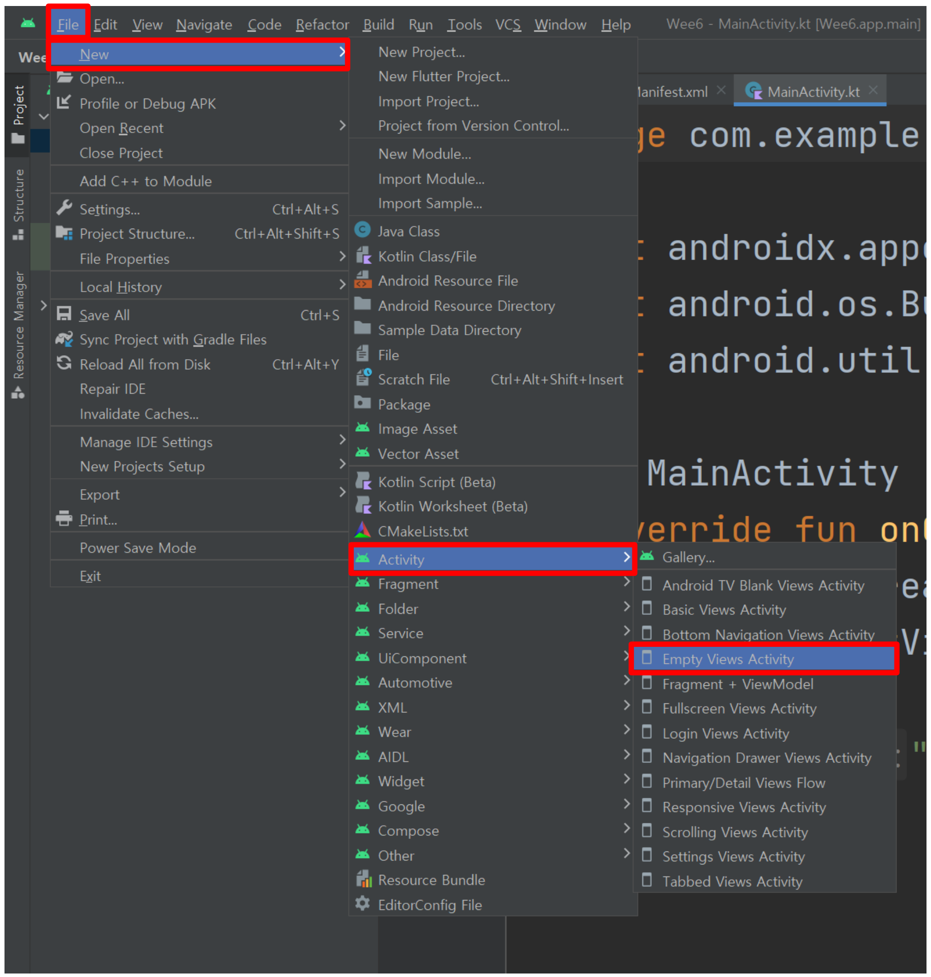
# Explicit Intent

- Why Extras needed?



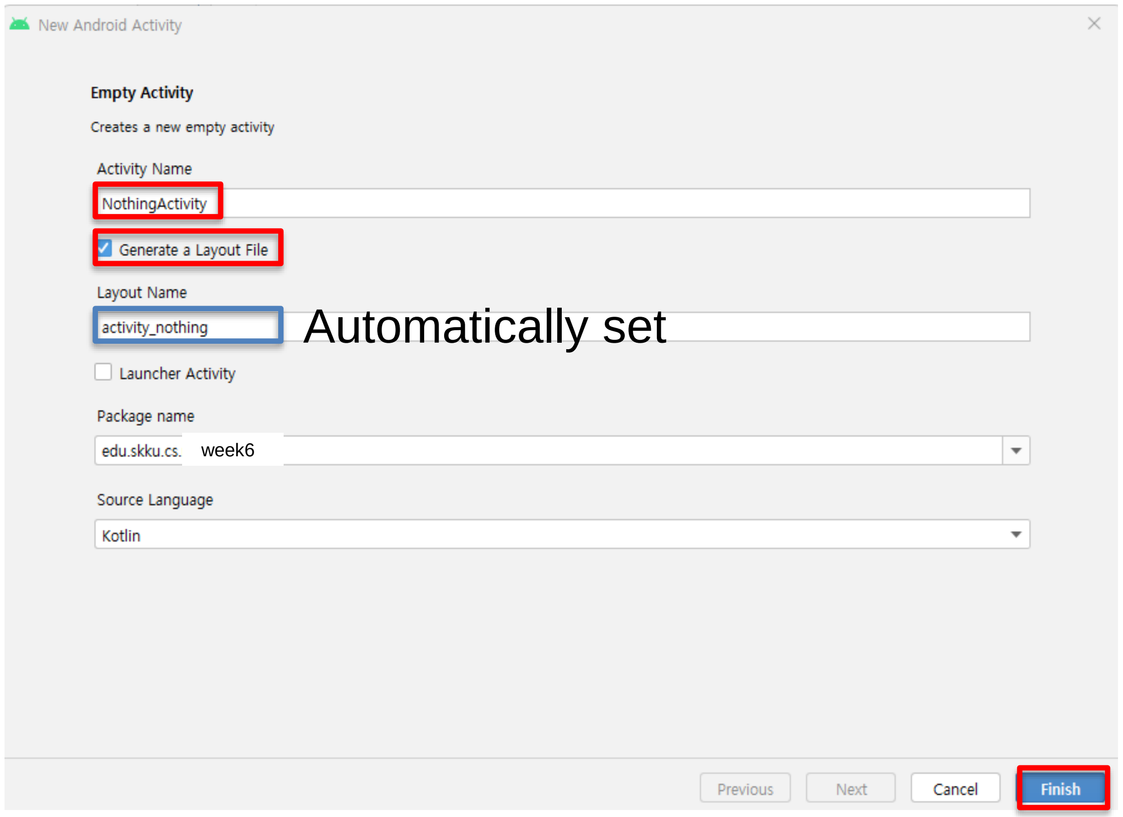
# Exercise: Another Activity

- Make another activity
  - File > New > Activity > Empty Views Activity



# Exercise: Another Activity

- Make another activity



New Android Activity

**Empty Activity**

Creates a new empty activity

Activity Name: NothingActivity

☑ Generate a Layout File

Layout Name: activity_nothing

Automatically set

☐ Launcher Activity

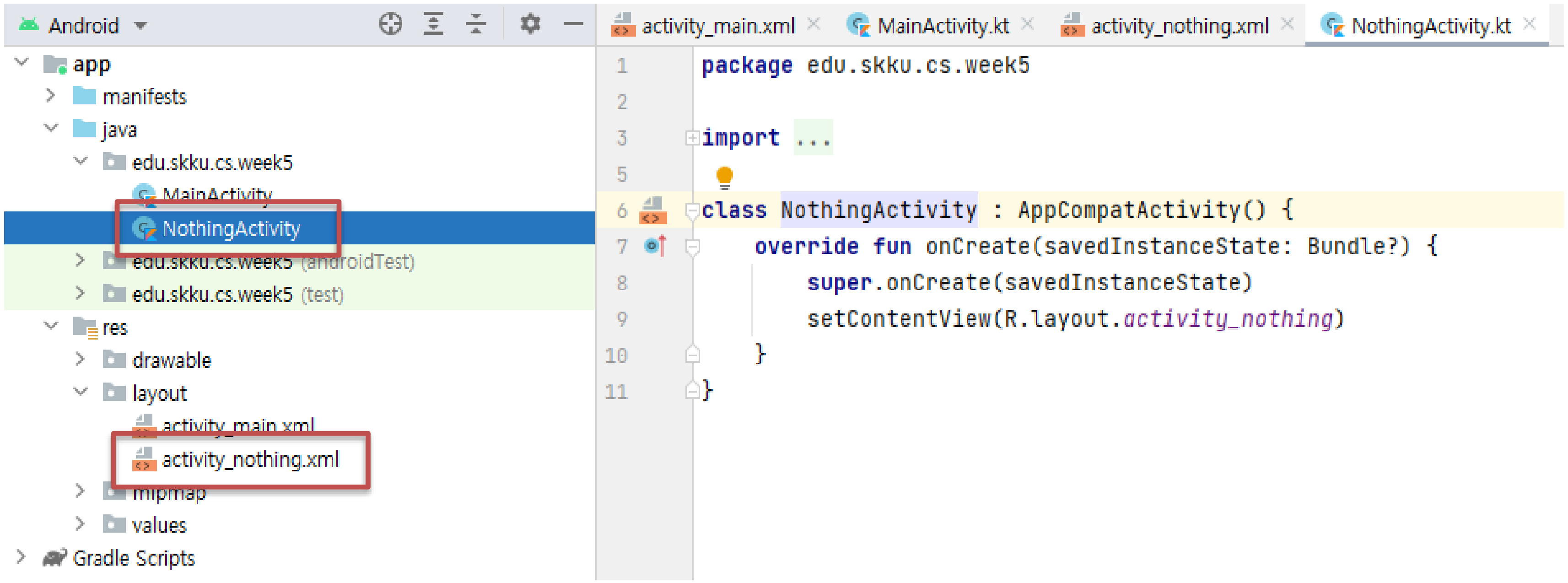
Package name: edu.skku.cs. week6

Source Language: Kotlin

https://gist.github.com/devquint/284293b8bf2d66f196bee5e6507c603e

# Exercise: Another Activity

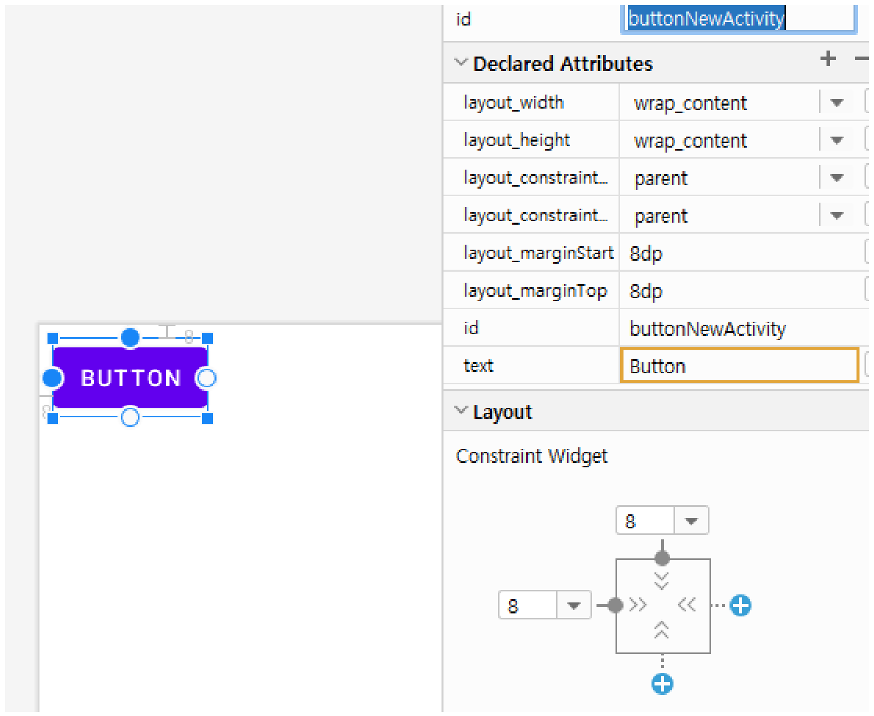
- Make another activity



```kotlin
package edu.skku.cs.week5

import ...

class NothingActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_nothing)
    }
}
```

## Exercise: Another Activity

- Add button on MainActivity



| | |
|---|---|
| id | buttonNewActivity |

Declared Attributes

| | |
|---|---|
| layout_width | wrap_content |
| layout_height | wrap_content |
| layout_constraint… | parent |
| layout_constraint… | parent |
| layout_marginStart | 8dp |
| layout_marginTop | 8dp |
| id | buttonNewActivity |
| text | Button |

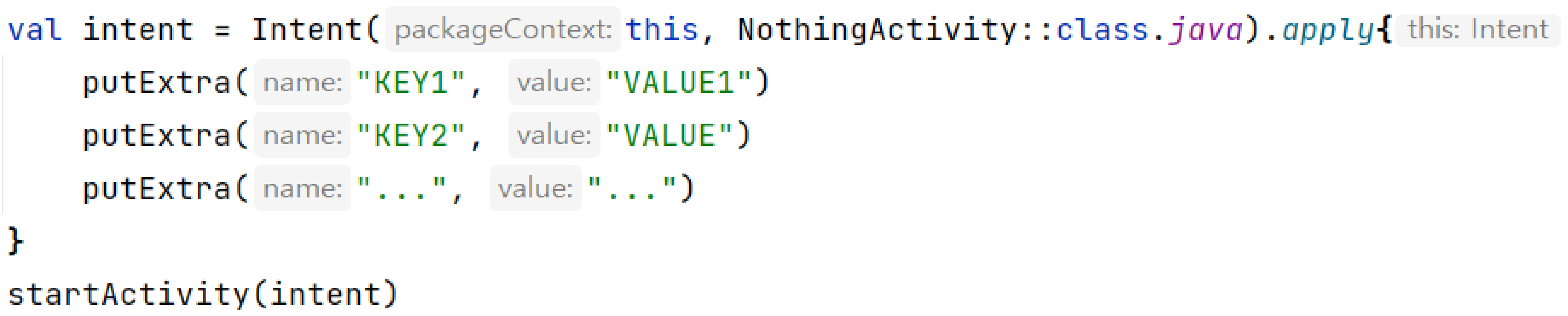
Layout

Constraint Widget

https://gist.github.com/devquint/284293b8bf2d66f196bee5e6507c603e

## Exercise: Another Activity

- Add onClickListener
  - to start new activity
    - explicit intent: specify activity class
    - Make Intent
    - Put extras
    - Call startActivity

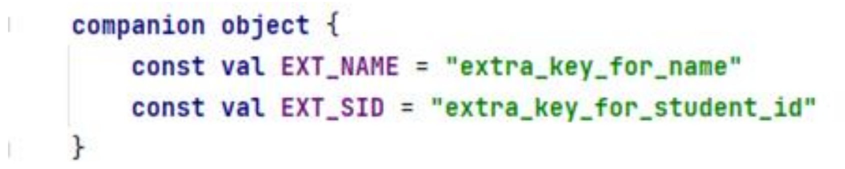


```
val intent = Intent( packageContext: this, NothingActivity::class.java).apply{ this: Intent
    putExtra( name: "KEY1",  value: "VALUE1")
    putExtra( name: "KEY2",  value: "VALUE")
    putExtra( name: "...",  value: "...")
}
startActivity(intent)
```

https://gist.github.com/devquint/284293b8bf2d66f196bee5e6507c603e

## Exercise: Another Activity

- Why we have to use companion object?
  - like Java 'static', objects declared within a class
  - If you don't use companion object...
    - Sender: intent.putExtra("student_name", "name_value")
    - Receiver: name = intent.getStringExtra("student_name")
  - It is okay, but what if "student_name" is used in another class?
    - It has risk of overwritten or data colliding
  - Using Companion Object can avoid those problems!
    - intent.putExtra(MainActivity.EXT_NAME, "name_value")

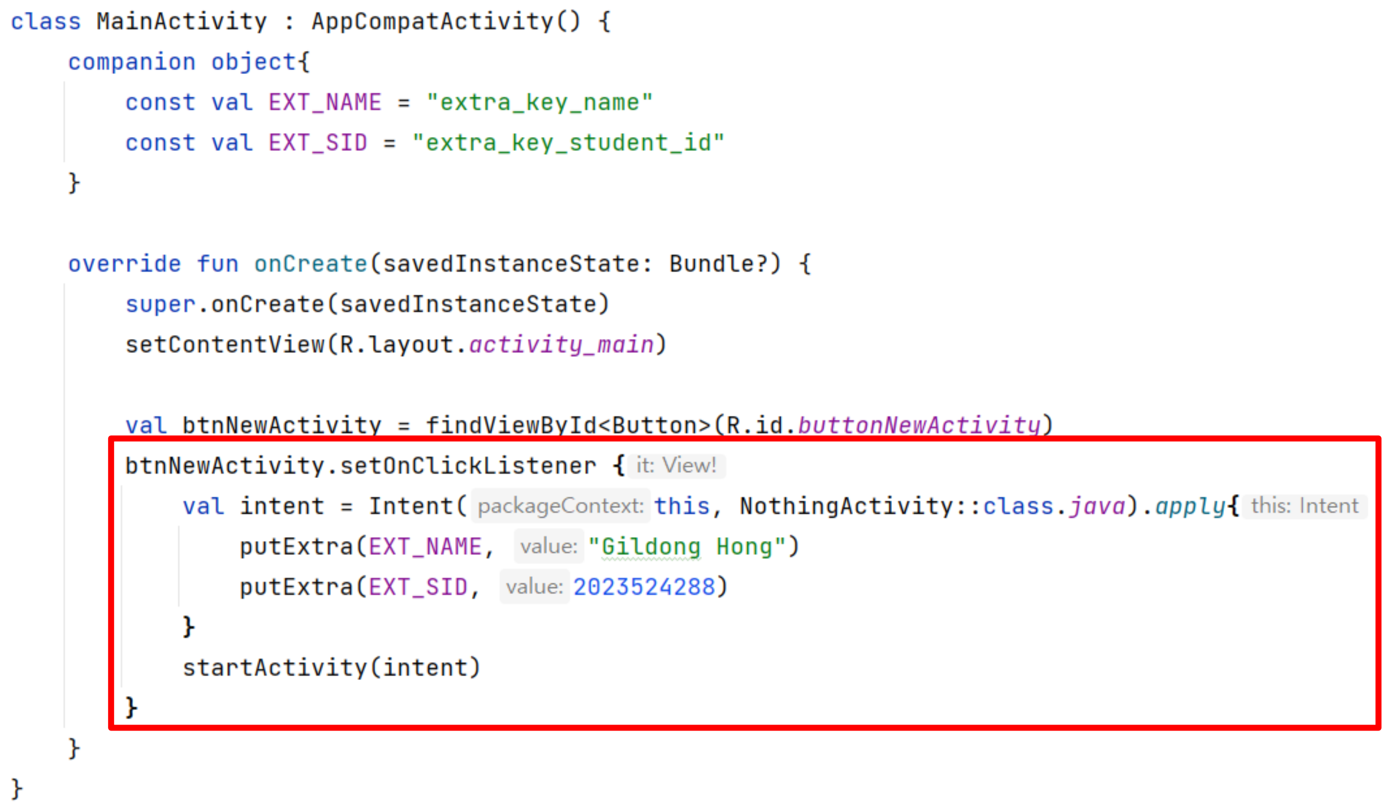


```
companion object {
    const val EXT_NAME = "extra_key_for_name"
    const val EXT_SID = "extra_key_for_student_id"
}
```

https://gist.github.com/devquint/284293b8bf2d66f196bee5e6507c603e

## Exercise: Another Activity

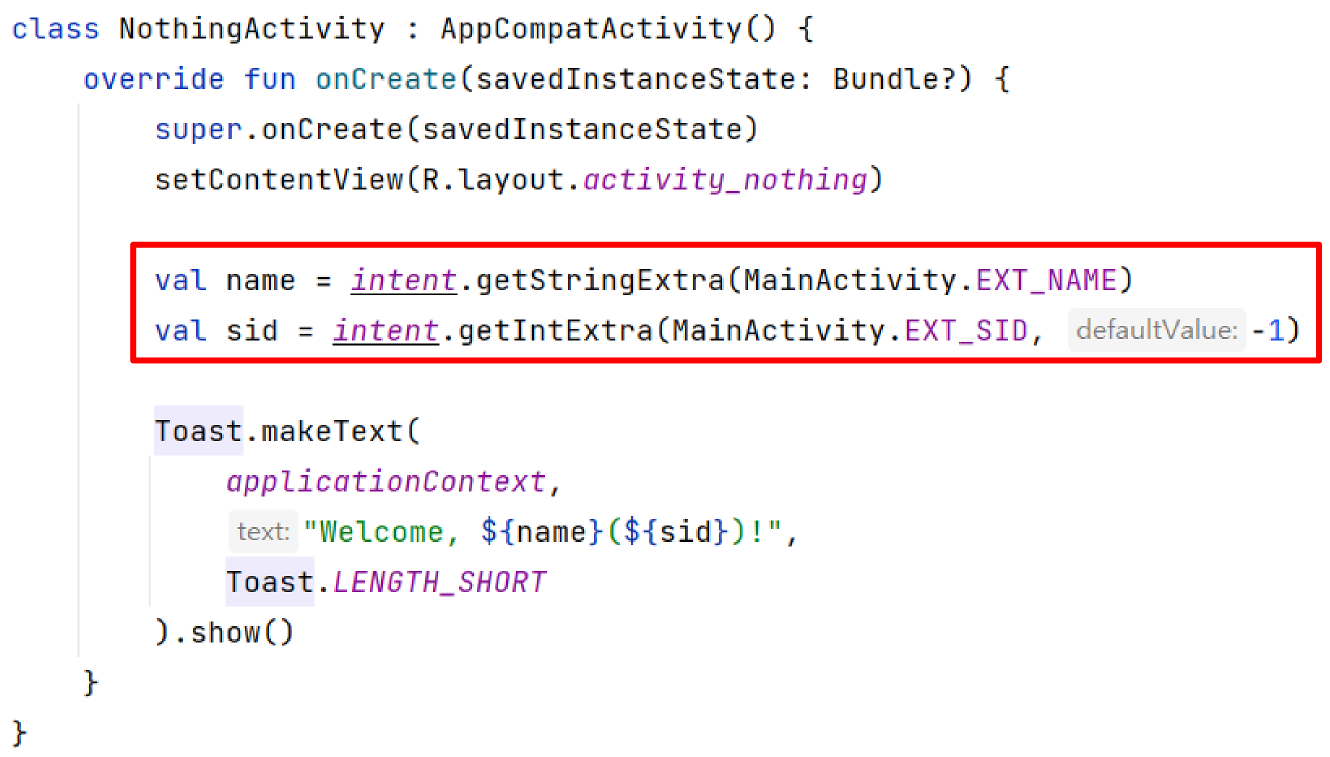
- Add onClickListener



```
class MainActivity : AppCompatActivity() {
    companion object{
        const val EXT_NAME = "extra_key_name"
        const val EXT_SID = "extra_key_student_id"
    }

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

        val btnNewActivity = findViewById<Button>(R.id.buttonNewActivity)
        btnNewActivity.setOnClickListener { it: View!
            val intent = Intent( packageContext: this, NothingActivity::class.java).apply{ this: Intent
                putExtra(EXT_NAME,  value: "Gildong Hong")
                putExtra(EXT_SID,  value: 2023524288)
            }
            startActivity(intent)
        }
    }
}
```

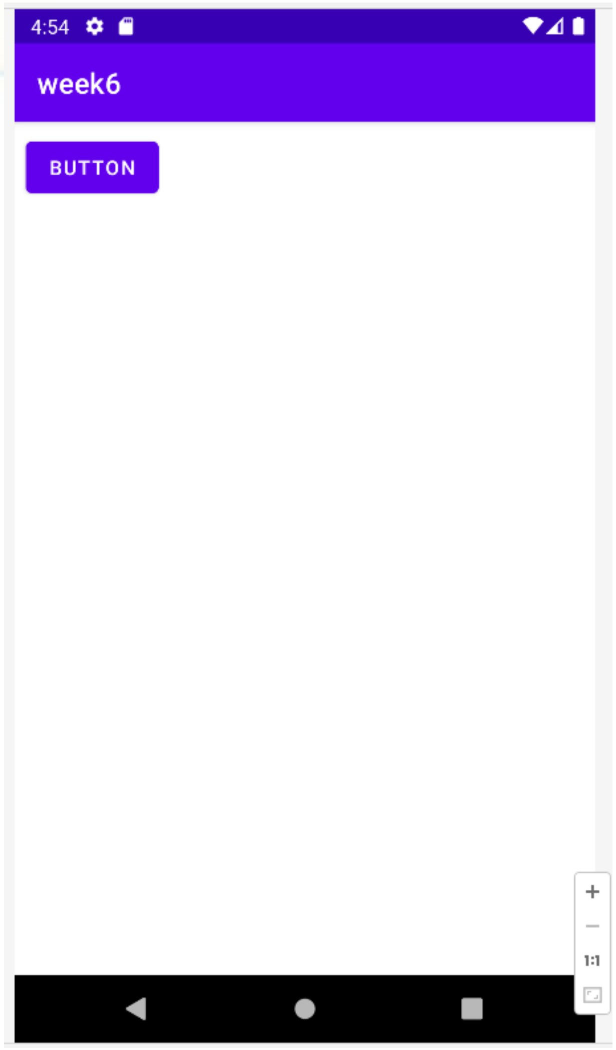
## Exercise: Another Activity

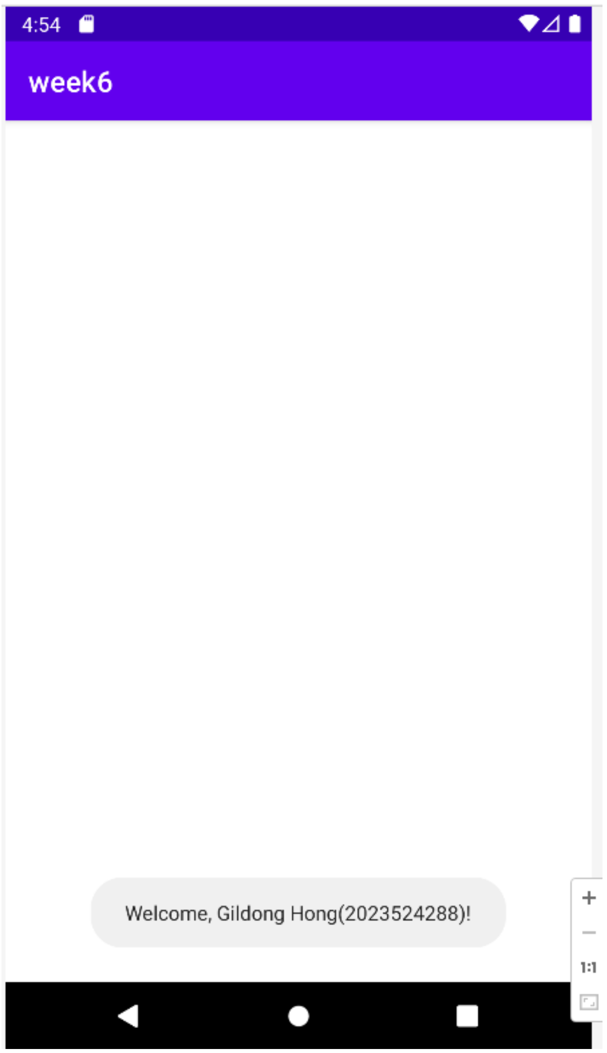
- Get extra and show toast on NothingActivity
  - onCreate() called when activity started!



```kotlin
class NothingActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_nothing)

        val name = intent.getStringExtra(MainActivity.EXT_NAME)
        val sid = intent.getIntExtra(MainActivity.EXT_SID, defaultValue: -1)

        Toast.makeText(
            applicationContext,
            text: "Welcome, ${name}(${sid})!",
            Toast.LENGTH_SHORT
        ).show()
    }
}
```

## Another Activity

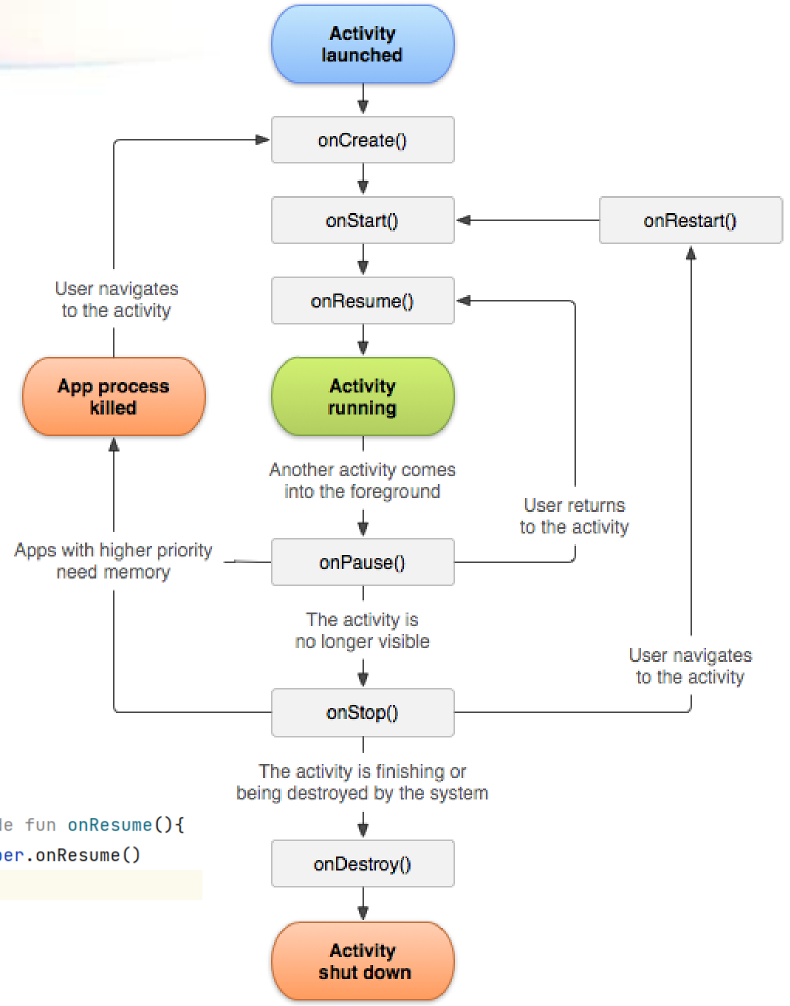




Welcome, Gildong Hong(2023524288)!

https://gist.github.com/devquint/2a672a6428c21879fe178ee64f556be3

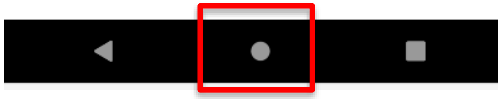
# Activity Lifecycle

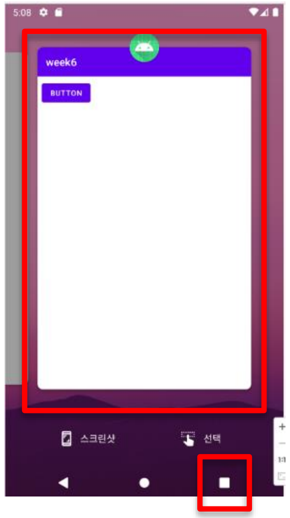


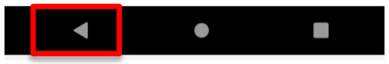
Activity launched
onCreate()
onStart()
onRestart()
User navigates to the activity
onResume()
App process killed
Activity running
Another activity comes into the foreground
User returns to the activity
Apps with higher priority need memory
onPause()
The activity is no longer visible
User navigates to the activity
onStop()
The activity is finishing or being destroyed by the system
onDestroy()
Activity shut down

```
override fun onResume(){
    super.onResume()

}
```

- startActivity
  - onCreate
  - onStart
  - onResume
- Pressing home button
  - onPause
  - onStop
- Get back to application
  - onRestart
  - onStart
  - onResume
- Back button(close)
  - onPause
  - onStop
  - onDestroy





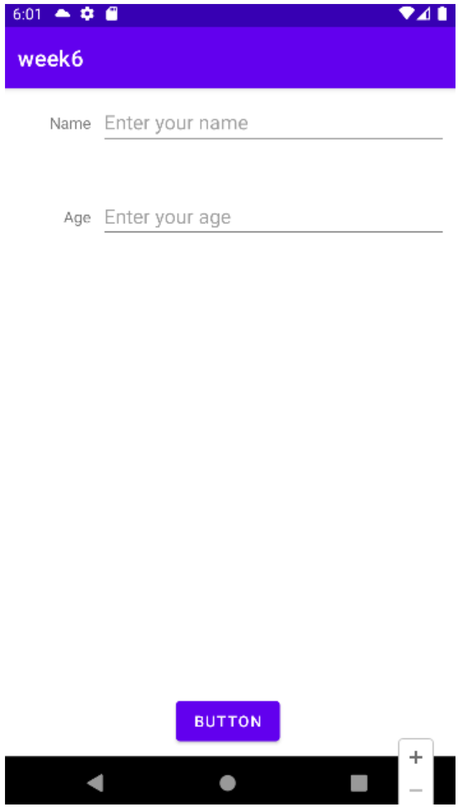


https://developer.android.com/guide/components/activities/activity-lifecycle

## [Lab-Practice #6] Hello!

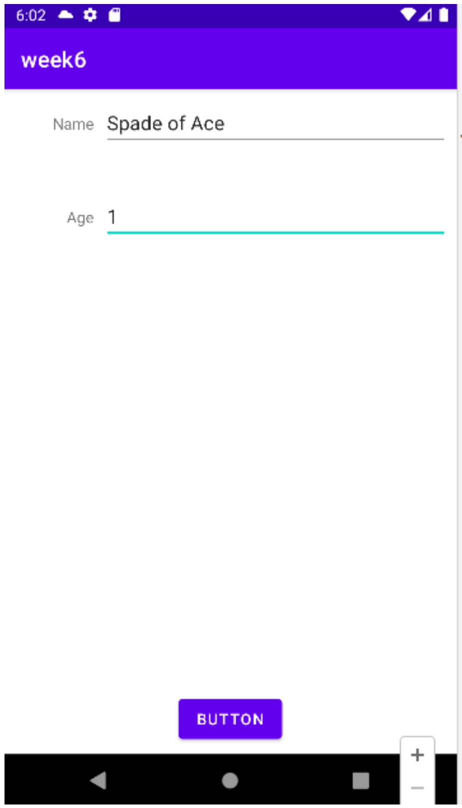
- We are going to make simple explicit intent app

**Activity 1**



Name Enter your name

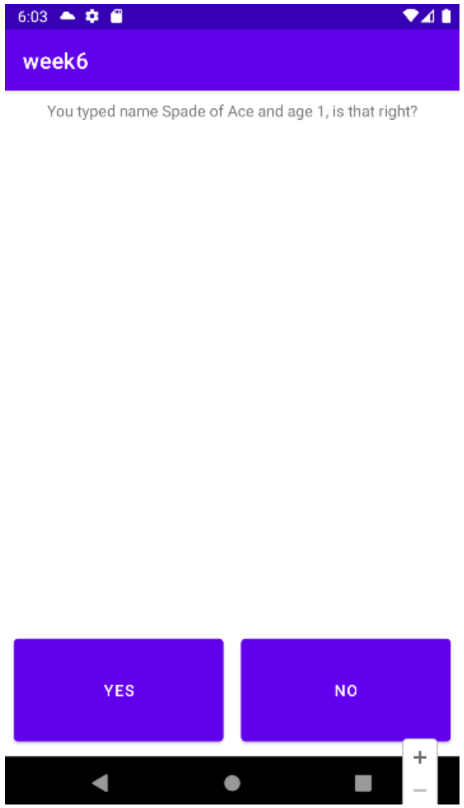
Age Enter your age



Name Spade of Ace

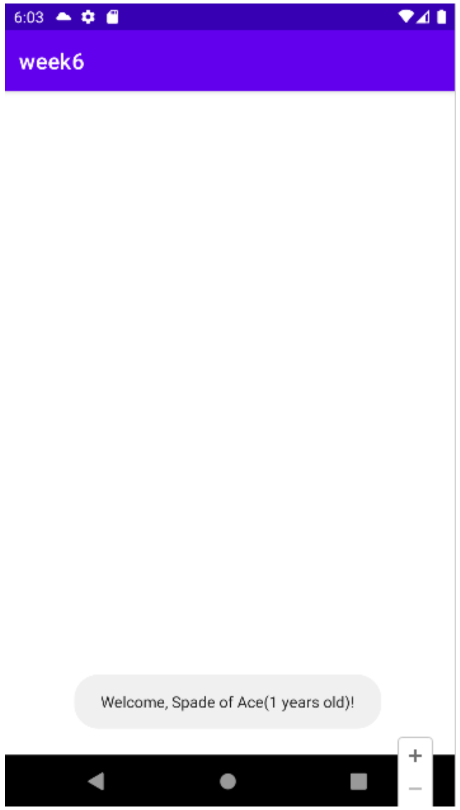
Age 1

**Activity 2**



You typed name Spade of Ace and age 1, is that right?

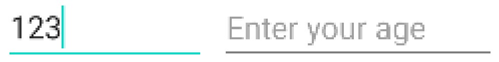
**Activity 3**

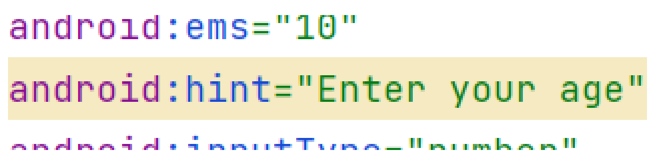


Welcome, Spade of Ace(1 years old)!

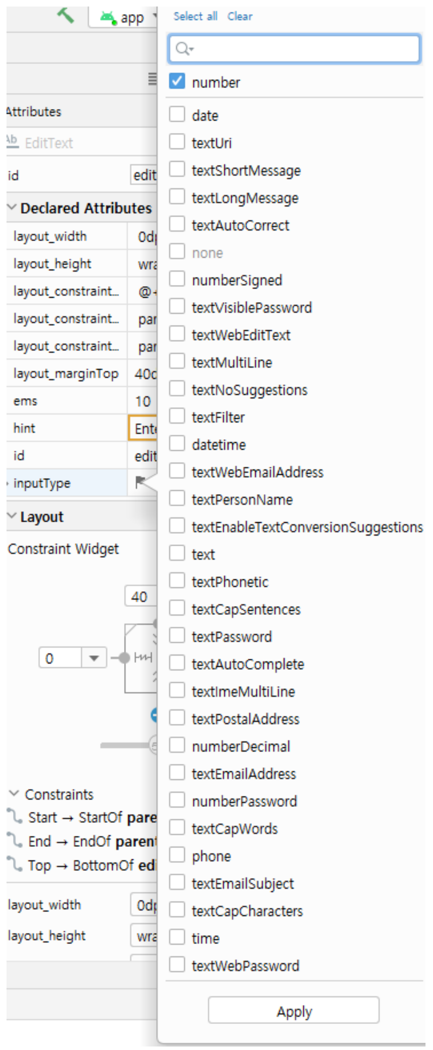
# [Lab-Practice #6] Hello!

- EditText
  - Editable text
  - Input type can be restricted
  - Hint text is shown when there is no text





```
android:ems="10"
android:hint="Enter your age"
android:inputType="number"
```



Attributes

EditText

| | |
|---|---|
| id | edit |

Declared Attributes

| | |
|---|---|
| layout_width | 0dp |
| layout_height | wra |
| layout_constraint... | @ |
| layout_constraint... | par |
| layout_constraint... | par |
| layout_marginTop | 40d |
| ems | 10 |
| hint | Ente |
| id | edit |
| inputType | |

Layout

Constraint Widget

Constraints
Start → StartOf pare
End → EndOf parent
Top → BottomOf ed

| | |
|---|---|
| layout_width | 0dp |
| layout_height | wra |

- ✓ number
- ☐ date
- ☐ textUri
- ☐ textShortMessage
- ☐ textLongMessage
- ☐ textAutoCorrect
- ☐ none
- ☐ numberSigned
- ☐ textVisiblePassword
- ☐ textWebEditText
- ☐ textMultiLine
- ☐ textNoSuggestions
- ☐ textFilter
- ☐ datetime
- ☐ textWebEmailAddress
- ☐ textPersonName
- ☐ textEnableTextConversionSuggestions
- ☐ text
- ☐ textPhonetic
- ☐ textCapSentences
- ☐ textPassword
- ☐ textAutoComplete
- ☐ textImeMultiLine
- ☐ textPostalAddress
- ☐ numberDecimal
- ☐ textEmailAddress
- ☐ numberPassword
- ☐ textCapWords
- ☐ phone
- ☐ textEmailSubject
- ☐ textCapCharacters
- ☐ time
- ☐ textWebPassword

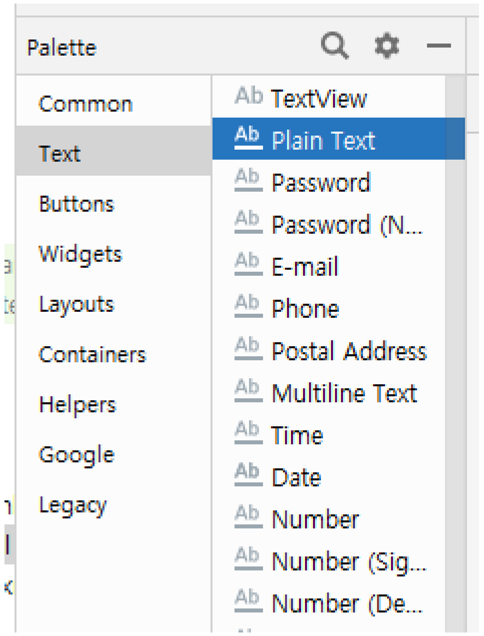
## [Lab-Practice #6] Hello!

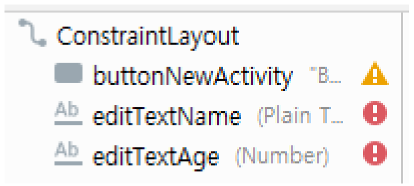
- EditText
  - editTextInstance.text = “”
    - text attribute will return “Editable”, not String
  - editableInstance.toString() to get String
  - Therefore,
    - editTextInstance.text.toString()
    - If integer needed,
      - editTextInstance.text.toString().toInt()
  - If you want to clear Text,
    - editTextInstance.text.clear()

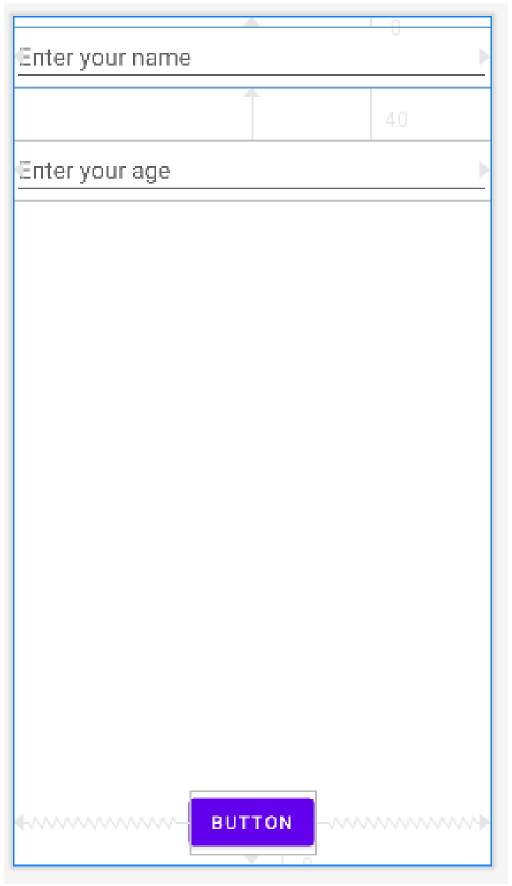
https://gist.github.com/devquint/153032cea7bad81aed83f3bdc0a7faf6

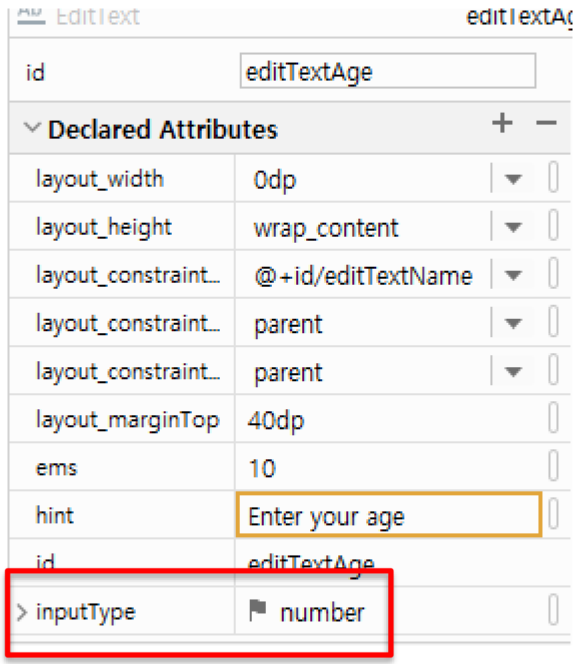
## [Lab-Practice #6] Hello!

- EditText








EditText editTextAg

| | |
|---|---|
| id | editTextAge |

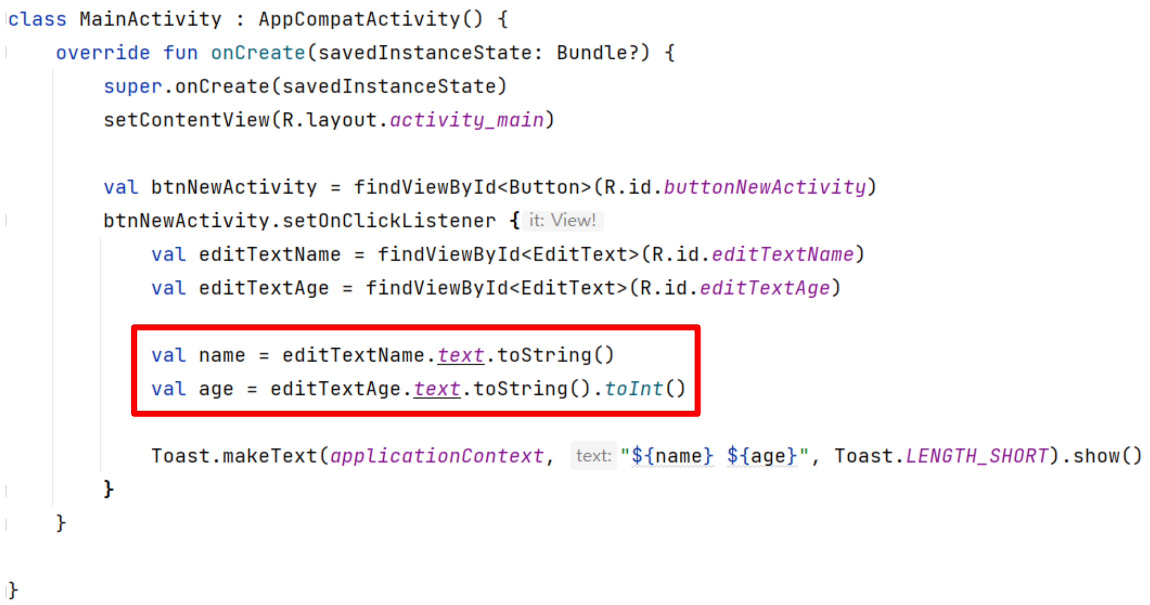
Declared Attributes

| | |
|---|---|
| layout_width | 0dp |
| layout_height | wrap_content |
| layout_constraint... | @+id/editTextName |
| layout_constraint... | parent |
| layout_constraint... | parent |
| layout_marginTop | 40dp |
| ems | 10 |
| hint | Enter your age |
| id | editTextAge |
| inputType | number |

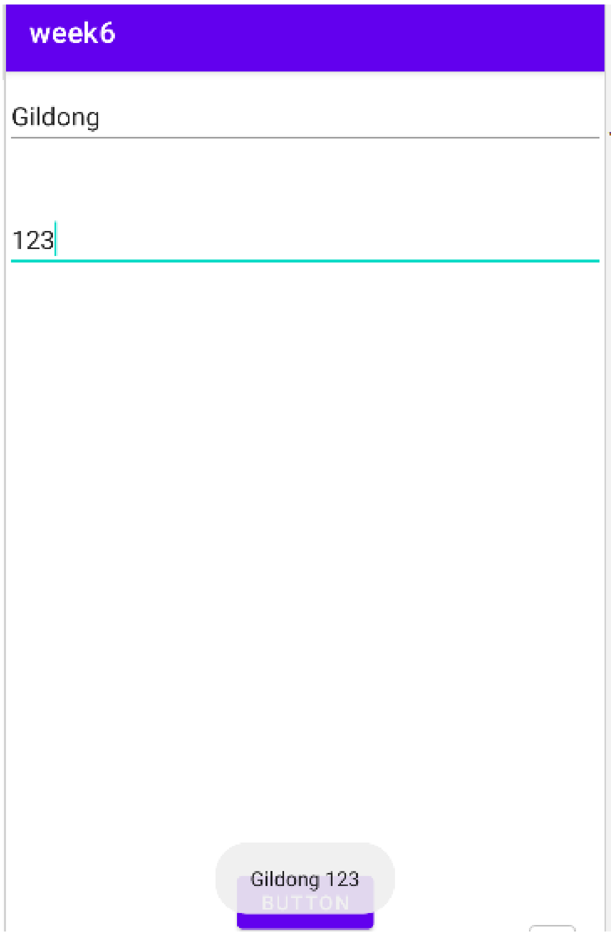
https://gist.github.com/devquint/153032cea7bad81aed83f3bdc0a7faf6

# [Lab-Practice #6] Hello!

- EditText



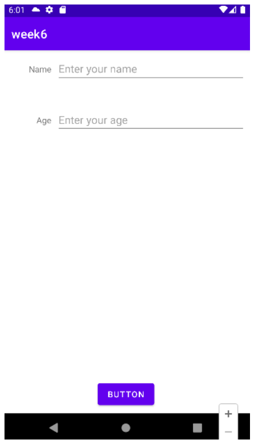
```
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val btnNewActivity = findViewById<Button>(R.id.buttonNewActivity)
        btnNewActivity.setOnClickListener { it: View!
            val editTextName = findViewById<EditText>(R.id.editTextName)
            val editTextAge = findViewById<EditText>(R.id.editTextAge)

            val name = editTextName.text.toString()
            val age = editTextAge.text.toString().toInt()

            Toast.makeText(applicationContext, text: "${name} ${age}", Toast.LENGTH_SHORT).show()
        }
    }

}
```

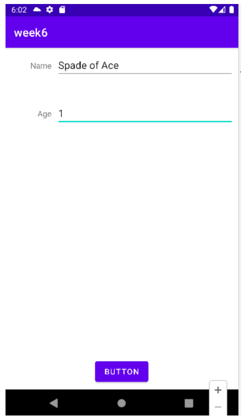


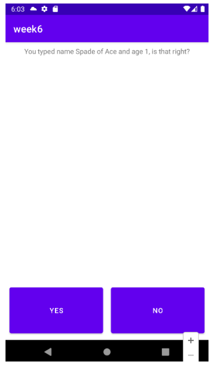
## [Lab-Practice #6] Hello!

- Now, you must do:
  - Change layout of the first activity
  - Add second activity to verify user input:

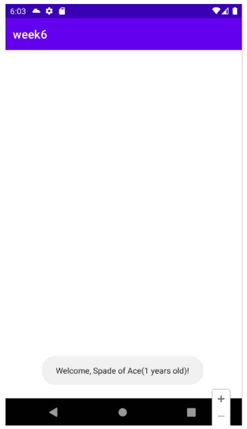
    You typed name <name> and age <age>, is that right?
  - Toast on the last activity: Welcome, <name>(<age> years old)!
  - **Clear two EditText when "No" pressed**







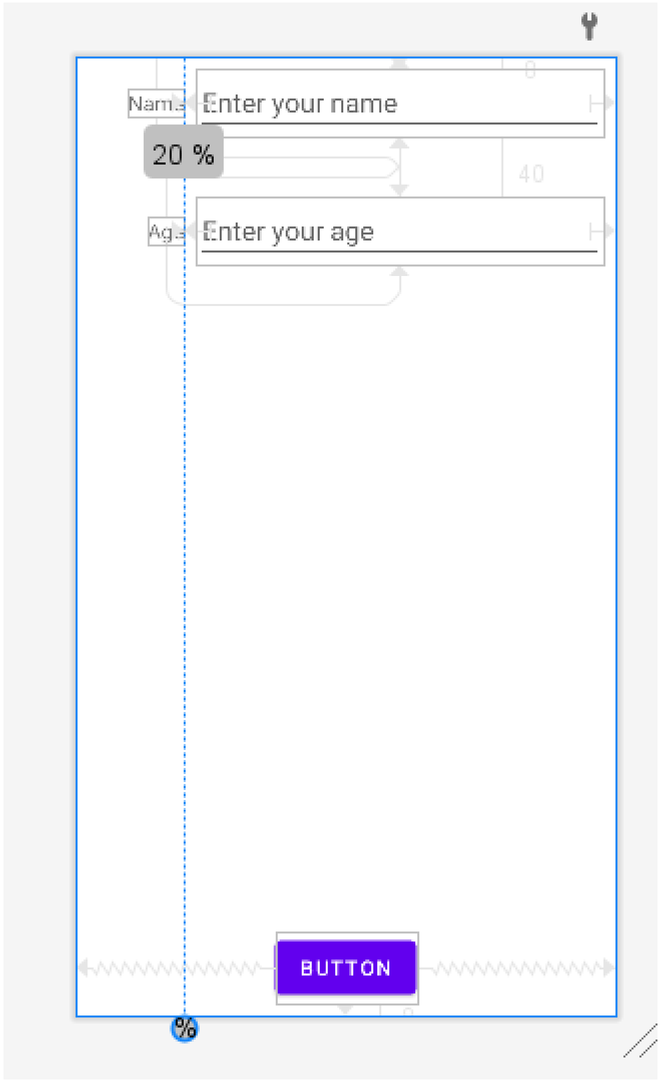
You typed name Spade of Ace and age 1, is that right?



Welcome, Spade of Ace(1 years old)!

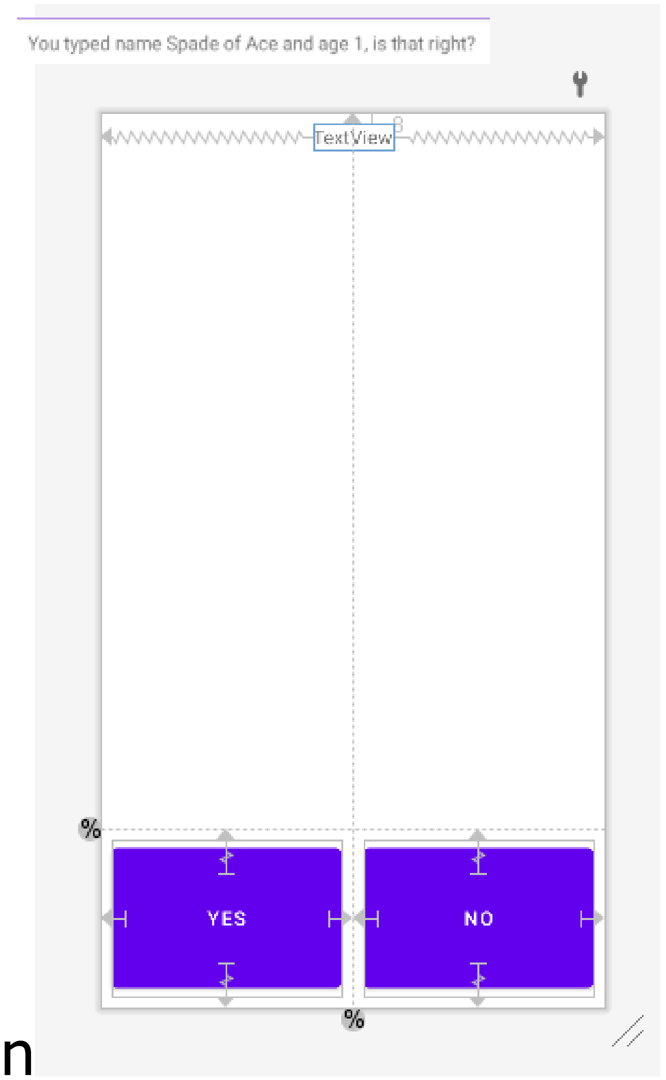
# [Lab-Practice #6] Hello!

- Now, you must do
  - Modify layout of first activity
    - Add textview to show what is the edittext
    - Same vertical center with the right edittext
    - Right aligned to 20% of screen
  - Modify EditText location
    - Fill right 80% portion of screen
    - with **8dp margin each**



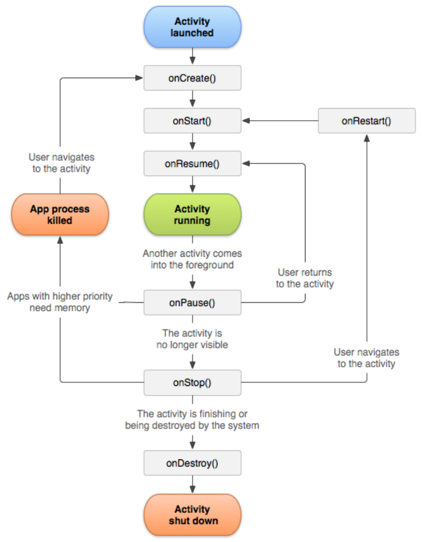
# [Lab-Practice #6] Hello!

- Now, you must do
  - Add verifying activity
    - Two button, yes and no
    - If press yes, go to the last activity that show toast
    - If press no, go back to first activity with clear editText
  - Button is scaled
    - 20% height of below screen
    - 50% width
    - a bit of margin and fill its portion

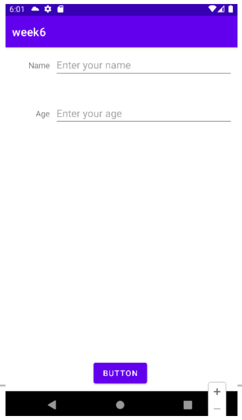


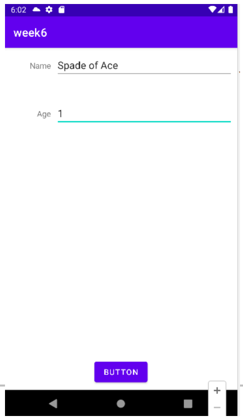
You typed name Spade of Ace and age 1, is that right?

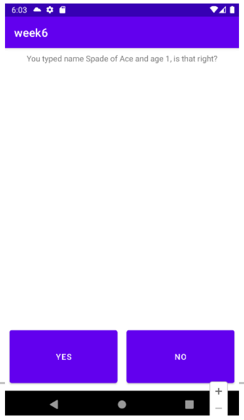
# [Lab-Practice #6] Hello!


Activity launched
onCreate()
onStart()
onRestart()
User navigates to the activity
onResume()
App process killed
Activity running
Another activity comes into the foreground
User returns to the activity
Apps with higher priority need memory
onPause()
The activity is no longer visible
User navigates to the activity
onStop()
The activity is finishing or being destroyed by the system
onDestroy()
Activity shut down

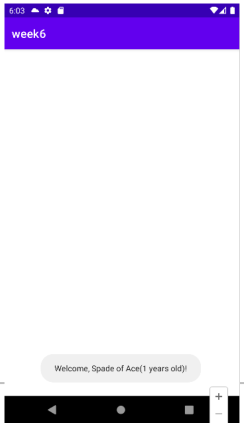
- Hint
  - finish() to kill activity
  - Text clear on Some function on activity lifecycle
    - edittext.text.clear() when activity re-loaded
    - Do NOT forget super call
  - Use guideline, and see layout on previous slides
  - Last activity is almost same with exercise ☺







You typed name Spade of Ace and age 1, is that right?



Welcome, Spade of Ace(1 years old)!

# [Lab-Practice #6] Hello!

- Criteria
  - Set up layout of first, second activity
    - Scaling with the size of the screen
  - Name and age data must pass until last(third) activity
  - Toast on last activity
  - **Clearing edittext after pressing NO**
    - **It is OK to clear on other cases.**
  - Execution
    - **Write name and age then press button: Check second activity opens and textview shows well**
    - **Press yes: Check third activity opens and toast shows well**
    - **Press back button: Check goes to second activity**
    - **Press no: Check goes to first activity and edittext cleared**
    - **Press back button: Check application closed**