

Perancangan Aplikasi Qeuangans)

logbook



Faris Muhammad Ihsan

1.18.4.099

Syabriena Putri Veriane

1.18.4.094

Informatics Research Center

Applied Bachelor Program of Informatics Engineering

Bandung

2019

‘Jika Kamu tidak dapat menahan lelahnya belajar,
Maka kamu harus sanggup menahan perihnya Kebodohan.’
Imam Syafi’i

Contents

1	Pertemuan 1	1
1.1	Issues #1	1
1.2	Issues #2	2
1.3	Issues #3	2
1.4	Issues #4	3
1.5	Issues #5	4
1.6	Issues #6	5
1.7	Issues #7	5
1.8	Issues #8	6
1.9	Issues #9	6
1.10	Issues #10	6

Chapter 1

Pertemuan 1

1.1 Issues #1

Pada *issues #1 (Hardcoded String)* *Hardcoded string* sebenarnya bukan merupakan *error*, namun hanya sebagai peringatan. Peringatan ini terjadi karena menyimpan *hard code string* pada file *layout*.

```
android:text="Tabungan anda"
```

Pemecahan dari Hardcoded String tersebut adalah dengan menuliskan *string* pada file terpisah yang telah disediakan yaitu String.XML. Penulisan *Hardcode String* yang terpisah pada file String.XML ini dapat memudahkan developer pada saat akan melakukan perubahan nama. Developer hanya perlu mengubah pada file String.XML.

Solusinya, Pada file String.XML dituliskan baris kode seperti:

```
<resources>
<string name="app_name">Qeuangans</string>
<string name="Tabungan_anda">Tabungan Anda</string>
</resources>
```

Kemudian, untuk Memanggil *Hardcoded String* menggunakan perintah:

```
android:text="@string/Tabungan_anda"
```

Perintah tersebut dituliskan pada MainActivity.XML

1.2 Issues #2

Pada *issues #2 (Error: Tag start is Not Closed)* berisi *Error* pada file XML. Masalah ini disebabkan karena pada file XML tersebut terdapat baris perintah tanpa *tag* penutup. XML merupakan file (*eXtensible Markup Language*) dan merupakan bahasa *Markup* layaknya bahasa HTML. Pada bahasa XML ini diperlukan tag pembuka dan penutup(contoh: $\langle \rangle$), jika ada salah satu tag (baik pembuka maupun penutup) yang tidak ditulis, maka akan terbentuk error dan baris perintah tidak akan di eksekusi.

```
<TextView android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_marginTop="18dp"
android:textColor="#000000"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintRight_toRightOf="parent"
app:layout_constraintTop_toTopOf="parent"
android:text="@string/Tabungan_anda"
```

Pada *error* ini memperbaikinya dengan cara menambahkan tag pada akhir baris perintah.

```
<TextView android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentTop="true"
android:layout_marginTop="18dp"
android:textColor="#000000"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintRight_toRightOf="parent"
app:layout_constraintTop_toTopOf="parent"
android:text="@string/Tabungan_anda" />
```

1.3 Issues #3

Pada *issues #3 (Unusend Import Statement)* yaitu ketika kita akan mengimport suatu *statement* maka akan menuliskannya pada awal baris kode. Mengimport disini

berarti mengambil method yang ada pada kelas lain. Namun, *Unused Import Statement* akan muncul ketika ada baris *import* yang tidak digunakan. Hal ini hanya berupa peringatan dan bukan error.

```
package com.example.qeuangans;

import androidx.appcompat.app.AppCompatActivity;

import android.database.Cursor;
import android.graphics.Color;
import android.graphics.drawable.ColorDrawable;
import android.os.Bundle;
import android.util.Log;
import android.view.View; //Unused Import Statement: Issues \#6
import android.widget.AdapterView; //Unused Import Statement: Issues \#6
import android.widget.AdapterView;
import android.widget.Toast;
```

Solusinya adalah dengan menghapus import yang tidak digunakan karena method-nya tidak dibuat, maka perintah import tersebut dihapus.

```
package com.example.qeuangans;

import androidx.appcompat.app.AppCompatActivity;

import android.database.Cursor;
import android.graphics.Color;
import android.graphics.drawable.ColorDrawable;
import android.os.Bundle;
import android.util.Log;
import android.widget.AdapterView;
import android.widget.Toast;
```

1.4 Issues #4

Pada *issues #4* (*Error: Cannot Resolve Method*) disini merupakan jenis *error* yang disebabkan karena pemanggilan method yang salah atau method yang belum dibuat. Namun ketika baris kode ini dijalankan tidak ada nama method yang sesuai. Pada

kasus ini, terdapat method yang belum dibuat namun sudah dipanggil. Kode Untuk memanggilnya menggunakan.

```
jmlSaldo();
```

Pada pemanggilan tersebut terdapat method yang belum dibuat.

Solusi yang harus dilakukan adalah dengan menambahkan method dengan nama jmlSaldo(); beserta isi dari method yang dipanggil tersebut.

```
private void jmlSaldo() {  
    Cursor cursor = db.saldo();  
    while (cursor.moveToNext()) {  
        textsaldo.append("Rp " + cursor.getString(0) + ",00");  
    }  
}
```

1.5 Issues #5

Pada *issues #5 (Error: Cannot Resolve Symbol)* Merupakan jenis error ketika ada objek yang dipanggil namun id nya tidak sama atau tidak ada (*typo*), yang membuat baris kode tersebut tidak terbaca dan error sehingga objek untuk menampilkan saldo pada tampilah aplikasi tidak akan muncul.

```
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="57dp"  
    android:layout_alignParentTop="true"  
    android:layout_marginTop="58dp"  
    android:hint="@string/Rp00"  
    android:textColor="#000000"  
    android:textSize="20sp"  
    android:id="@+id/textado" <!-- Cannot Resolve Symbol: Issues #3--> />
```

Solusinya adalah memberi id sesuai dengan yang dipanggil sehingga tampilan saldo akan muncul.

```
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="57dp"
```

```

        android:layout_alignParentTop="true"
        android:layout_marginTop="58dp"
        android:hint="@string/Rp00"
        android:textColor="#000000"
        android:textSize="20sp"
        android:id="@+id/textsaldo" <!-- Cannot Resolve Symbol: Issues #3--> />

```

1.6 Issues #6

Pada *issues #6 (intent Pindah())* ini digunakan sebagai perintah untuk pindah dari *Activity* yang satu ke yang lainnya. Intent merupakan objek yang menyediakan fungsi untuk pindah dari satu *activity* ke *activity* lainnya. Penggunaan intent dari *method* pindah sebagai berikut:

```

public void Pindah(View view) {
    Intent intent = new Intent(MainActivity.this, InputData.class);
    startActivity(intent);
}

```

Intent pada method ini digunakan untuk pindah dari *Activity* MainActivity.class ke *Activity* InputData.class. Intent disini diaktifkan dengan perintah startActivity(intent);

1.7 Issues #7

Pada *issues #7 (Error: Semicolon Expected)* *Semicolon Expected* adalah *error* yang terjadi ketika sebuah baris program yang sudah ditulis kekurangan tanda *semicolon(;)* pada akhir dari program, sehingga ketika baris program di eksekusi akan memunculkan peringatan *error Semicolon Expected*. Contoh error:

```
startActivity(intent)
```

Solusinya adalah dengan menambahkan *semicolon(;)* pada akhir baris program.

```
startActivity(intent);
```


1.8 Issues #8

Pada *issues #8 (Error: Cannot resolve symbol) Cannot Resolve Symbol* disini adalah ketika sebuah method yang menggunakan method dari kelas lainnya tapi tidak melakukan impor method maka akan muncul error ini. Solusinya adalah dengan cara menambahkan import method pada awal kode program.

1.9 Issues #9

Pada *issues #9 (Error: } Expected)* merupakan error yang disebabkan karena kurangnya tanda (}) pada akhir kode program. Sama halnya seperti file XML yang memerlukan tag pembuka dan penutup, sebuah method pada java juga harus menggunakan tanda pembuka({) dan tanda penutupnya(}). Contoh *Error: } Expected:*

```
public void Pindah(View view) {  
    Intent intent = new Intent(MainActivity.this, InputData.class);  
    startActivity(intent);
```

Solusinya adalah dengan menambahkan tanda penutup} pada akhir method

```
public void Pindah(View view) {  
    Intent intent = new Intent(MainActivity.this, InputData.class);  
    startActivity(intent);  
}
```

1.10 Issues #10

Pada *issues #10 (Error: Missing Parent) Missing Parent* adalah *error* yang disebabkan ketika kita ingin mendefinisikan objek namun belum mengimport library nya.