



UPN Veteran Jawa Timur

# MODUL V

# PEMROGRAMAN MOBILE

## Sistem Informasi



2  
0  
2  
5

**Disusun Oleh**

Abdul Rezha Efrat Najaf, S.Kom.,M.Kom  
Lintang Firdaus  
Taufiq Arinta A

# PENGANTAR



Modul ini merupakan panduan praktis yang disusun untuk memperkenalkan konsep dasar dalam pemrograman mobile kepada para pembaca. Dengan berkembangnya teknologi, kebutuhan akan aplikasi mobile semakin meningkat, dan pemahaman akan teknik-teknik pengembangan aplikasi menjadi kunci dalam menciptakan solusi yang inovatif dan berdaya saing. Modul ini dirancang untuk memberikan pemahaman yang komprehensif tentang pemrograman mobile, mulai dari pemahaman dasar tentang platform mobile berbasis android hingga teknik-teknik pengembangan aplikasi yang efektif.

Penulis berharap pembaca dapat memanfaatkan modul ini sebagai sumber referensi yang berguna dalam perjalanan mereka dalam memahami dan menguasai pemrograman mobile. Terima kasih kepada semua pihak yang telah berkontribusi dalam penyusunan modul ini. Semoga modul ini dapat memberikan kontribusi positif dalam pengembangan keterampilan dan pengetahuan pembaca dalam bidang pemrograman mobile

**Semangat Selalu!** 😊



Berikut penjelasan fungsi dari tiap method di atas:

- `onCreate()` Di-method ini Activity sudah dimulai tapi belum terlihat oleh pengguna. Inisialisasi sebagian besar dimulai di sini. Misalnya memanggil `setContentView()` untuk membaca layout, membaca View, dll.
- `onStart()` Activity sudah terlihat tapi belum bisa berinteraksi. Method ini jarang dipakai, tapi bisa sangat berguna untuk mendaftarkan sebuah `BroadcastReceiver` untuk mengamati perubahan yang dapat mempengaruhi UI.
- `onResume()` Activity sudah terlihat dan pengguna sudah dapat berinteraksi. Di sini adalah tempat terbaik untuk menjalankan animasi, membuka akses seperti camera, mengupdate UI, dll.
- `onPause()` Kebalikan dari `onResume()`. Activity sudah akan bersiap-siap meninggalkan layar (masih terlihat) dan sudah tidak berinteraksi dengan pengguna. Biasanya bila perlu melakukan undo untuk pekerjaan yang dilakukan di `onResume()` kita lakukan di sini.
- `onStop()` Kebalikan dari `onStart()` Activity sudah tidak terlihat. Biasanya kita melakukan undo untuk pekerjaan yang dilakukan di dalam `onStart()`.
- `onDestroy()` Kebalikan dari `onCreate()`. Method ini dapat terpanggil karena memanggil method `finish()` atau karena sistem membutuhkan memori lebih. Di dalam `onDestroy()` kita biasanya membersihkan proses-proses yang ada di belakang layar. Misalnya pengunduhan data dari internet yang mungkin masih berjalan jika tidak dihentikan di `onDestroy()`.

- `onRestart()` Dipanggil saat activity sudah melalui `onStop()` tapi akan diaktifkan lagi. Method ini jarang di implementasi.

## II. INTENT EXPLICIT & IMPLICIT

Intent adalah sebuah objek messaging yang digunakan untuk melakukan permintaan sebuah aksi atau berkomunikasi antar app component seperti activity, service, dan broadcast.

Intent dalam Android terbagi menjadi dua jenis, yaitu: Intent Eksplisit Intent ini digunakan untuk berkomunikasi atau menjalankan komponen dalam satu aplikasi dengan menentukan nama kelas secara spesifik. Biasanya, intent ini digunakan untuk membuka aktivitas atau komponen lain di dalam aplikasi yang sama. Intent Implisit Intent ini digunakan untuk menjalankan komponen dari aplikasi lain tanpa menyebutkan nama kelas secara langsung. Contohnya, membuka kamera, mengakses browser, atau membagikan konten ke aplikasi lain.

Implementasi Intent pada project Android Studio

- a. Buatlah project baru pada android studio terlebih dahulu.
- b. Pada `activity_main.xml` tambahkan kode berikut ini :

```
<Button
    android:id="@+id/btn_move_activity"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="72dp"
    android:text="@string/move_activity"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.526"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

```

<Button
    android:id="@+id/btn_move_browser"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="40dp"
    android:text="@string/move_browser"

    app:layout_constraintEnd_toEndOf="@+id/btn_move_acti
vity"
    app:layout_constraintHorizontal_bias="0.0"

    app:layout_constraintStart_toStartOf="@+id/btn_move_a
ctivity"

    app:layout_constraintTop_toBottomOf="@+id/btn_move_
activity" />

```

Pada kodingan di atas kita akan membuat 2 buah button yang nantinya akan digunakan untuk implementasi intent.

- c. Buatlah activity baru dengan contoh activity bernama AnotherActivity. Kemudian pada activity\_another.xml tambahkan kode berikut ini :

```

<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/berhasil_pindah_activity"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"

```

```
app:layout_constraintStart_toStartOf="parent"  
app:layout_constraintTop_toTopOf="parent" />
```

- d. Kemudian pada MainActivity.kt tambahkan kode berikut ini

```
val btnMoveActivity: Button =  
findViewById(R.id.btn_move_activity)  
val btnMoveBrowser: Button =  
findViewById(R.id.btn_move_browser)  
  
btnMoveActivity.setOnClickListener{  
    val intent = Intent(this, AnotherActivity::class.java)  
    startActivity(intent)  
}  
  
btnMoveBrowser.setOnClickListener{  
    val intent = Intent(Intent.ACTION_VIEW,  
Uri.parse("https://developer.android.com"));  
    startActivity(intent)  
}
```

Kode di atas merupakan implementasi intent, intent explicit pada button move activity (untuk berpindah activity) dan intent implicit pada button move browser (untuk membuka alamat url di browser contoh di atas menggunakan url <https://developer.android.com>)

- e. Jalankan project tersebut, maka hasilnya akan seperti berikut ini :



PINDAH ACTIVITY

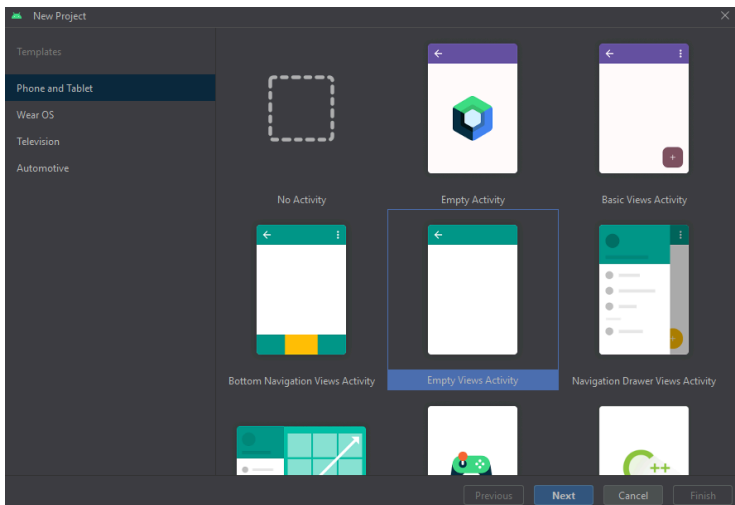
OPEN BROWSER

Berhasil pindah activity

## Praktikum

Wah tidak terasa sudah dipenghujung modul nih teman-teman! saatnya kita praktikum ya!

Pertama-tama buka android studio kalian dan buat project menggunakan layout



Nah setelah itu untuk nama project kalian bisa mengetikkan "MyIntentApp"

Jika sudah selanjutnya kita akan Kondisikan activity\_main.xml menjadi seperti ini.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

    <Button
        android:id="@+id/btn_move_activity"
        android:layout_width="match_parent"
```

```

        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:text="@string/move_activity" />
</LinearLayout>

```

Tambahkan juga resource string-nya. Tambahkan semua string yang akan digunakan di project ini. Buka berkas strings.xml dan tambahkan kode berikut ini.

```

<resources>
    <string name="app_name">MyIntentApp</string>
    <string name="move_activity">Pindah
Activity</string>
    <string name="move_with_result">Pindah Activity
untuk Result</string>
    <string name="result_from_activity">Hasil dari
Activity</string>
    <string name="choose_number">Pilih Tim Sepak Bola
Kesukaanmu</string>
    <string name="Barcelona">Barcelona</string>
    <string name="Arsenal">Arsenal</string>
    <string name="AC_Milan">AC Milan</string>
    <string name="Napoli">Napoli</string>
    <string name="choose">Pilih</string>
    <string name="this_is_moveactivity">Ini
MoveActivity</string>
</resources>

```

Kemudian masuk ke MainActivity, tambahkan beberapa baris kode yang berfungsi untuk memperkenalkan button yang sudah ditambahkan di layout seperti ini.

```

package com.example.myintentapp

import android.os.Bundle
import android.widget.Button
import androidx.appcompat.app.AppCompatActivity

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

```

```

        val btnMoveActivity: Button =
            findViewById(R.id.btn_move_activity)
    }
}

```

Lalu, tambahkan beberapa baris yang berfungsi untuk menambahkan event **onClick** pada button **btnMoveActivity** seperti ini.

```

package com.example.myintentapp

import android.os.Bundle
import android.view.View
import android.widget.Button
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity(),
View.OnClickListener {

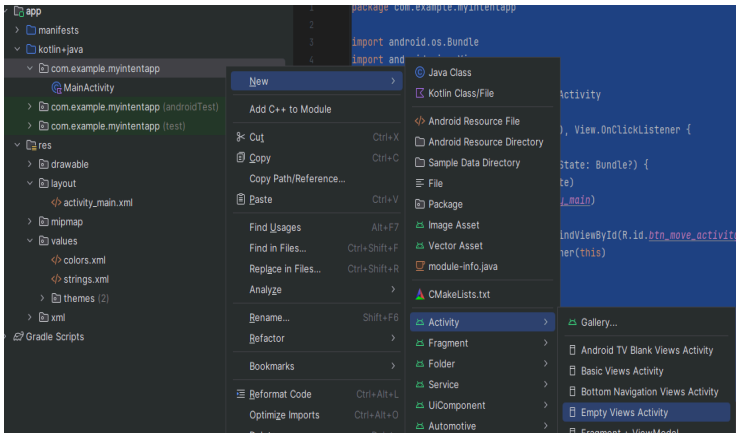
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val btnMoveActivity: Button =
            findViewById(R.id.btn_move_activity)
        btnMoveActivity.setOnClickListener(this)
    }

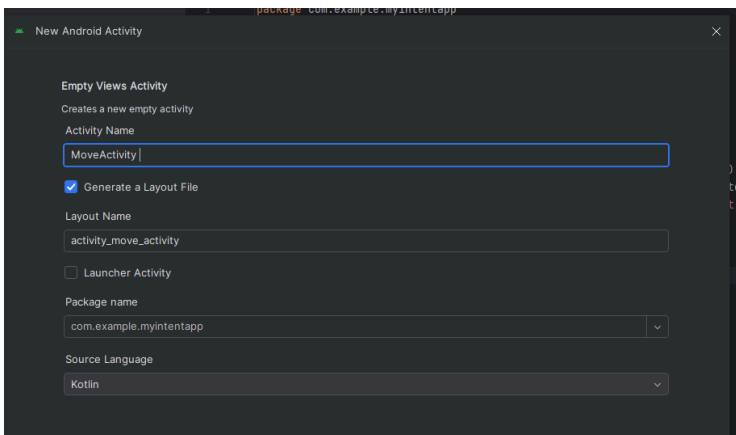
    override fun onClick(v: View?) {
        when (v?.id) {
            R.id.btn_move_activity -> {
            }
        }
    }
}

```

Button **btnMoveActivity** akan memiliki fungsi untuk berpindah Activity ke Activity lain. Sekarang kita buat Activity baru dengan cara sebagai berikut: klik kanan di package utama aplikasi package name → New → Activity → Empty Views Activity.



Isikan nama Activity dengan **MoveActivity** pada dialog. Ketika sudah klik Finish.



Untuk menandakan bahwa perpindahan Activity berhasil, silakan tambahkan satu **TextView** dan kondisikan activity\_move.xml menjadi seperti berikut.

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:padding="16dp"
tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/this_is_moveactivity" />
</RelativeLayout>

```

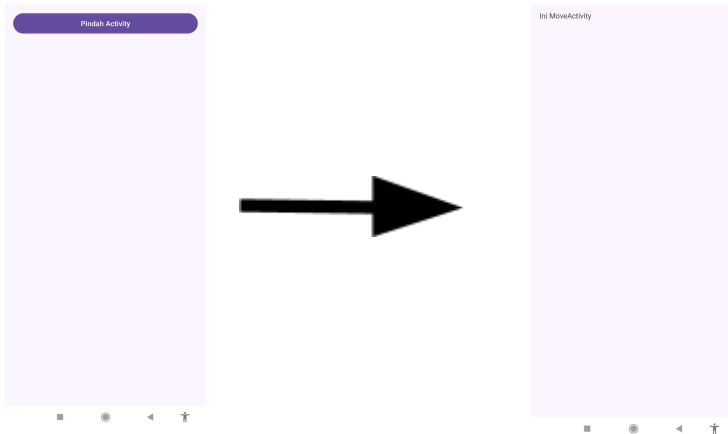
Setelah activity tujuan sudah berhasil diciptakan, sekarang saatnya menambahkan suatu intent pada method **onClick()** di MainActivity menjadi sebagai berikut.

```

override fun onClick(v: View?) {
    when (v?.id) {
        R.id.btn_move_activity -> {
            val moveIntent = Intent(this@MainActivity,
MoveActivity::class.java)
            startActivity(moveIntent)
        }
    }
}

```

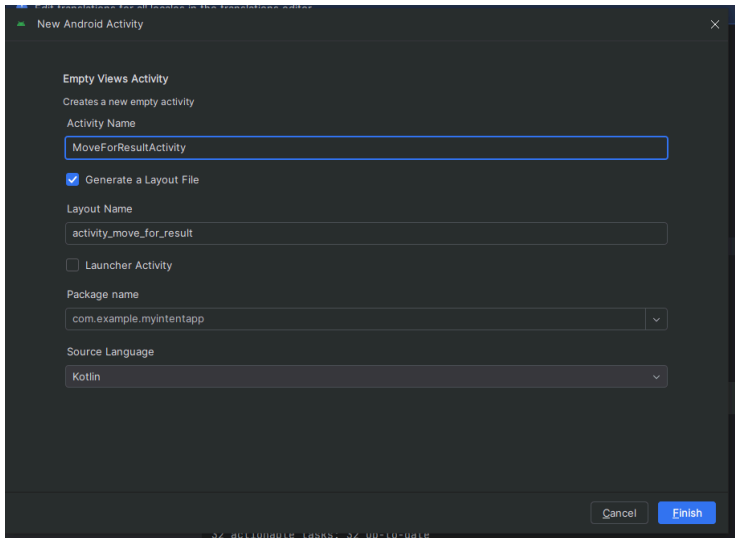
Selesai! Langkah pertama untuk memindahkan satu activity ke activity lain sudah selesai, sekarang silakan jalankan aplikasi Anda dengan mengklik tombol pada menu bar. Seharusnya sekarang Anda sudah bisa memindahkan activity dengan mengklik tombol Pindah Activity.



Sekarang kita akan membedah hubungan Activity dan Intent dalam menerima nilai balik. Kadang kala ketika menjalankan sebuah Activity dari Activity lain, kita mengharapkan ada nilai hasil balik dari Activity yang dijalankan ketika ia ditutup.

Contohnya kita memiliki Activity A yang menjalankan Activity B untuk melakukan sebuah proses. Lalu, nilai hasil dari proses tersebut dikirimkan kembali ke Activity A sebelum Activity B ditutup dengan pemanggilan metode `finish()`. Itulah yang dinamakan sebuah Activity menerima nilai hasil balik dari Activity yang dia jalankan. Supaya lebih jelas, ayo kita lanjutkan lagi proses ngoding-nya.

Sekarang buat lagi sebuah Empty Views Activity baru dengan cara yang sama seperti cara di atas tadi namun dengan nama yang berbeda yaitu **MoveForResultActivity**.



Kemudian pada `activity_move_for_result.xml`, silakan kondisikan layout dengan `LinearLayout`. Di dalam layout tersebut terdapat satu `TextView`, satu `RadioGroup`, empat `RadioButton`, dan satu `Button`.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:text="@string/choose_number" />

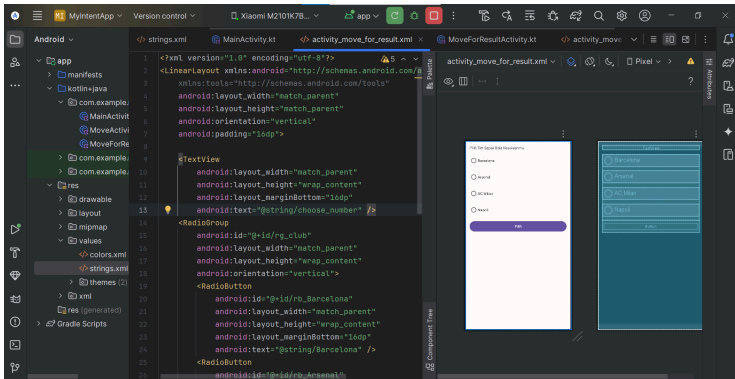
    <RadioGroup
        android:id="@+id/rg_club"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```

android:orientation="vertical">
  <RadioButton
    android:id="@+id/rb_Barcelona"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:text="@string/Barcelona" />
  <RadioButton
    android:id="@+id/rb_Arsenal"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:text="@string/Arsenal" />
  <RadioButton
    android:id="@+id/rb_Ac_Milan"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:text="@string/AC_Milan" />
  <RadioButton
    android:id="@+id/rb_Napoli"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:text="@string/Napoli" />
</RadioGroup>
<Button
  android:id="@+id/btn_choose"
  android:layout_width="match_parent"
  android:layout_height="wrap_content"
  android:text="@string/choose" />
</LinearLayout>

```

Tampilannya akan jadi seperti ini.



Setelah selesai, lanjutkan ke berkas MoveForResultActivity dengan mengenalkan komponen yang sudah ditambahkan.

```
package com.example.myintentapp

import android.os.Bundle
import android.widget.Button
import android.widget.RadioGroup
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat

class MoveForResultActivity : AppCompatActivity() {
    private lateinit var btnChoose: Button
    private lateinit var rgClub: RadioGroup
    companion object {
        const val EXTRA_SELECTED_VALUE =
"extra_selected_value"
        const val RESULT_CODE = 110
    }

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

```

        btnChoose = findViewById(R.id.btn_choose)
        rgClub= findViewById(R.id.rg_club)
    }
}

```

Kemudian beri aksi pada Button dengan menuliskan kode berikut.

```

package com.example.myintentapp

import android.content.Intent
import android.os.Bundle
import android.view.View
import android.widget.Button
import android.widget.RadioGroup
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat

class MoveForResultActivity : AppCompatActivity(),
View.OnClickListener {
    private lateinit var btnChoose: Button
    private lateinit var rgClub: RadioGroup
    companion object {
        const val EXTRA_SELECTED_VALUE =
"extra_selected_value"
        const val RESULT_CODE = 110
    }

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

        btnChoose = findViewById(R.id.btn_choose)
        rgClub = findViewById(R.id.rg_club)

        btnChoose.setOnClickListener(this)
    }

    override fun onClick(v: View?) {
        if (v?.id == R.id.btn_choose) {
            if (rgClub.checkedRadioButtonId > 0) {
                var value = ""
                when (rgClub.checkedRadioButtonId) {

```



```

        android:layout_height="match_parent"
        android:orientation="vertical"
        android:padding="16dp">

        <Button
            android:id="@+id/btn_move_activity"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginBottom="16dp"
            android:text="@string/move_activity" />

        <Button
            android:id="@+id/btn_move_for_result"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginBottom="16dp"
            android:text="@string/move_with_result" />

        <TextView
            android:id="@+id/tv_result"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:text="@string/result_from_activity"
            android:textSize="24sp" />

</LinearLayout>

```

Setelah selesai, kita lanjut menginisialisasi TextView dan Button ke dalam berkas MainActivity.

```

class MainActivity : AppCompatActivity(),
View.OnClickListener {

    private lateinit var tvResult: TextView

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

        ...

        val btnMoveForResult: Button =
findViewById(R.id.btn_move_for_result)

```

```

        btnMoveForResult.setOnClickListener(this)
        tvResult = findViewById(R.id.tv_result)
    }
}

```

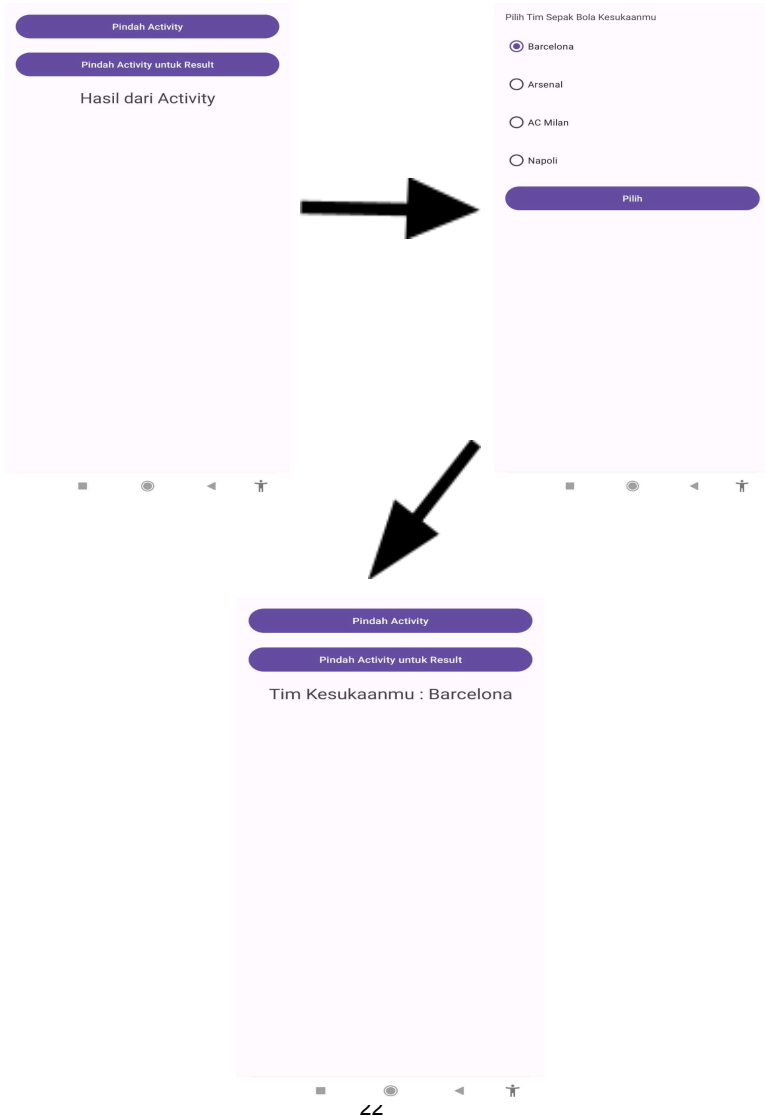
Selanjutnya, buat launcher `registerForResult` untuk menerima nilai balik. Alih-alih menggunakan `startActivity`, gunakan launcher tersebut untuk menjalankan Intent seperti berikut.

```

class MainActivity : AppCompatActivity(),
View.OnClickListener {
    private lateinit var tvResult: TextView
    private val resultLauncher =
registerForResult(
    ActivityResultContracts.StartActivityForResult()
) { result ->
    if (result.resultCode ==
MoveForResultActivity.RESULT_CODE && result.data !=
null) {
        val selectedValue =
result.data?.getStringExtra(MoveForResultActivity.EXTRA
_SELECTED_VALUE) ?: "Timnas Indonesia"
tvResult.text = "Tim Kesukaanmu : $selectedValue"
    }
}
...
override fun onClick(v: View?) {
    when (v?.id) {
        ...
        R.id.btn_move_for_result -> {
            val moveForResultIntent =
Intent(this@MainActivity,
MoveForResultActivity::class.java)
            resultLauncher.launch(moveForResultIntent)
        }
    }
}
}

```

Setelah selesai, pastikan tidak ada bagian kode yang tertinggal. Jalankan kembali aplikasinya dan coba pilih tombol Pindah Activity untuk Result. Pilih tim sepakbola yang Anda sukai dan lihat hasilnya. Di MainActivity sudah ada tim sepakbola yang tadi Anda pilih di objek TextView.



## **Tugas**

Setelah kalian mencoba praktikum di atas tugas kalian adalah untuk mengimplementasikan praktikum di atas dan juga menambahkan gambar pada radio button setiap masing-masing klub pada pilihan radio button yang ada sebelumnya. Selamat mencoba ^^