

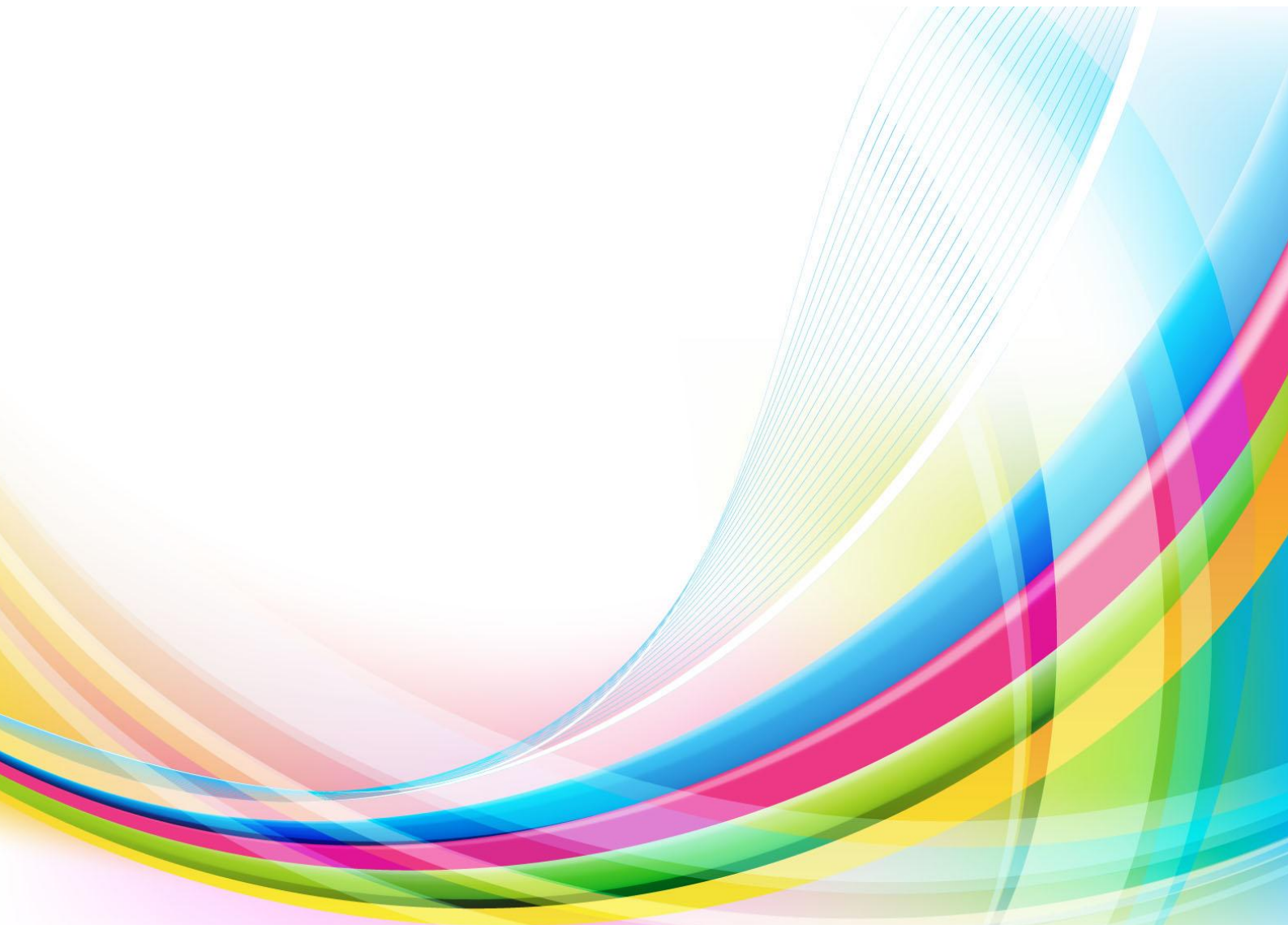


MODUL PRAKTIKUM 3

INTENT AND ACTIVITY

BAHASA PEMROGRAMAN II S1 SI

STEVI EMA WIJAYANTI



INTENT AND ACTIVITY IN ANDROID STUDIO

Overview :

Pada modul ini akan dibahas mengenai Activity dan Intent serta implementasinya dalam Android Studio. Intent dan activity merupakan komponen utama yang sangat penting dan vital. Dua komponen tersebut merupakan elemen basic yang ada di semua aplikasi Android.

Tujuan dari praktikum 3 :

1. Mengetahui definisi activity dan siklus hidupnya (*life cycle*)
2. Mampu mendesain UI (*User Interface*) dan mengimplementasikan pada Android Studio
3. Mengetahui definisi Intent dan macam-macam intent
4. Mengetahui implementasi intent pada Android Studio

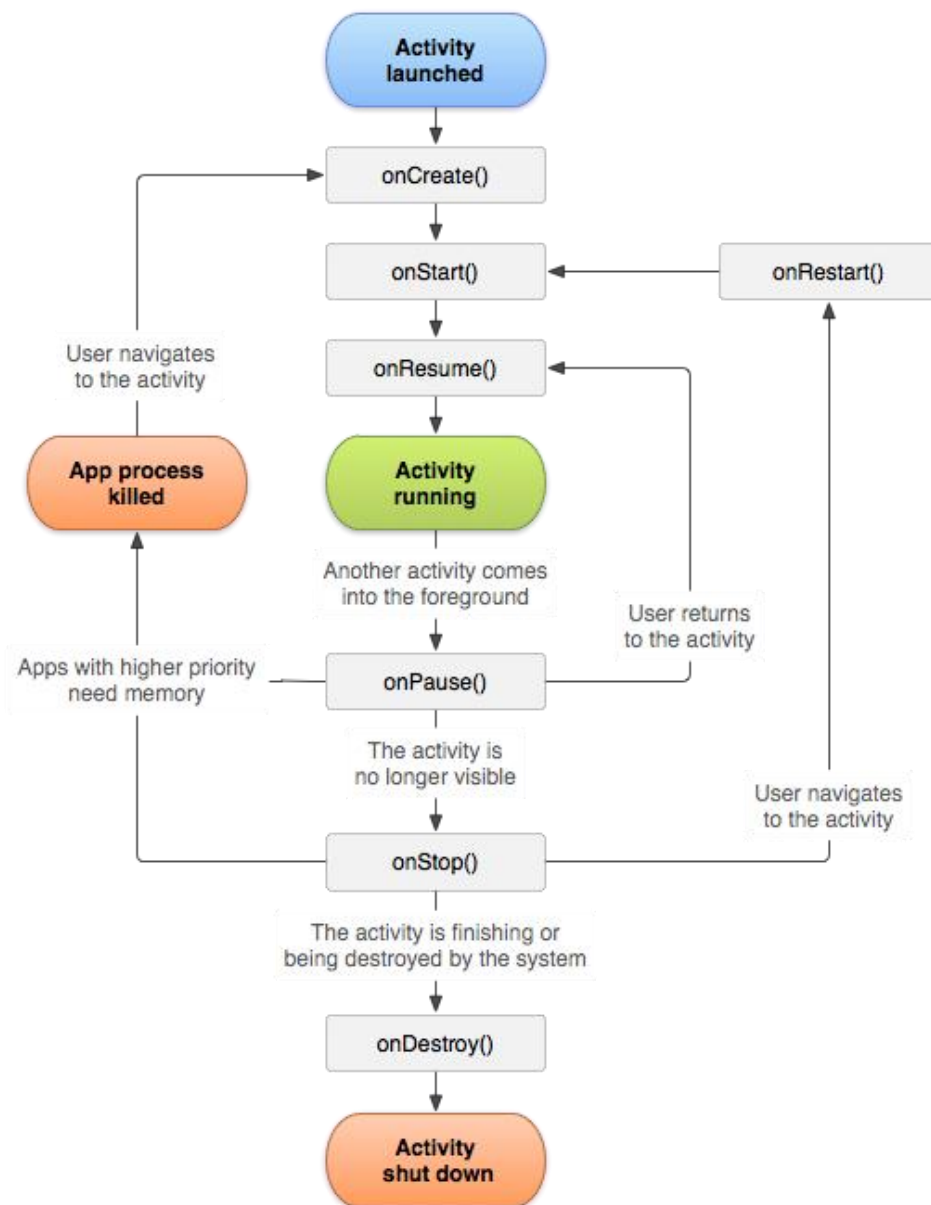
ACTIVITY (AKTIVITAS)

Activity adalah sebuah komponen aplikasi Android yang menampilkan dan mengatur halaman aplikasi sebagai tempat interaksi antara pengguna dengan aplikasi, seperti untuk membuat panggilan telp, mengambil foto, mengirim email, mengirim pesan singkat, bahkan menampilkan peta (maps). Sebuah *activity* mengatur satu halaman UI aplikasi, sehingga apabila sebuah aplikasi android memiliki beberapa halaman UI yang saling berinteraksi, berarti aplikasi tersebut memiliki beberapa *activity* yang saling berinteraksi juga.

Activity merupakan salah satu blok fundamental untuk membangun aplikasi berbasis android, karena menyediakan layanan agar pengguna dapat berinteraksi dengan aplikasi. Pada sebuah aplikasi biasanya ditetapkan satu aktivitas utama, yang ditampilkan kepada pengguna saat membuka aplikasi android untuk pertama kali. Aktivitas utama tersebut akan memanggil aktivitas lain, tiap kali aktivitas baru dimulai, aktivitas sebelumnya akan dihentikan, namun sistem tetap mempertahankan aktivitas sebelumnya dalam sebuah tumpukan (*back-stack*). Bila aktivitas baru dimulai, aktivitas sebelumnya akan didorong ke atas *back-stack* dengan mekanisme LIFO (*Last In First Out*). Sehingga apabila pengguna telah selesai dengan aktivitas yang baru kemudian menekan tombol kembali, aktivitas sebelum aktivitas yang baru akan dikeluarkan dari tumpukan *back-stack* dan dimusnahkan.

Apabila aktivitas lama dihentikan karena dimulainya aktivitas yang baru, maka aktivitas yang lama akan diberi informasi mengenai perubahan status melalui metode *callback* daur hidupnya. Ada beberapa metode *callback* yang mungkin diterima oleh *activity*, apakah sistem sedang membuatnya, menghentikannya, melanjutkannya atau menghapuskannya, dan masing-masing *callback* memberi

anda kesempatan melakukan pekerjaan tertentu yang sesuai untuk perubahan status tersebut. Jika ada ingin mengakses sebuah informasi dari server dan koneksi jaringan/database terputus, maka aktivitas secara otomatis akan dihentikan dan akan dilanjutkan kembali setelah mendapatkan sumber data yang diperlukan untuk melanjutkan tindakan yang terputus. Transisi ini semua adalah bagian dari daur hidup aktivitas. Siklus daur hidup aktivitas secara detail dapat anda lihat pada gambar berikut ini :



Gambar 1. Daur hidup aktivitas (*activity life cycle*)

Untuk membuat sebuah aktivitas, anda harus membuat subkelas Activity. Dalam subkelas tersebut, anda perlu mengimplementasikan metode-metode *callback* yang dipanggil sistem saat aktivitas bertransisi di antara berbagai status daur hidupnya, misalkan saat aktivitas sedang dibuat,

dihentikan, dilanjutkan atau dimusnahkan. Mengelola daur hidup aktivitas dengan mengimplementasikan metode-metode *callback* sangat penting untuk mengembangkan aplikasi yang kuat dan fleksibel. Pada dasarnya, sebuah aktivitas bisa berada dalam tiga status antara lain :

Dilanjutkan

Aktivitas berada di latar depan layar dan mendapatkan fokus pengguna dalam artian aktivitas sedang berjalan (*running*)

Dihentikan sementara

Aktivitas lain berada di latar depan dan mendapat fokus, namun aktivitas ini masih terlihat, artinya aktivitas lain terlihat diatas aplikasi dan aktivitas itu setengah transparan/tidak menutupi seluruh layar.

Dihentikan

Aktivitas ditutupi sepenuhnya oleh aktivitas lain (aktivitas sekarang berada di latar belakang).

Contoh implementasinya adalah seperti berikut ini :

```
public class ExampleActivity extends Activity {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        // The activity is being created.
    }
    @Override
    protected void onStart() {
        super.onStart();
        // The activity is about to become visible.
    }
    @Override
    protected void onResume() {
        super.onResume();
        // The activity has become visible (it is now "resumed").
    }
    @Override
    protected void onPause() {
        super.onPause();
        // Another activity is taking focus (this activity is about to
        be "paused").
    }
    @Override
    protected void onStop() {
        super.onStop();
        // The activity is no longer visible (it is now "stopped")
    }
}
```

```

@Override
protected void onDestroy() {
    super.onDestroy();
    // The activity is about to be destroyed.
}
}

```

INTENT

Intent adalah sebuah class dalam pemrograman android yang berfungsi untuk perpindahan halaman. Intent juga merupakan sebuah objek yang terdapat dalam suatu aktivitas (*activity*) dimana objek tersebut dapat berkomunikasi dengan aktivitas yang lain. Intent dibagi menjadi dua yaitu :

Explicit Intent

Berfungsi untuk mengaktifkan komponen-komponen dalam satu aplikasi yang sama, misalkan berpindah aktivitas

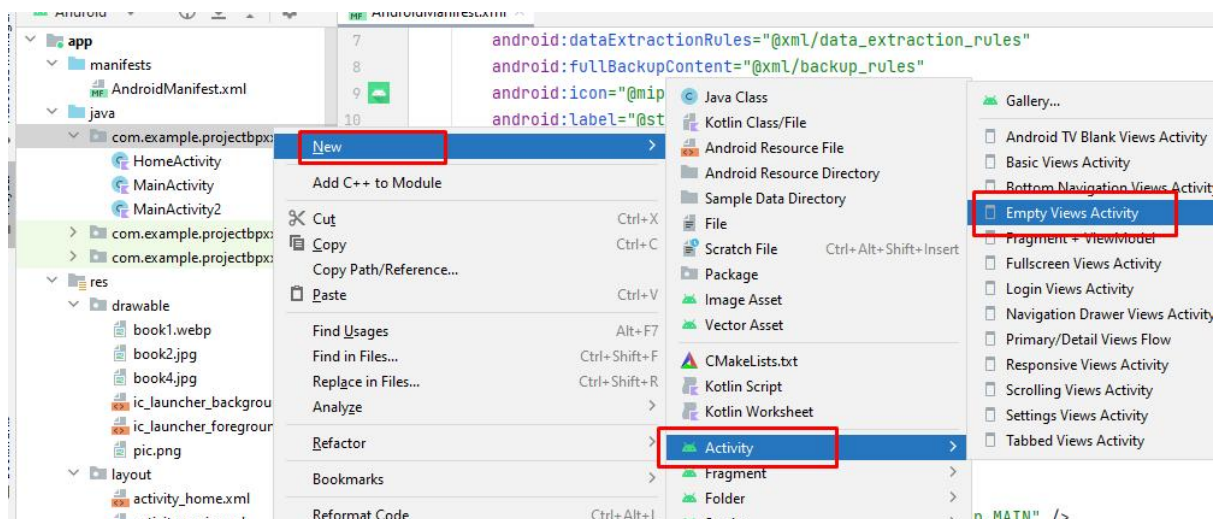
Implicit Intent

Berfungsi untuk memanggil fungsi aktivitas yang berada di fungsi internal android seperti dial number, open browser dan lain sebagainya

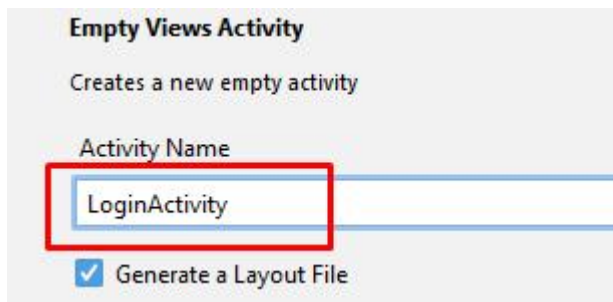
IMPLEMENTASI INTENT DAN ACTIVITY

Silahkan buka kembali project (ProjectBPIIXXXX) yang telah anda kerjakan pada praktikum sebelumnya. Jika sebelumnya anda telah menambahkan activity Home (Homepage) pada aplikasi anda, maka pada praktikum 3 akan dibahas terkait dengan intent baik implisit maupun eksplisit, langkah-langkahnya :

1. Pastikan anda telah membuka project yang telah anda buat pada praktikum 2. Jika sudah tambahkan activity baru pada project tersebut untuk halaman login ke aplikasi.



Berikan nama activity **LoginActivity**

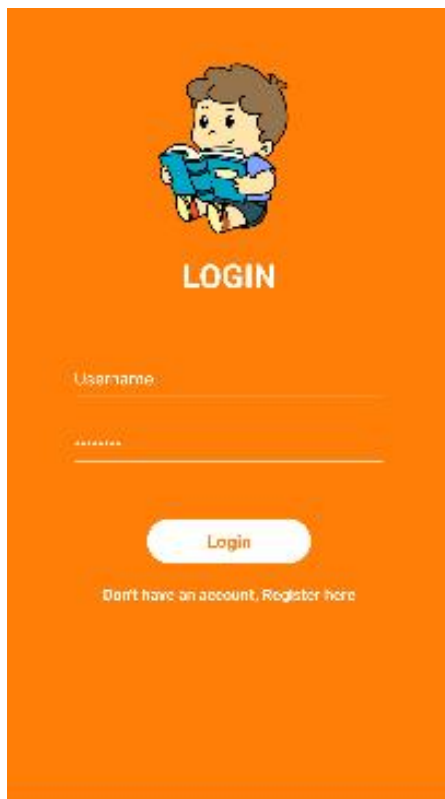


Empty Views Activity
Creates a new empty activity

Activity Name

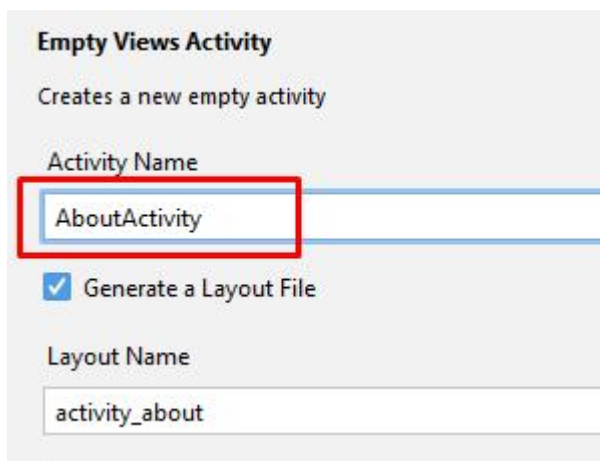
☒ Generate a Layout File

2. Silahkan buat desain UI (*user interface*) untuk halaman login, seperti contoh gambar di bawah ini :



Desain bebas, tidak harus sama persis dengan contoh gambar, yang penting terdapat inputan untuk username dan password serta button Login

3. Tambahkan activity baru lagi, kemudian berikan nama **AboutActivity**



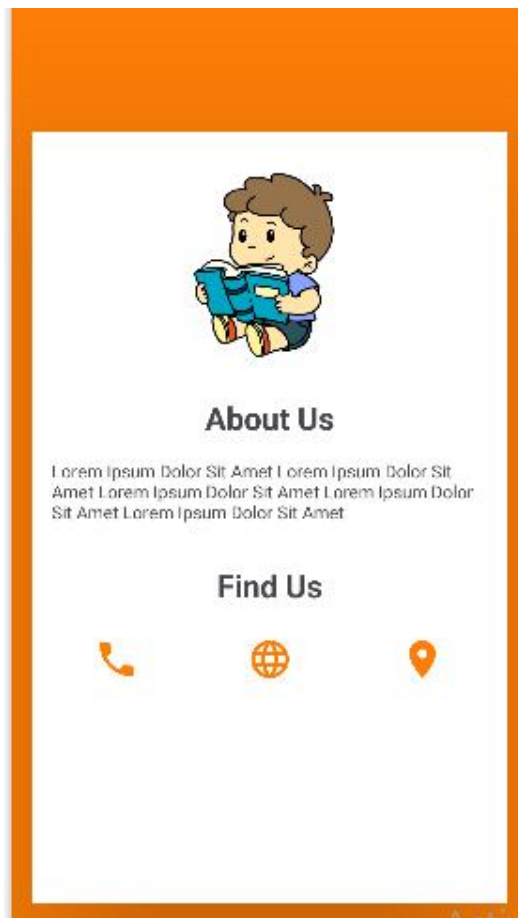
Empty Views Activity
Creates a new empty activity

Activity Name

☒ Generate a Layout File

Layout Name

4. Kemudian design halaman UI dari About Activity, seperti contoh di gambar di bawah ini :



Desain bebas, tidak harus sama persis dengan contoh gambar, yang penting terdapat icon akses telp, website dan maps

5. Buka kembali AndroidManifest dan setting agar halaman Login ditampilkan pertama kali saat programnya di-load.



6. Pada activity login, saat button Login di-klik akan ada proses pengecekan apakah username dan password yang diinputkan benar atau salah. Jika benar, maka akan masuk ke activity home (homepage), namun jika salah akan muncul notifikasi username dan password salah.

```

class LoginActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_login)

        //instance
        val btnLogin:Button = findViewById(R.id.buttonLogin)

        //event saat button Login click
        btnLogin.setOnClickListener{ it: View!
            //call homepage
            val intent = Intent( packageContext: this,HomeActivity::class.java)
            startActivity(intent)
        }
    }
}

```

Jalankan program anda, kemudian klik button login, pastikan homepage activity ditampilkan.

7. Pada HomePage terdapat button menu More, jika diklik akan masuk ke activity About, maka baris kodenya :

```

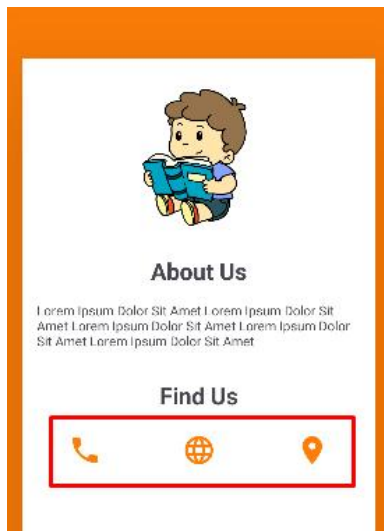
class HomeActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_home)

        //instance
        val btnMore:Button = findViewById(R.id.menu4)

        //event button more click
        btnMore.setOnClickListener { it: View!
            val intent = Intent( packageContext: this, AboutActivity::class.java)
            startActivity(intent)
        }
    }
}

```

8. Jika dua contoh source code diatas menerapkan explicit intent, maka source code berikut akan menerapkan implisit intent. Jika melihat pada About Activity terdapat 3 implicit intent untuk dial number, akses alamat website dengan menggunakan browser dan akses lokasi menggunakan map.



Buat instance untuk palette yang digunakan untuk mengakses implicit Intent

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

        //instance
        val call:ImageView = findViewById(R.id.imageViewCall)
        val web:ImageView = findViewById(R.id.imageViewWeb)
        val location:ImageView = findViewById(R.id.imageViewLocation)
    }
}
```

Dial number

```
//dial number
call.setOnClickListener { it: View!
    val callIntent: Intent= Uri.parse( uriString: "tel:085743439096").let {
        number-> Intent(Intent.ACTION_DIAL,number)
    }
    startActivity(callIntent)
}
```

Akses Website

```
//website
web.setOnClickListener { it: View!
    val webIntent:Intent = Uri.parse( uriString: "https://home.amikom.ac.id/").let{
        webpage->Intent(Intent.ACTION_VIEW,webpage)
    }
    startActivity(webIntent)
}
```

Akses Lokasi

```
//lokasi
location.setOnClickListener { it: View!
    val gmmIntentUri = Uri.parse( uriString: "geo:47.6, -122,3?z=11")
    val mapIntent = Intent(Intent.ACTION_VIEW, gmmIntentUri)
    mapIntent.setPackage("com.google.android.apps.maps")
    if (mapIntent.resolveActivity(packageManager) != null) {
        startActivity(mapIntent)
    }
}
```

9. Jalankan aplikasi android yang telah anda tambahkan baris kode tersebut.

TUGAS

Jika pada tugas sebelumnya, anda diminta untuk membuat homepage, pada pertemuan kali ini :

1. Buat desain activityLogin
2. Buat desain activity About dimana berisi tentang pengembang/tentang aplikasi, sertakan pula alamat website dan nomor HP dari pengembang atau perusahaan pengembang aplikasi. Boleh ditambahkan lokasi seperti contoh di modul.
3. Implementasikan implicit dan explicit intent untuk kedua activity yang anda buat.