

Soal Essay

1. Apa fungsi `setOnClickListener`?

`setOnClickListener` adalah method pada class `View` di Android yang berfungsi untuk memberikan callback ketika view tersebut (contoh: `Button`, `ImageView`) diklik oleh pengguna. Ketika dipanggil, kita mengimplementasikan interface `View.OnClickListener` yang akan menjalankan kode di dalam method `onClick(View v)` saat view disentuh.

2. Apa syarat pemanggilan method `setOnClickListener`?

- View sudah diinisialisasi: Pastikan objek view (misal `Button`) sudah terhubung dengan layout XML menggunakan `findViewById`.
- Tidak null: Objek view tidak boleh bernilai null.
- Dipanggil setelah `setContentView`: Panggilan `findViewById` dan `setOnClickListener` harus setelah `setContentView(R.layout.nama_layout)` di `onCreate()`.

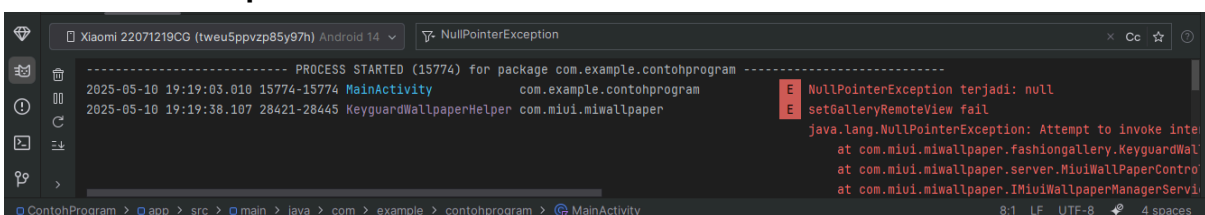
Screenshot source code:

```
class MainActivity : AppCompatActivity() {  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.activity_main)  
        val btnClickMe: Button = findViewById(R.id.btnClickMe)  
        btnClickMe.setOnClickListener {  
            Toast.makeText(this, "Button diklik!", Toast.LENGTH_SHORT).show()  
        }  
    }  
}
```

3. Error apa yang terjadi jika file kotlin salah menginisialisasi `findViewById` atau objek pada xml belum diinisialisasi? Screenshot logcat-nya!

Jika `findViewById` memanggil ID yang salah atau sebelum `setContentView`, maka objek view akan null. Saat memanggil `setOnClickListener` atau akses properti lain, aplikasi akan crash dengan `NullPointerException`.

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



Studi Kasus

Source Code Splash Screen

```
package com.example.contohprogram

import android.content.Intent
import android.os.Bundle
import android.os.Handler
import androidx.appcompat.app.AppCompatActivity

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

        Handler().postDelayed({ // Handler digunakan untuk menunggu beberapa detik

            val intent = Intent(packageContext: this, RegisterActivity::class.java)
            startActivity(intent) // Mengarahkan ke LoginActivity setelah delay 3 detik
            finish() // Menutup SplashScreenActivity setelah pindah activity
        }, delayMillis: 3000) // untuk delay 3 detik
    }
}
```

Source Code RegisterActivity

```
1 package com.example.contohprogram
2
3 import android.content.Intent
4 import android.os.Bundle
5 import android.widget.Button
6 import android.widget.EditText
7 import android.widget.ImageView
8 import android.widget.TextView
9 import android.widget.Toast
10 import androidx.appcompat.app.AppCompatActivity
11
12 <?xml> class RegisterActivity : AppCompatActivity() { // Mendefinisikan kelas RegisterActivity yang merupakan subclass dari AppCompatActivity.
13     override fun onCreate(savedInstanceState: Bundle?) { // Fungsi onCreate() dipanggil saat activity pertama kali dibuat.
14         super.onCreate(savedInstanceState) // Memanggil onCreate dari AppCompatActivity agar fungsi standar tetap dijalankan.
15         setContentView(R.layout.activity_register) // Menentukan layout XML (activity_register.xml) yang digunakan untuk tampilan activity ini.
16
17         val edtUsername: EditText = findViewById(R.id.edtUsername) // Menghubungkan komponen layout dengan kode Kotlin (mengambil referensi view
18         val edtPassword: EditText = findViewById(R.id.edtPassword)
19         val edtConfirmPassword: EditText = findViewById(R.id.edtConfirmPassword)
20         val btnRegister: Button = findViewById(R.id.btnRegister)
21         val imageView: ImageView = findViewById(R.id.imageView)
22
23         btnRegister.setOnClickListener { // Menentukan aksi saat tombol Register ditekan.
24             val username = edtUsername.text.toString().trim() // Mengambil input dari pengguna dan mengubahnya menjadi String.
25             val password = edtPassword.text.toString()
26             val confirmPassword = edtConfirmPassword.text.toString()
27
28             // Validasi input kosong
```

```

29 // Validasi input kosong
    if (username.isEmpty() || password.isEmpty() || confirmPassword.isEmpty()) {
        Toast.makeText(context, this, text: "All fields must be filled", Toast.LENGTH_SHORT).show()
        return@setOnClickListener
    }

    // Validasi kecocokan password
    if (password != confirmPassword) {
        Toast.makeText(context, this, text: "Passwords do not match", Toast.LENGTH_SHORT).show()
        return@setOnClickListener
    }

    // Jika validasi berhasil
    Toast.makeText(context, this, text: "Registration Successful", Toast.LENGTH_SHORT).show()
    android.util.Log.d(tag: "RegisterActivity", msg: "Registration Successful: $username")

    // Pindah ke LoginActivity
    val intent = Intent(packageContext, this, LoginActivity::class.java)
    startActivity(intent)
    finish()
}
}
}

```

Source Code LoginActivity

```

1 package com.example.contohprogram
2
3 > import ...
4
5 class LoginActivity : AppCompatActivity() { // Mendefinisikan kelas LoginActivity yang merupakan turunan dari AppCompatActivity.
6     override fun onCreate(savedInstanceState: Bundle?) { // Fungsi onCreate() dipanggil saat activity dibuat pertama kali.
7         super.onCreate(savedInstanceState) // Memanggil fungsi onCreate dari superclass untuk menyelesaikan proses inisialisasi dasar.
8         setContentView(R.layout.activity_login) // Menetapkan layout XML (activity_login.xml) yang akan digunakan untuk tampilan activity ini.
9
10         val btnLogin: Button = findViewById(R.id.btnLogin) // Menghubungkan elemen UI dari layout ke dalam variabel Kotlin.
11         val edtUsername: EditText = findViewById(R.id.edtUsername)
12         val edtPassword: EditText = findViewById(R.id.edtPassword)
13
14         btnLogin.setOnClickListener { // Menentukan aksi yang dilakukan saat tombol Login ditekan.
15             val username = edtUsername.text.toString() // Mengambil input teks dari EditText dan mengubahnya ke dalam bentuk String.
16             val password = edtPassword.text.toString()
17
18             if (username.isNotEmpty() && password.isNotEmpty()) { // Mengecek apakah kedua field (username dan password) tidak kosong.
19
20                 // Menampilkan Toast dan Log ketika login sukses
21                 Toast.makeText(context, this, text: "Login Successful", Toast.LENGTH_SHORT).show()
22                 android.util.Log.d(tag: "LoginActivity", msg: "Login Successful: $username")
23
24                 // Berpindah ke ListChattingActivity
25                 val intent = Intent(packageContext, this, ListChattingActivity::class.java)
26                 startActivity(intent)
27
28             } else {
29                 // Menampilkan Toast jika username atau password kosong
30                 Toast.makeText(context, this, text: "Please enter both fields", Toast.LENGTH_SHORT).show()
31             }
32         }
33     }
34 }
35
36 }
37
38 }

```

Source Code ListChattingActivity

```
1 package com.example.contohprogram
2
3 > import ...
4
5
6
7
8 class ListChattingActivity : AppCompatActivity() {
9     override fun onCreate(savedInstanceState: Bundle?) {
10         super.onCreate(savedInstanceState)
11         setContentView(R.layout.activity_list_chatting)
12
13         val listView: ListView = findViewById(R.id.listView) //Menghubungkan komponen ListView dari layout ke variabel Kotlin.
14         val chats = arrayOf("Chat 1","Chat 2","Chat 3","Chat 4","Chat 5","Chat 6","Chat 7","Chat 8","Chat 9",
15             "Chat 10") // Daftar percakapan untuk ditampilkan di ListView
16         val adapter = ArrayAdapter(context=this, android.R.layout.simple_list_item_1, chats)
17         listView.adapter = adapter // ArrayAdapter digunakan untuk menghubungkan data dengan ListView
18     }
19 }
```