

## Kuis Pemrograman Perangkat Mobile

Nama : Muhammad Tyas Firdaus

NIM : 11222101 (A2)

Jurusan : Teknik Informatika

### Soal!

Jawablah pertanyaan di bawah ini dengan uraian yang jelas dan terstruktur!

1. Apa itu Android Studio dan mengapa alat ini penting dalam pengembangan aplikasi Android? Jelaskan fitur-fitur utama yang dimiliki oleh Android Studio.
2. Dalam pengembangan aplikasi Android, apa itu `Activity` dan bagaimana cara kerjanya? Berikan contoh kasus penggunaan `Activity` dalam sebuah aplikasi sederhana.
3. Jelaskan apa itu `Intent` dalam Android dan bagaimana cara penggunaannya. Berikan contoh kode yang menunjukkan bagaimana mengirim data dari satu `Activity` ke `Activity` lain menggunakan `Intent`.
4. Apa yang dimaksud dengan `Fragment` dalam pengembangan aplikasi Android? Bagaimana `Fragment` berbeda dengan `Activity`? Berikan contoh penggunaan `Fragment` dalam sebuah aplikasi.
5. Apa itu Gradle dalam konteks pengembangan aplikasi Android? Jelaskan peran dan fungsi Gradle dalam proses build aplikasi.

### Jawab :

1. **Android Studio** adalah Integrated Development Environment (IDE) resmi yang disediakan oleh Google untuk pengembangan aplikasi Android. Dibangun di atas IntelliJ IDEA, Android Studio menawarkan berbagai alat dan fitur yang membantu pengembang dalam membuat, menguji, dan mengoptimalkan aplikasi Android mereka.

Fitur Utama Android Studio:

**Editor Kode yang Kuat:** Android Studio menyediakan editor kode yang lengkap dengan fitur seperti auto-completion, code refactoring, dan debugging.

**Emulator dan Perangkat Fisik:** Dapat menguji aplikasi pada emulator Android yang realistis atau langsung di perangkat fisik.

**Build Tools:** Android Studio menyertakan build tools terintegrasi untuk membangun dan mem-package aplikasi untuk distribusi.

**Dukungan Gradle:** Gradle adalah sistem build yang kuat yang membantu mengelola dependensi dan mengotomatiskan proses build.

**Layout Designer Visual:** Android Studio menyediakan layout designer visual untuk membangun antarmuka pengguna aplikasi dengan mudah.

**Pengujian Unit dan UI:** Android Studio memudahkan untuk menulis dan menjalankan pengujian unit dan UI untuk memastikan aplikasi berfungsi dengan baik.

**Lint Tools untuk Menemukan Bug:** Android Studio memiliki lint tools yang membantu menemukan dan memperbaiki bug dalam kode.

**Integrasi dengan Google Play Console:** Android Studio terintegrasi dengan Google Play Console, memungkinkan untuk mengelola dan mempublikasikan aplikasi di Google Play Store.

**Banyak Plugin dan Template:** Ada banyak plugin dan template yang tersedia untuk Android Studio, yang dapat membantu mempercepat pengembangan aplikasi.

2. **Activity** adalah komponen dasar aplikasi Android yang merepresentasikan satu layar dengan antarmuka pengguna. Setiap Activity biasanya berfokus pada satu tugas pengguna tertentu. Contohnya, sebuah aplikasi email mungkin memiliki satu Activity untuk melihat daftar email dan Activity lain untuk menulis email baru.

### **Cara Kerja Activity:**

- Setiap Activity didefinisikan oleh kelas Java atau Kotlin yang meng-extend `android.app.Activity`.
- Siklus hidup Activity dikelola oleh sistem Android, dimulai dari pembuatan (`onCreate()`), berjalan (`onStart()`), dan ditampilkan (`onResume()`), hingga dihentikan (`onPause()`, `onStop()`), dan dihancurkan (`onDestroy()`).

**Contoh Kasus Penggunaan Activity:** Misalnya, aplikasi sederhana yang memiliki dua Activity: satu untuk menampilkan daftar item dan satu lagi untuk menampilkan detail item yang dipilih.

```
java
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        // Kode untuk menampilkan daftar item
    }
}

public class DetailActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_detail);
        // Kode untuk menampilkan detail item
    }
}
```

3. **Intent** adalah mekanisme yang digunakan untuk memulai Activity, Service, atau mengirimkan pesan antar komponen dalam aplikasi Android. Intent bisa bersifat eksplisit (menyatakan komponen tujuan secara langsung) atau implisit (menyatakan tindakan yang ingin dilakukan, tanpa menentukan komponen tujuan).

Contoh kode berikut menunjukkan bagaimana mengirim data dari satu Activity ke Activity lain menggunakan Intent:

```
java
Copy code
// Mengirim data dari MainActivity ke DetailActivity
Intent intent = new Intent(MainActivity.this, DetailActivity.class);
intent.putExtra("EXTRA_DATA", "Some data to send");
startActivity(intent);

// Menerima data di DetailActivity
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_detail);

    Intent intent = getIntent();
    String data = intent.getStringExtra("EXTRA_DATA");
    // Gunakan data yang diterima
}
```

4. **Fragment** adalah komponen modular dari antarmuka pengguna yang bisa digunakan kembali, yang mengelola bagian dari UI atau perilaku aplikasi dalam sebuah Activity. Fragment memungkinkan aplikasi untuk memiliki lebih dari satu UI di layar yang sama, dan dapat dikelola secara independen.

#### Perbedaan Fragment dengan Activity:

- **Fragment:** Biasanya digunakan untuk membangun UI yang fleksibel dan dinamis di dalam satu Activity. Fragment harus selalu menjadi bagian dari Activity.
- **Activity:** Merupakan layar penuh dan berfungsi sebagai container untuk Fragment atau komponen lainnya.

**Contoh Penggunaan Fragment:** Misalnya, aplikasi yang menampilkan daftar artikel di satu fragment dan detail artikel di fragment lain.

```
java
Copy code
public class ListFragment extends Fragment {
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {
        return inflater.inflate(R.layout.fragment_list, container, false);
    }
}

public class DetailFragment extends Fragment {
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {
        return inflater.inflate(R.layout.fragment_detail, container,
false);
    }
}
```

5. **Gradle** adalah sistem build yang digunakan oleh Android Studio untuk mengotomatiskan proses build aplikasi. Gradle mengelola semua dependensi, konfigurasi build, dan tugas build, sehingga memudahkan pengembang untuk mengelola proyek mereka.

**Peran dan Fungsi Gradle:**

1. **Mengelola Dependensi:** Gradle memungkinkan pengembang untuk mendeklarasikan dependensi proyek mereka dalam file `build.gradle`, dan secara otomatis mengunduh serta mengelolanya.
2. **Konfigurasi Build:** Pengembang dapat mengonfigurasi berbagai aspek dari build, seperti varian build (debug vs release), sumber daya, dan pengaturan lain.
3. **Automasi Tugas:** Gradle memungkinkan pengembang untuk membuat tugas otomatis untuk menguji, membangun, dan mendistribusikan aplikasi mereka.