

UJIAN TENGAH SEMESTER PEMROGRAMAN MOBILE 1

Dosen Pengampu: Nova Agustina, S.T., M.Kom.



Disusun oleh:

23552011266 – Muhammad Ilham Abdul Halim

TIF RP 23 CID B

PROGRAM STUDI TEKNIK INFORMATIKA

UNIVERSITAS TEKNOLOGI BANDUNG

2025

Perhatikan setiap butir soal berikut dengan teliti. Pastikan Anda sudah memenuhi semua kriteria sebelum disubmit!

Essay

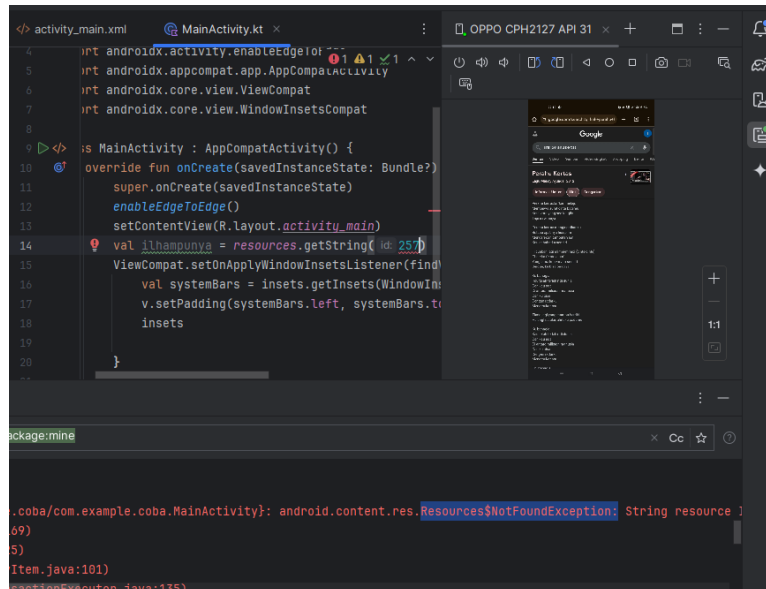
1. Apa fungsi findViewById?
2. Apa syarat pemanggilan method findViewById? Buat contohnya dan screenshot source code nya!
3. Error apa yang terjadi jika file kotlin salah menginisialisasi findViewById atau objek pada xml belum diinisialisasi?
4. Buat sebuah contoh program untuk menampilkan pesan error Resources.NotFoundException! Screenshot logcat-nya!
5. Kumpulkan dalam bentuk pdf di Elearning (Soal essay digabung dengan soal studi kasus cek point 7 Studi Kasus)

Studi Kasus

1. Buatlah sebuah program sederhana yang terdiri dari 4 Activity menggunakan Android Native (Java + XML) yang terdiri dari:
 - a. SplashScreen Activity
 - b. Login Activity
 - c. Register Activity
 - d. News Portal Dashboard
2. Ketentuan: Silahkan membuat splashscreen dengan baik.
3. Pada Register Activity, minimal terdapat objek: TextView, EditText, Button, ImageView!
4. Tampilkan event Log, Toast dan Toast pada saat Button Register di klik.
5. Pada News Portal Dashboard terdapat data yang ditampilkan dalam listview
6. Upload project di Github.
7. Jelaskan fungsi setiap baris source code pada file kotlin dan submit dalam bentuk pdf pada Elearning
8. Link Github harus tercantum pada pdf (point 7)

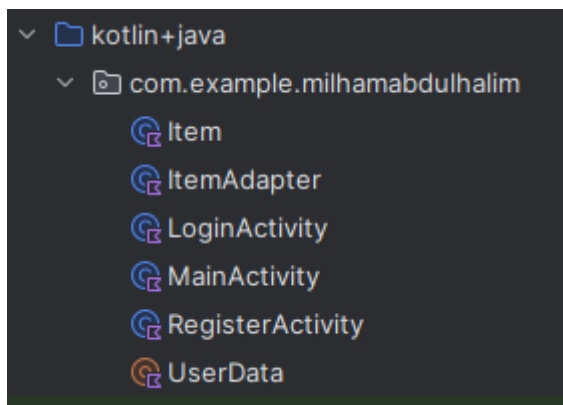
- **Jawaban**

1. Fungsi dari findViewById adalah untuk mengambil nilai dari menu layout yang ada di file XML
2. Untuk melakukan method ini, kita harus menginisialisasikan terlebih dahulu dalam file XML nya seperti android:id="@+id/misal"
3. Unsolved Reference, NullPointerException
4. Contoh program menampilkan pesan error Resources.NotFoundException:



- **Soal Studi Kasus:**

1. Daftar activity yang saya buat:



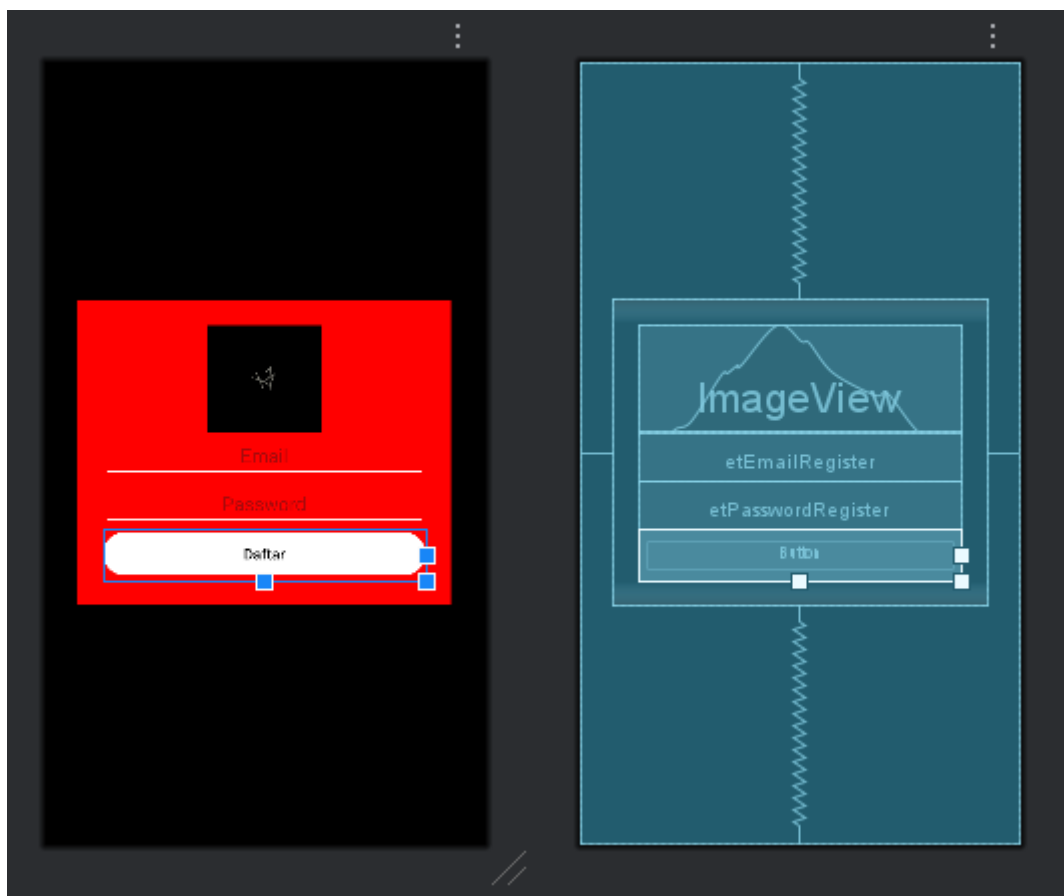
2. Bagian splash screen, sebagai berikut

```

1  <resources xmlns:tools="http://schemas.android.com/tools">
2      <!-- Base application theme. -->
3      <style name="Base.Theme.MilhamAbdulHalim" parent="Theme.Material3.DayNight.NoActionBar">
4          |
5      </style>
6
7      <style name="Theme.MilhamAbdulHalim" parent="Base.Theme.MilhamAbdulHalim" />
8      <style name="Theme.App.SplashScreen" parent="Theme.SplashScreen">
9          <item name="windowSplashScreenBackground">@color/white</item>
10         <item name="windowSplashScreenAnimatedIcon">@drawable/g1</item>
11         <item name="postSplashScreenTheme">@style/Base.Theme.MilhamAbdulHalim</item>
12     </style>
13 </resources>

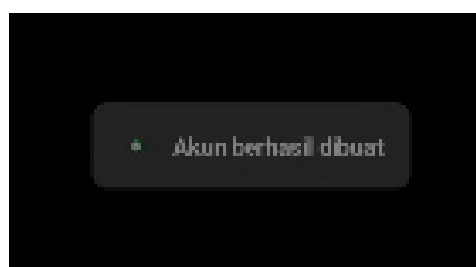
```

3. Layout pada activity_register:

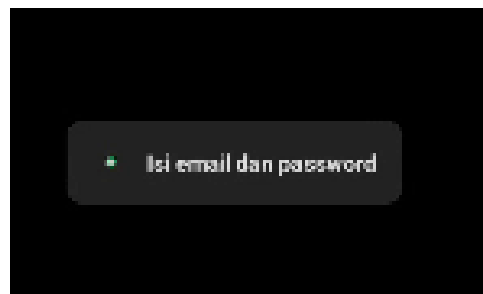


4. Toast yang muncul ketika button register diklik:

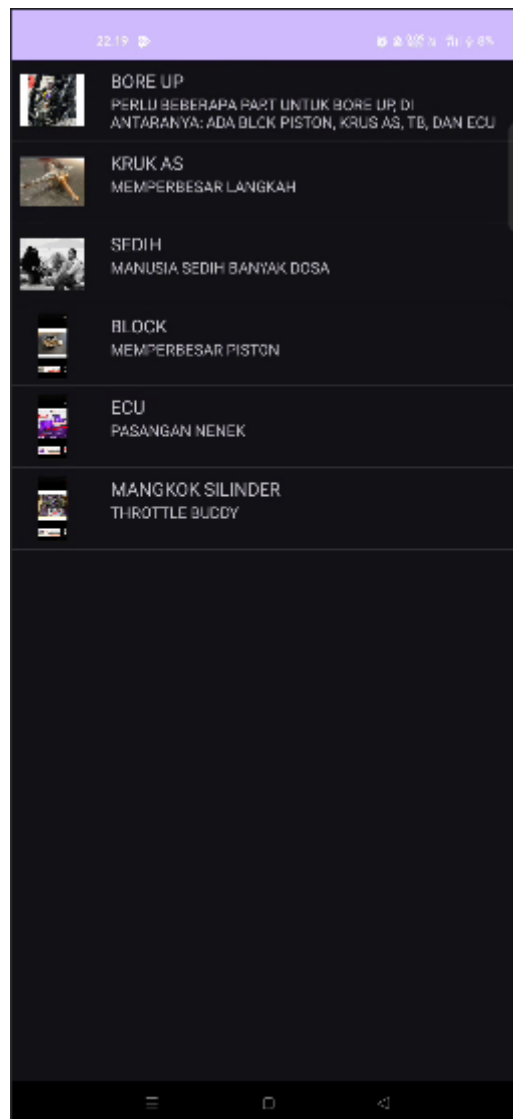
- Ketika diisi dengan benar:



- Ketika tidak diisi:



5. News Portal Dashboard yang berupa ListView:



6. Link Github: https://github.com/Illham-halim/UTS_Pemrograman-mobile-1

7. Penjelasan Coding Kotlin, sebagai berikut:

- Item.kt

```
data class Item(val judul: String, val isi: String, val imageResId: Int)
```

Deklarasi data-data yang akan digunakan dalam proyek ini, ditulis dengan tipe datanya

- ItemAdapter.kt

```
import android.content.Context
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.*
import androidx.core.content.ContextCompat
```

Mengimpor berbagai kelas yang diperlukan: untuk layout, tampilan (ImageView, TextView), dan utilitas kompatibilitas (ContextCompat).

```
class ItemAdapter(private val context: Context, private val itemList: List<Item>) :
    ArrayAdapter<Item>(context, 0, itemList) {
```

- ItemAdapter adalah class turunan dari ArrayAdapter<Item>, digunakan untuk mengisi tampilan dari ListView atau Spinner dengan data dari list.
- Menerima:
- context: informasi lingkungan (biasanya activity)
- itemList: daftar Item yang akan ditampilkan.

0 di parameter ArrayAdapter berarti tidak memakai layout bawaan (karena kita inflate manual dalam getView()).

```
override fun getView(position: Int, convertView: View?, parent: ViewGroup): View {
```

- Fungsi ini **dipanggil setiap kali ListView perlu menampilkan satu item**.
- position: posisi item dalam list.
- convertView: view daur ulang (agar tidak inflate baru terus-menerus → lebih efisien).
- parent: parent view (biasanya ListView itu sendiri).

```
val view = convertView ?:
    LayoutInflater.from(context).inflate(R.layout.activity_item_adapter, parent, false)
```

- Jika convertView null, maka inflate layout activity_item_adapter.xml.
- Ini adalah layout XML yang kamu buat untuk setiap item (harus berisi itemImage, itemName, itemNameee).

```
val item = itemList[position]
val imageView = view.findViewById<ImageView>(R.id.itemImage)
val textVieww = view.findViewById<TextView>(R.id.itemName)
val textView = view.findViewById<TextView>(R.id.itemNameee)
```

- ☐ Ambil data dari itemList berdasarkan posisi.
- ☐ Temukan komponen dari layout XML (image dan 2 text).
- imageView.setImageDrawable(ContextCompat.getDrawable(context, item.imageResId))
- textVieww.text = item.judul
- textView.text = item.isi
- ☐ Atur gambar berdasarkan resource ID (item.imageResId).

- Set teks dari item.judul dan item.isi.
- LoginActivity.kt

```
import android.content.Intent
import android.os.Bundle
import android.widget.*
import androidx.appcompat.app.AppCompatActivity
import androidx.core.splashscreen.SplashScreen.Companion.installSplashScreen
```

Lakukan impor pada library yang dibutuhkan

```
class LoginActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
```

Membuka login activity

```
installSplashScreen()
setContentView(R.layout.activity_login)
```

Membuka splash screen

```
val emailInput = findViewById<EditText>(R.id.etEmail)
val passwordInput = findViewById<EditText>(R.id.etPassword)
val btnMasuk = findViewById<Button>(R.id.btnMasuk)
val btnDaftar = findViewById<Button>(R.id.btnDaftar)
```

Deklarasi variable yang akan digunakan

```
btnMasuk.setOnClickListener {
    val email = emailInput.text.toString()
    val password = passwordInput.text.toString()

    if (UserData.userList.contains(Pair(email, password))) {
        startActivity(Intent(this, MainActivity::class.java))
    } else {
        Toast.makeText(this, "Email atau password salah", Toast.LENGTH_SHORT).show()
    }
}
```

Aksi bila tombol masuk ditekan. Bila sesuai akan masuk ke main activity dan bila tidak akan muncul suatu toast pemberitahuan

- MainActivity.kt
- import android.os.Bundle
- import android.widget.ListView
- import androidx.appcompat.app.AppCompatActivity

Pengimporan library yang akan digunakan

```
private val items = listOf(
    Item("BORE UP", "PERLU BEBERAPA PART UNTUK BORE UP, DI ANTARANYA: ADA BLOK PISTON, KRUS AS, TB, DAN ECU", R.drawable.g1),
    Item("KRUK AS", "MEMPERBESAR LANGKAH ", R.drawable.g2),
    Item("SEDIH", "MANUSIA SEDIH BANYAK DOSA", R.drawable.g3),
    Item("BLOCK", "MEMPERBESAR PISTON", R.drawable.g4),
    Item("ECU", "PASANGAN NENEK", R.drawable.g5),
```

```
Item("MANGKOK SILINDER", "THROTTLE BUDDY", R.drawable.g6)  
)
```

Pengisian list item yang akan ditampilkan di list view

```
override fun onCreate(savedInstanceState: Bundle?) {  
    super.onCreate(savedInstanceState)  
    setContentView(R.layout.activity_main)  
  
    val listView = findViewById<ListView>(R.id.listView)  
    val adapter = ItemAdapter(this, items)  
    listView.adapter = adapter  
}
```

Membuka layout activity main dan pendeklarasian variabel item

- RegisterActivity.kt
- ```
import android.content.Intent
import android.os.Bundle
import android.widget.*
import androidx.appcompat.app.AppCompatActivity
```

Impor library yang akan digunakan

```
val emailInput = findViewById<EditText>(R.id.etEmailRegister)
val passwordInput = findViewById<EditText>(R.id.etPasswordRegister)
val btnDaftarAkun = findViewById<Button>(R.id.btnDaftarAkun)
```

Deklarasi variabel yang akan digunakan

```
btnDaftarAkun.setOnClickListener {
 val email = emailInput.text.toString()
 val password = passwordInput.text.toString()

 if (email.isNotEmpty() && password.isNotEmpty()) {
 UserData.userList.add(Pair(email, password))
 Toast.makeText(this, "Akun berhasil dibuat", Toast.LENGTH_SHORT).show()
 startActivity(Intent(this, LoginActivity::class.java))
 finish()
 } else {
 Toast.makeText(this, "Isi email dan password", Toast.LENGTH_SHORT).show()
 }
}
```

Aksi bila button daftar akun ditekan. Bila diisi maka akan pindah ke login activity, bila tidak maka akan muncul toast

- UserData.kt
- ```
object UserData {  
    val userList = arrayListOf<Pair<String, String>>() // Pair(email, password)  
}
```

Deklarasi untuk menyimpan data pengguna yang telah kita register di awal