

UTS
PEMROGRAMAN MOBILE 1

Disusun Untuk Memenuhi Tugas Kuliah
Mata Kuliah: Pemrograman Mobile 1

Dosen Mata Kuliah:
Ibu Nova



Disusun Oleh:
Mochammad Andwi Haikal
(23552011011)

Kelas: CID 23-PB

PROGRAM STUDI TEKNIK INFORMATIKA
UNIVERSITAS TEKNOLOGI BANDUNG
TAHUN
2025

LINK GITHUB :

https://github.com/andwihaikal/UTS_PM1_MochammadAndwi_23552011011.git

Essay

1. Apa fungsi `setOnClickListener`?
2. Apa syarat pemanggilan method `setOnClickListener`? Buat contohnya dan screenshot source code nya!
3. Error apa yang terjadi jika file kotlin salah menginisialisasi `findViewById` atau objek pada xml belum diinisialisasi? Screenshot logcat-nya!
4. Buat sebuah contoh program untuk menampilkan pesan error `NullPointerException`!
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 List Chating
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 List Chating 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)**

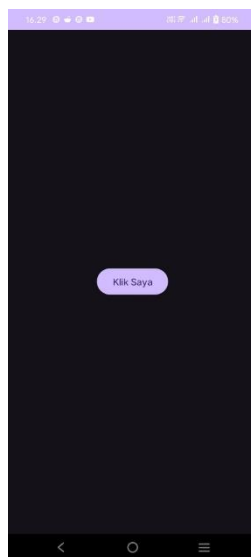
JAWABAN

1. `setOnClickListener` digunakan untuk menangani event klik pada sebuah View (misalnya: Button, ImageView, dll) di Android. Fungsi ini menentukan tindakan apa yang dilakukan ketika elemen tersebut diklik oleh pengguna.

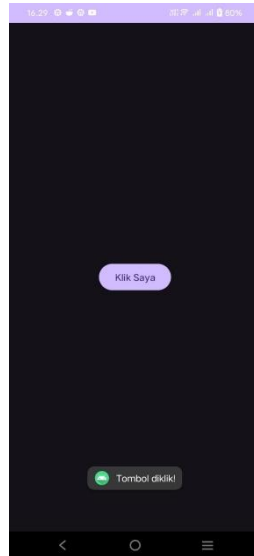
2. Syarat:

- View yang ingin diberi listener (misalnya Button) **sudah diinisialisasi** dengan `findViewById`.
- Pemanggilan `setOnClickListener` dilakukan **setelah view ditemukan**.

Hasil



Jika set On Click di klik



Maka akan muncul notifikasi Tombol diklik

Source Code :

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical">
    <Button
        android:id="@+id/myButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Klik Saya" />
</LinearLayout>
```

MainActivity.kt

```

1 // MainActivity.kt
2 package com.example.onclicklistenerdemo
3
4 import android.os.Bundle
5 import android.widget.Button
6 import android.widget.Toast
7 import androidx.appcompat.app.AppCompatActivity
8
9 class MainActivity : AppCompatActivity() {
10     override fun onCreate(savedInstanceState: Bundle?) {
11         super.onCreate(savedInstanceState)
12         setContentView(R.layout.activity_main)
13
14         val myButton = findViewById<Button>(R.id.myButton)
15
16         myButton.setOnClickListener {
17             Toast.makeText(this, "Tombol diklik!", Toast.LENGTH_SHORT).show()
18         }
19     }
20 }
21

```

3. Jika findViewById dipanggil pada ID yang tidak ada, atau objeknya belum didefinisikan di layout XML, maka akan terjadi:

Error: NullPointerException

4. Buat sebuah contoh program untuk menampilkan pesan error NullPointerException!

11.07 ● ○ ⊕ •

📶 📶 📶 📶 📶 83%

NullPointerException

Terjadi NullPointerException: null



Source Code :

```
1 package com.example.nullpointerexampleactivity
2
3 import android.os.Bundle
4 import android.util.Log
5 import android.widget.TextView
6 import androidx.appcompat.app.AppCompatActivity
7
8 class MainActivity : AppCompatActivity() {
9     override fun onCreate(savedInstanceState: Bundle?) {
10         super.onCreate(savedInstanceState)
11         setContentView(R.layout.activity_main)
12
13         val myTextView: TextView? = null
14         val errorTextView: TextView = findViewById(R.id.errorTextView)
15
16         try {
17             myTextView!!.text = "Ini akan error!"
18         } catch (e: NullPointerException) {
19             val errorMessage = "Terjadi NullPointerException: ${e.message}"
20             Log.e(tag: "NullPointerException", errorMessage)
21             errorTextView.text = errorMessage // Menampilkan error di layar
22         }
23     }
24 }
25
```

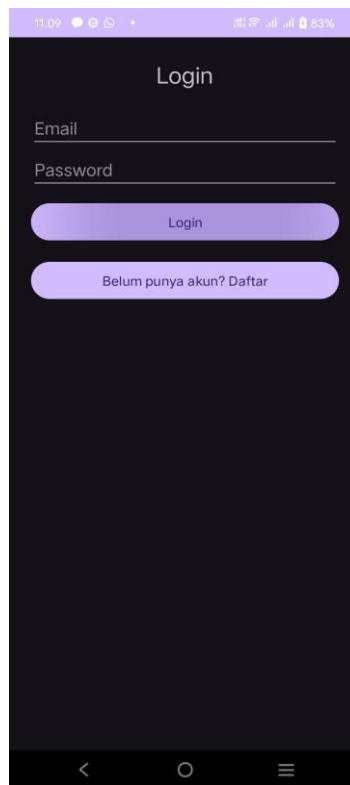
```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:orientation="vertical"
4     android:padding="24dp"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent">
7
8     <TextView
9         android:text="NullPointerException"
10        android:textSize="18sp"
11        android:layout_width="wrap_content"
12        android:layout_height="wrap_content" />
13
14     <TextView
15        android:id="@+id/errorTextView"
16        android:text=""
17        android:textColor="#FF0000"
18        android:layout_marginTop="20dp"
19        android:layout_width="wrap_content"
20        android:layout_height="wrap_content" />
21 </LinearLayout>
22
```

STUDI KASUS

a SplashScreen Activity



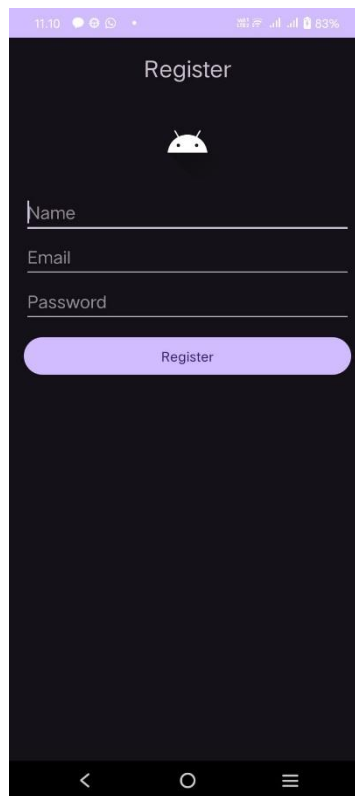
b. Login Activity



Jika Login Ditekan

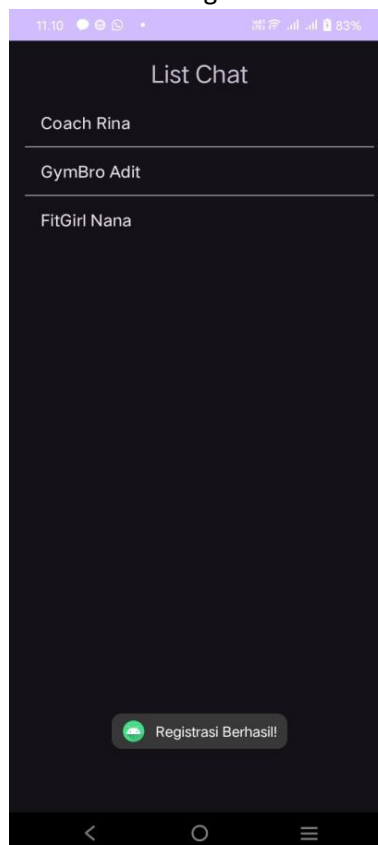


c. Register Activity



A screenshot of an Android application's 'Register' screen. The screen has a dark blue background. At the top, the status bar shows the time 11:10, signal strength, and 83% battery. The title 'Register' is centered at the top. Below it is a white Android robot icon. There are three text input fields labeled 'Name', 'Email', and 'Password'. Below these fields is a red rounded rectangular button labeled 'Register'. At the bottom, there is a navigation bar with a back arrow, a home circle, and a menu icon.

Jika tombol Register Ditekan

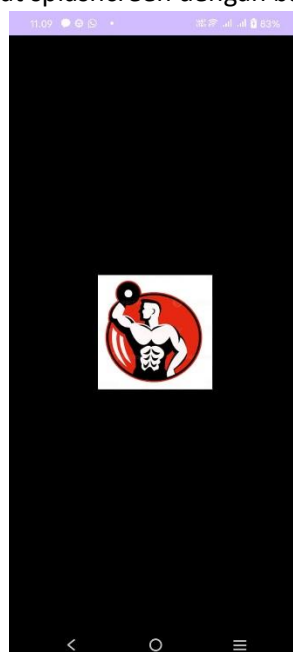


A screenshot of an Android application's 'List Chat' screen. The screen has a dark blue background. At the top, the status bar shows the time 11:10, signal strength, and 83% battery. The title 'List Chat' is centered at the top. Below it are three chat entries, each with a name and a horizontal line below it: 'Coach Rina', 'GymBro Adit', and 'FitGirl Nana'. At the bottom, there is a green rounded rectangular button with a white checkmark icon and the text 'Registrasi Berhasil!'. At the very bottom, there is a navigation bar with a back arrow, a home circle, and a menu icon.

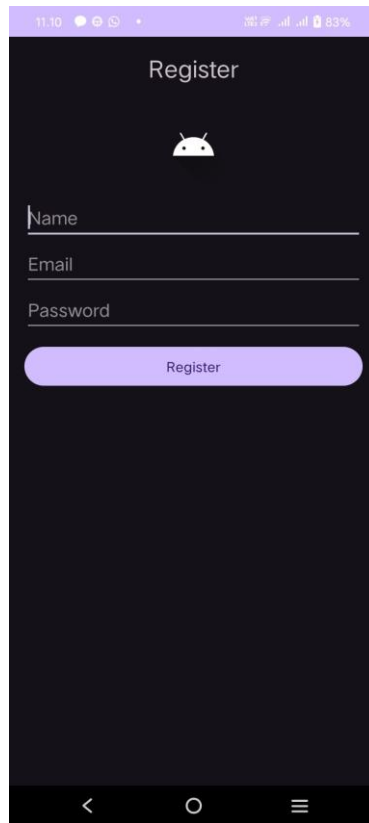
d. List Chating



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.
Toast pada saat tombol register ditekan



Toast pada saat tombol login ditekan

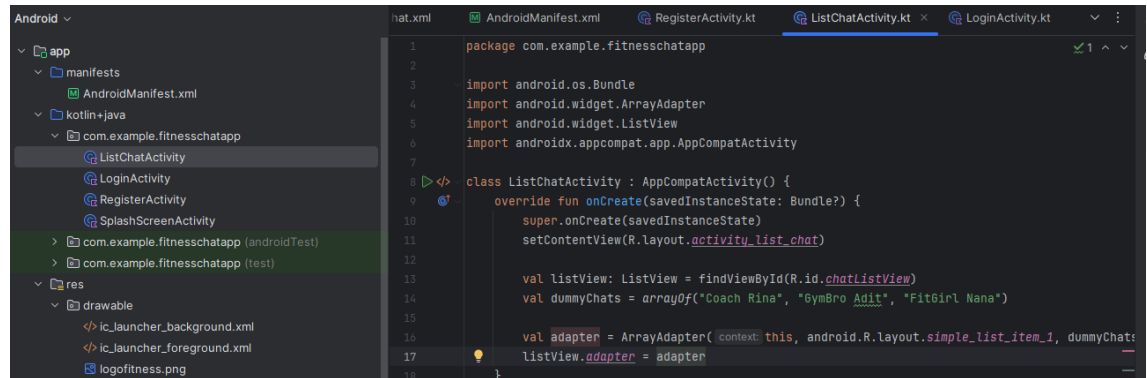


5. Pada List Chating terdapat data yang ditampilkan dalam listview



6. Jelaskan fungsi setiap baris source code pada file kotlin dan submit dalam bentuk pdf pada Elearning

ListChatActivity.kt



class ListChatActivity : AppCompatActivity() {

Membuat activity baru bernama ListChatActivity.

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity_list_chat)

Menjalankan saat activity dibuat, dan menampilkan layout.

val listView: ListView = findViewById(R.id.chatListView)

Menghubungkan ListView dari XML ke Kotlin.

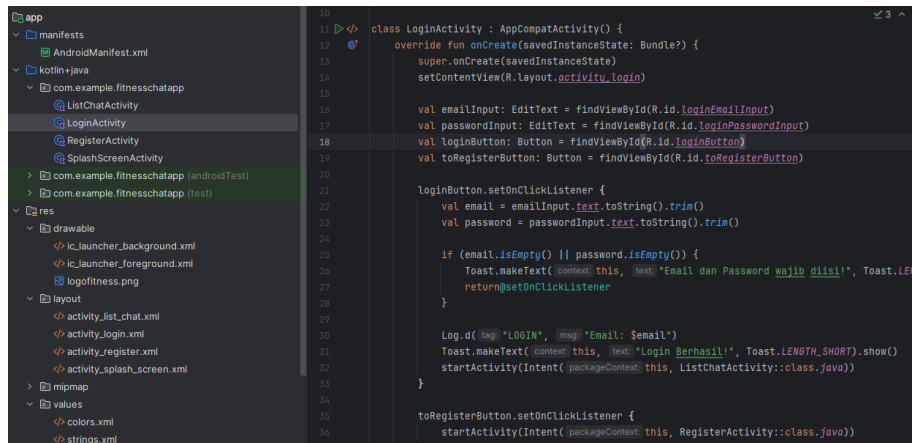
val dummyChats = arrayOf("Coach Rina", "GymBro Adit", "FitGirl Nana")

Membuat data chat dummy.

val adapter = ArrayAdapter<Context, android.R.layout.simple_list_item_1, String>(this, android.R.layout.simple_list_item_1, dummyChats)

Membuat adapter untuk menampilkan array ke ListView

Login Activity



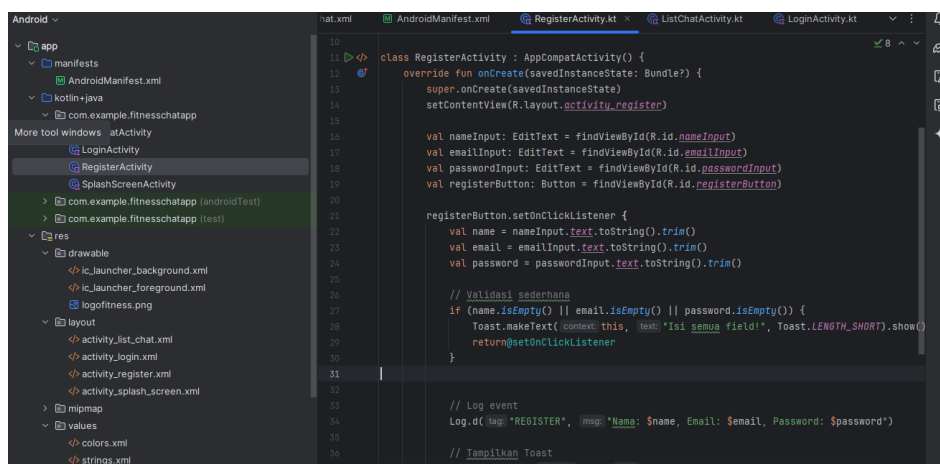
setContentView(...): Menampilkan layout activity_login

findViewById(...): Menghubungkan UI (EditText & Button) ke kode.

loginButton.setOnClickListener:

- Validasi input email dan password.
- Tampilkan toast jika kosong.
- Jika valid, tampilkan toast "Login Berhasil" & pindah ke ListChatActivity.
- **toRegisterButton.setOnClickListener:**
 - Pindah ke RegisterActivity.

RegisterActivity.kt



setContentView(...): Menampilkan layout activity_register.

findViewById(...): Menghubungkan EditText & Button ke kode.

registerButton.setOnClickListener:

- Validasi input (nama, email, password).
- Tampilkan toast jika ada yang kosong.
- Jika valid:
 - Cetak ke Log.
 - Tampilkan toast "Registrasi Berhasil".
 - Pindah ke ListChatActivity.

SplashScreenActivity.kt

```
class SplashScreenActivity : AppCompatActivity() {  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.activity_splash_screen)  
  
        // Delay 2 detik lalu pindah ke LoginActivity  
        Handler(Looper.getMainLooper()).postDelayed({  
            startActivity(Intent(packageContext, LoginActivity::class.java))  
            finish()  
        }, delayMillis = 2000)  
    }  
}
```

setContentView(...): Menampilkan layout splash screen.

Handler(...).postDelayed(...):

- Menunggu 2 detik.
- Lalu pindah ke LoginActivity.
- finish() menutup splash agar tidak bisa kembali ke sini.