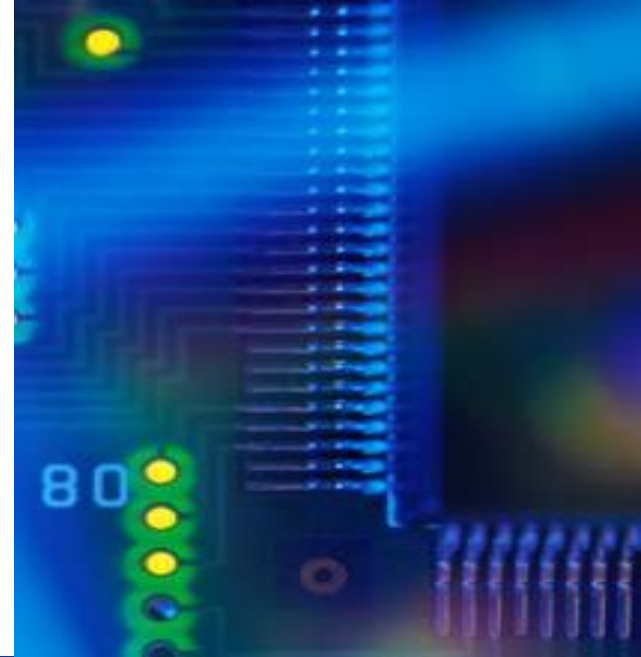




KEMENTERIAN KOMUNIKASI DAN INFORMATIKA  
REPUBLIK INDONESIA

*Menuju Masyarakat Informasi Indonesia*



## JUNIOR MOBILE PROGRAMMER

# Merancang Mobile Interface Menggunakan Android Studio

## Deskripsi Singkat

### Deskripsi Singkat mengenai Topik

Materi pelatihan ini memfasilitasi pembentukan kompetensi dalam membuat aksi-aksi atau design yang estetik yang ada dalam suatu layar mobile yang sesuai dengan kebutuhan

### Tujuan Pelatihan

1. Peserta mampu membuat Icon atau gambar yang mempresentasikan suatu informasi dibuat berdasarkan spesifikasi aplikasi.
2. Peserta mampu mengubah Jenis font dipilih sesuai kebutuhan.
3. Peserta mampu membuat Desain ukuran font dibuat agar membuat nyaman pengguna.

### Materi Yang akan disampaikan:

1. Pengenalan Model-View-Presenter pattern
2. Pengenalan Layout dan Merancang Layout
3. Menambahkan Komponen Tombol Pada Tampilan
4. Menambahkan Komponen Menu
5. Memberi Perintah atau Aksi Pada Tombol dan Menu

# Merancang Mobile Interface android studio

## Pelatihan

**Tugas : *Membuat aplikasi sederhana yang berisi tombol dan menu seperti program game tictac atau puzzle ABC***

### Outcome/Capaian Pelatihan

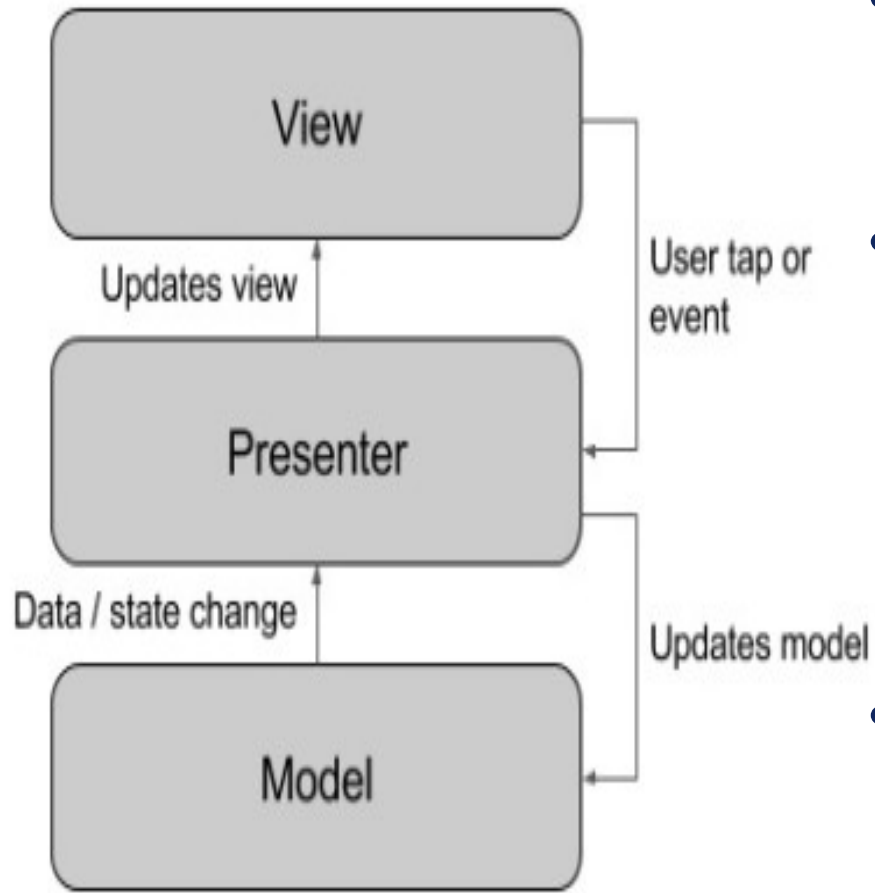
- mempraktikkan cara menyesuaikan desain ukuran tombol atau menu dengan ukuran *form* aplikasi
- Mampu menyesuaikan ukuran tombol atau menu dengan ukuran *form* aplikasi sesuai dengan kebutuhan aplikasi.
- Dapat menjelaskan desain ukuran tombol atau menu disesuaikan dengan ukuran *form* aplikasi dibuat sesuai dengan kebutuhan aplikasi.

## 1. Pengenalan Model-View-Presenter pattern

## Pelatihan

- **Model-View-Presenter (MVP)** adalah salah satu pendekatan pengembangan program yang membagi pengembangan menjadi 3 kelompok utama yaitu Model, View dan Presenter.
- Konsep ini umum digunakan dalam pemrograman mobile dengan menggunakan Android Studio. Konsep ini mirip dengan konsep MVC pada OOP. 3 kelompok resource dalam MVP tersebut adalah:

## Pelatihan



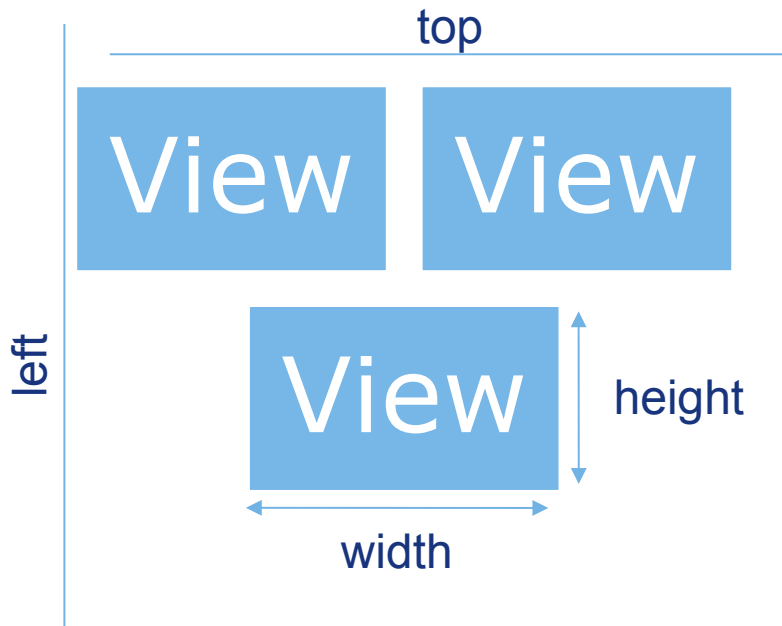
- **View** □ adalah elemen user interface yang menampilkan data dan dapat melakukan respon terhadap aksi yang dilakukan oleh user
- **Presenters** □ adalah resource yang menghubungkan antara view dan model, dimana presenter ini dapat memberikan data yang dispesifikasi oleh model kepada view, dan sebaliknya memberikan data yang diinputkan ke view oleh user kepada model.
- **Model** □ adalah spesifikasi dari struktur data dari sebuah aplikasi dimana model juga dapat berisi perintah untuk mengakses dan memanipulasi data tersebut

## View (1.1)

## Pelatihan

Catatan:

- User interface pada pemrograman android terdiri dari hirarki object yang diberi nama **view**
- **Class View** merepresentasikan blok dasar dari semua komponen UI yang akan dipasang
- View memiliki lokasi penempatan yang diterjemahkan kedalam pasangan koordinat **left** dan **top** (titik kiri dan atas layer tampilan) dan dengan bentuk 2(dua) dimensi yang diwakili oleh **width** dan **height** (Panjang dan tinggi) dari view tersebut.
- Satuan yang digunakan dalam penentuan lokasi dan dimensi ini biasanya adalah device-independent pixel (**dp**).



Satuannya dp (device-independent pixel)

## View (1.2)

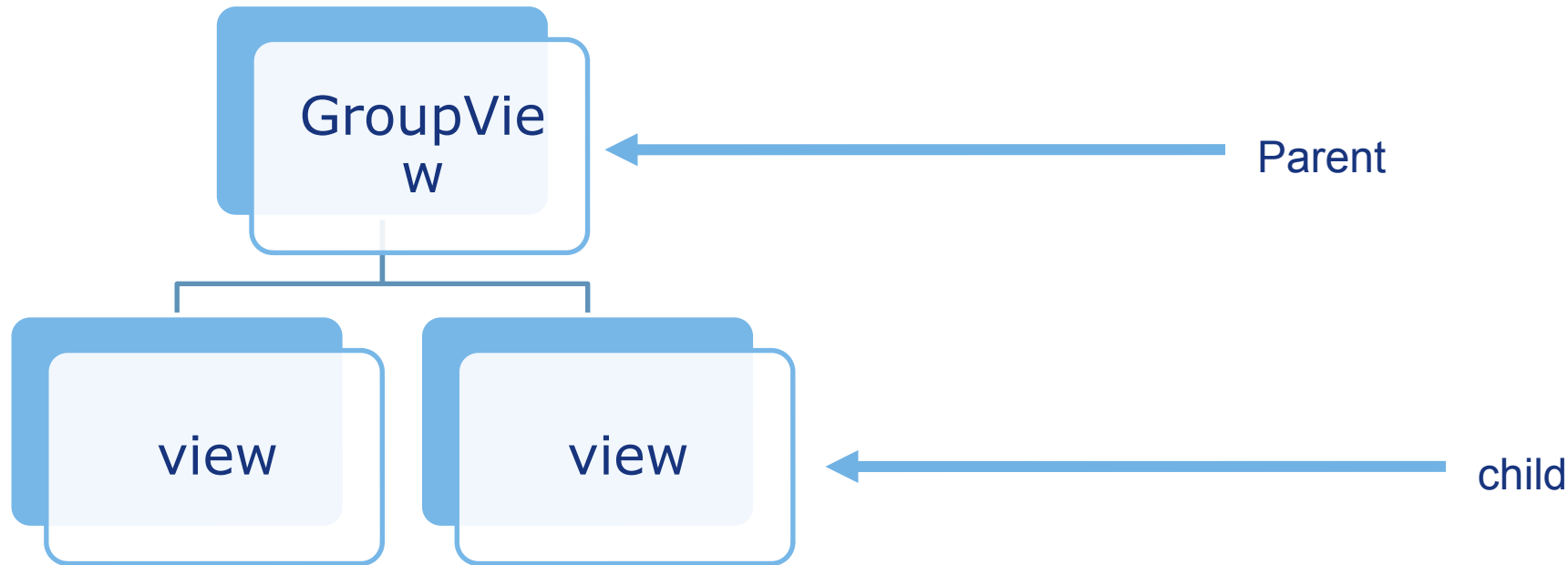
## Pelatihan

Sistem Android menyediakan banyak class view yang sudah didefinisikan, diantaranya:

- Text, (TextView)
- Fields, untuk input dan editing text (EditText)
- Buttons, dimana user dapat melakukan tap (Button) dan komponen interactive lainnya
- Scrollable text (ScrollView)
- Scrollable items (RecyclerView)
- Images (ImageView)
- Dan yang lainnya

## Group View (1.3)

## Pelatihan



**Views** dapat dikelompokkan kedalam sebuah view group (**ViewGroup**) yang berperan sebagai sebuah container dari views (container adalah sebuah kotak penampung view).

Hubungan dari view dalam container adalah **parent-child** (bapak-anak atau kelompok besar yang terdiri dari bagian kecil) dimana bapak adalah view group dan anak adalah view atau dapat juga berupa group view lainnya yang memiliki bagian anak lagi.



## Group View (1.4)

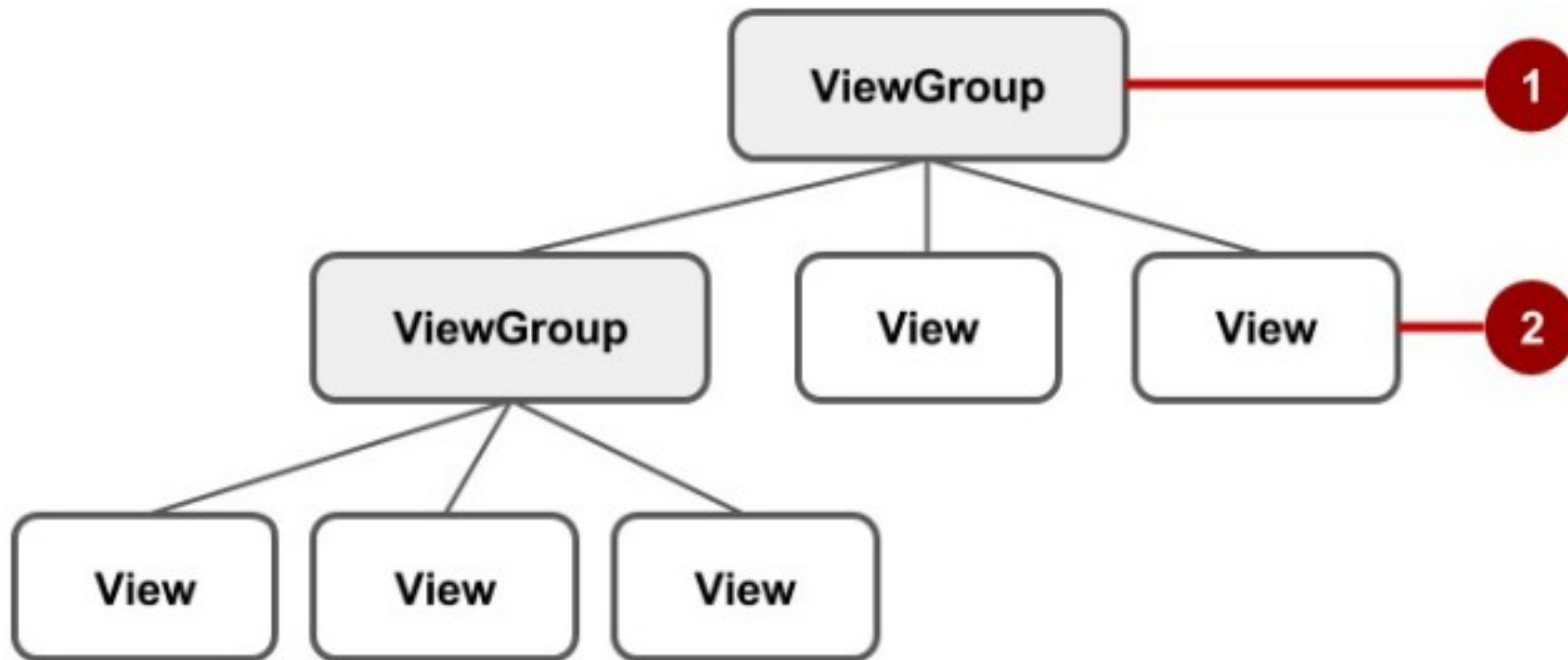
## Pelatihan

Bentuk ViewGroup yang umum adalah:

- **ScrollView** : adalah group yang terdiri dari satu atau lebih child view yang dapat di scrolling ke atas atau bawah
- **RecyclerView** : Adalah group yang terdiri dari list view lain atau group view yang dapat di scrolling dan juga dapat di tambahkan view baru atau menghapus view yang sudah ada secara dinamis

## Layout Group View (1.5)

## Pelatihan



### Keterangan

1. Root view group (view group paling utama)
2. View Group atau view yang merupakan anak dari viewgroup root

## 2. Pengenalan Layout dan Merancang Layout

## Pelatihan

- Beberapa ViewGroup sudah dirancang dalam bentuk Layout yang akan menjadi parent dari sebuah view atau groupview
- Layout adalah tata letak komponen,
- Layout berfungsi untuk mengatur antarmuka aplikasi dan posisi penempatan komponen seperti tombol atau text.
- Layout pada Android Studio disimpan dalam bentuk file XML, dalam path **/res/layout** folder

## 2. Pengenalan Layout dan Merancang Layout

## Pelatihan

- Beberapa jenis layout pada Android Studio yaitu:
  - a. Linear Layout
  - b. Relative Layout
  - c. Constraint Layout
  - d. Table Layout
  - e. Absolute Layout
  - f. Frame Layout
  - g. Grid Layout
  - h. List View

## Layout (2.1)

## Pelatihan

### a. Linear Layout

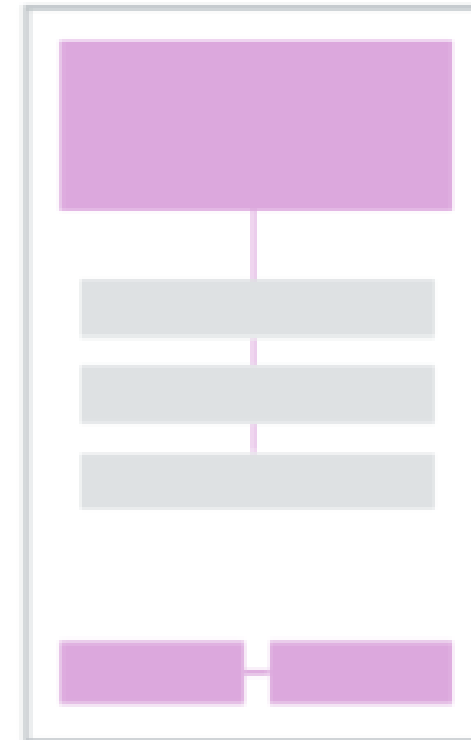


LinearLayout Horizontal



LinearLayout Vertical

### b. Relative Layout



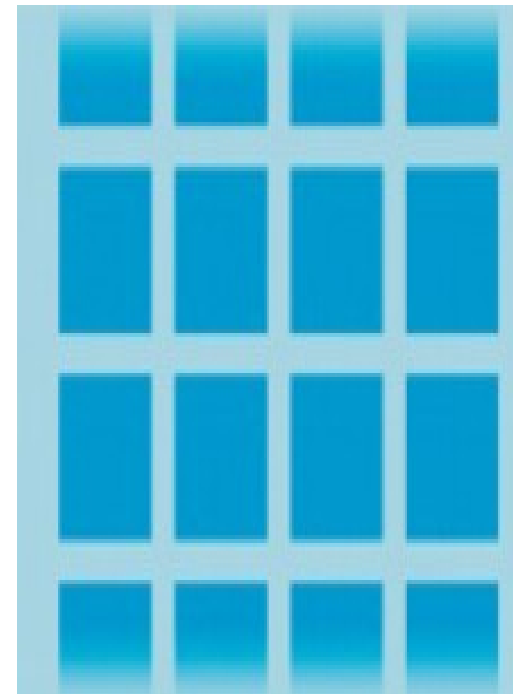
Setiap komponen yang terpasang pada relative layout posisinya akan mengikuti atau bergantung pada komponen lainnya yang berdekatan

## Layout (2.2)

## Pelatihan

**c. Constraint Layout** adalah group dari child view yang menggunakan anchor points, edges dan guidelines untuk mengontrol bagaimana view di posisikan secara relative dengan elemen lainnya pada layout.

### d. Table Layout

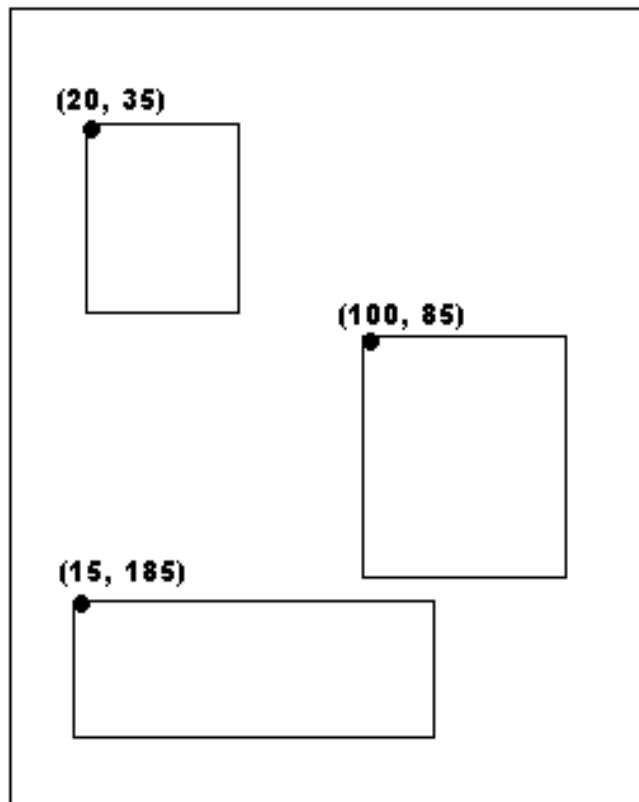


Komponen yang terpasang dalam bentuk baris dan kolom biasanya memiliki ukuran yang sama antara lebar atau tinggi dari masing-masing komponen

## Layout (2.3)

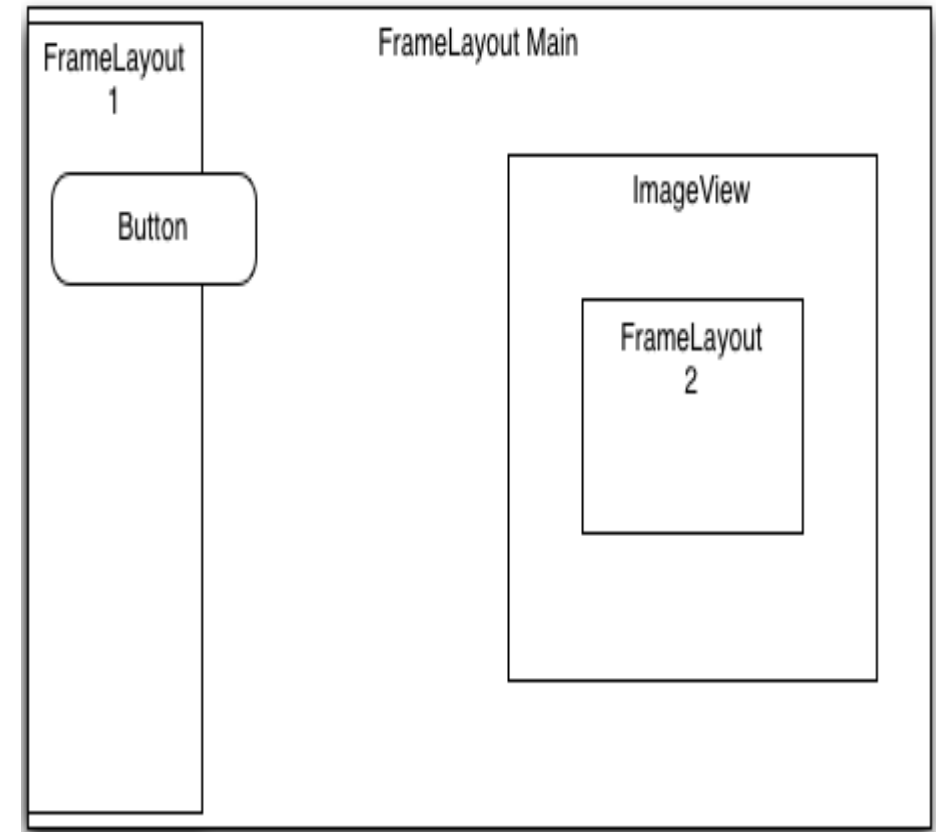
## Pelatihan

### e. Absolute Layout



Absolute Layout

### f. Frame Layout

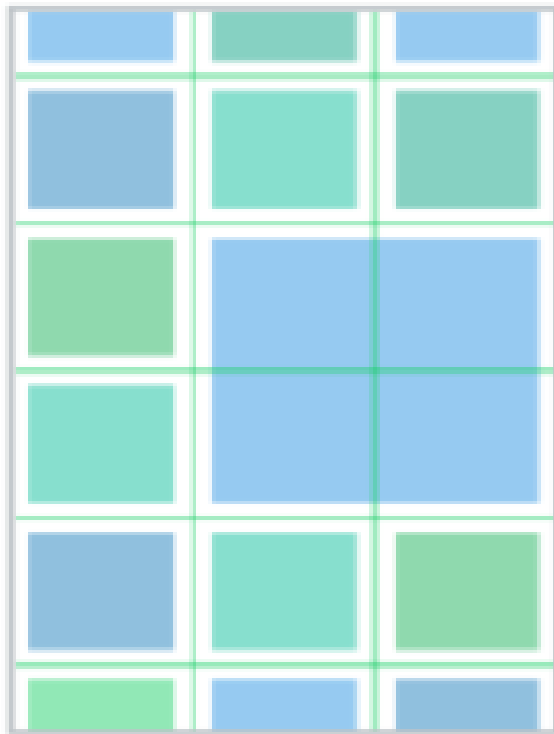


Frame Layout

## Layout (2.4)

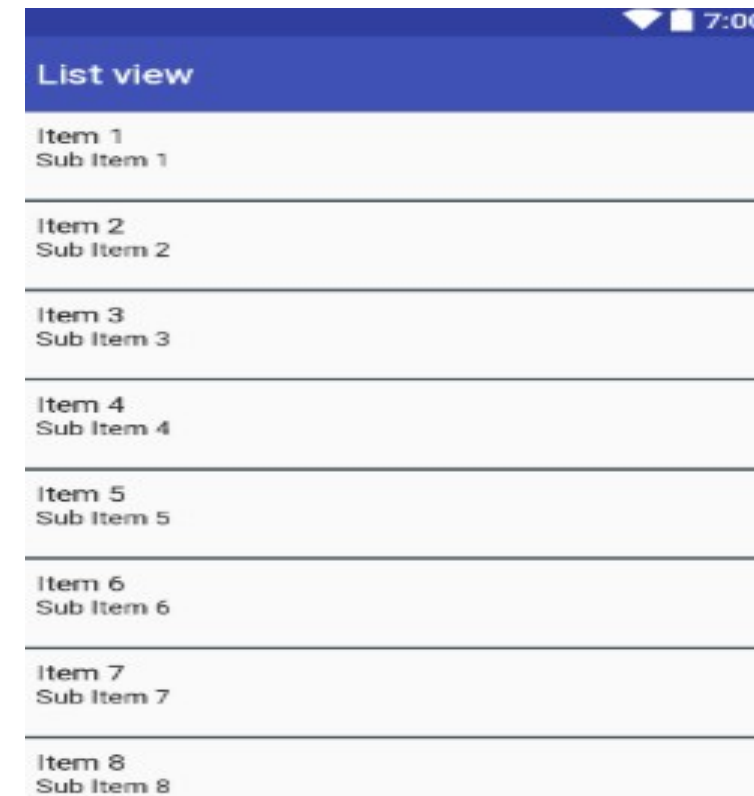
## Pelatihan

### g. Grid Layout



Grid Layout

### h. List View

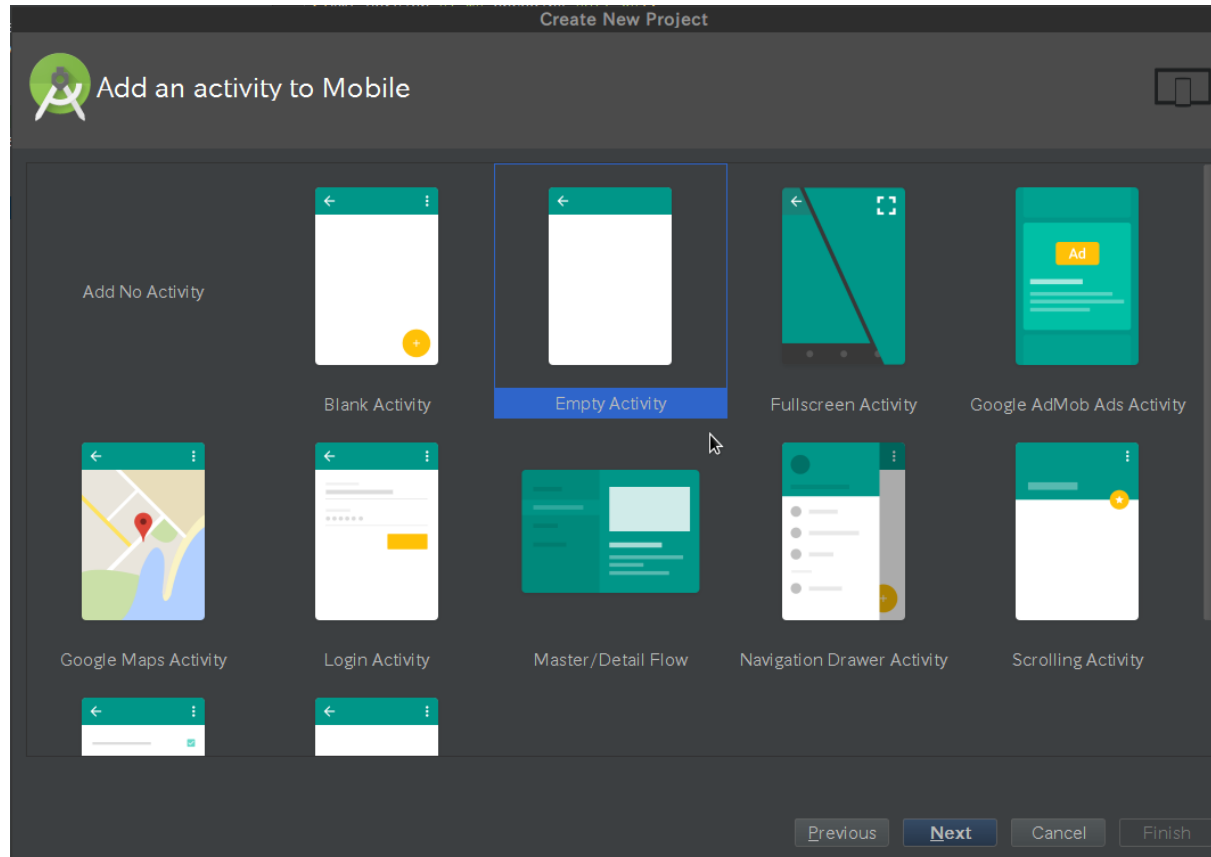


List View



## Merancang Tampilan (2.5)

## Pelatihan



- Untuk merancang tampilan pada android studio dapat dilakukan dengan beberapa cara seperti menggunakan template yang telah disediakan, atau dapat juga merancang sendiri tampilan dengan memilih blank view.
- Tidak ada aturan baku dalam merancang layout tampilan. Tampilan aplikasi dapat dibuat sesuai dengan kebutuhan system.

## Merancang Tampilan (2.6)

## Pelatihan

Beberapa hal yang perlu diperhatikan dalam merancang tampilan diantaranya:

- Komposisi peletakan komponen
- Efektifitas pemakaian komponen (misalnya kapan menggunakan check box atau combo)
- Pemilihan warna yang baik (misalnya tidak mengganggu pengelihatatan ex: banyak warna merah)
- Penggunaan gambar dan icon yang baik dan tidak mengganggu tujuan utama dari sebuah tampilan
- Pemilihan menu dan atau tombol yang sesuai dengan kebutuhan system
- Jumlah penggunaan frame yang efektif

### 3. Menambahkan Komponen Tombol Pada Tampilan

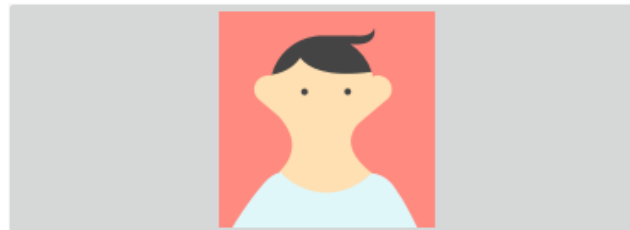
### Pelatihan

- Tombol / Button adalah sebuah komponen interface yang sangat penting untuk melakukan suatu aksi pada aplikasi, dimana dengan button user dapat melakukan interaksi dengan aplikasi
- Beberapa komponen yang termasuk kedalam group buttons pada adroid yaitu:

a. Button ( adalah komponen tombol biasa)



b. Image Button (tombol dengan gambar)



## Komponen Tombol Pada Tampilan

## Pelatihan

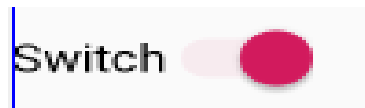
- c. Check Box (check box untuk membuat pilihan misalnya true atau false dengan bentuk kotak tercentang)



- d. Radio Button (tombol yang berbentuk bulat yang kalau dipilih akan berwarna hitam)  RadioButton1

- e. Radio Group (group dari radio button dimana kalau salah satu radio button dipilih maka otomatis radio button yang lain di group tidak akan terpilih)

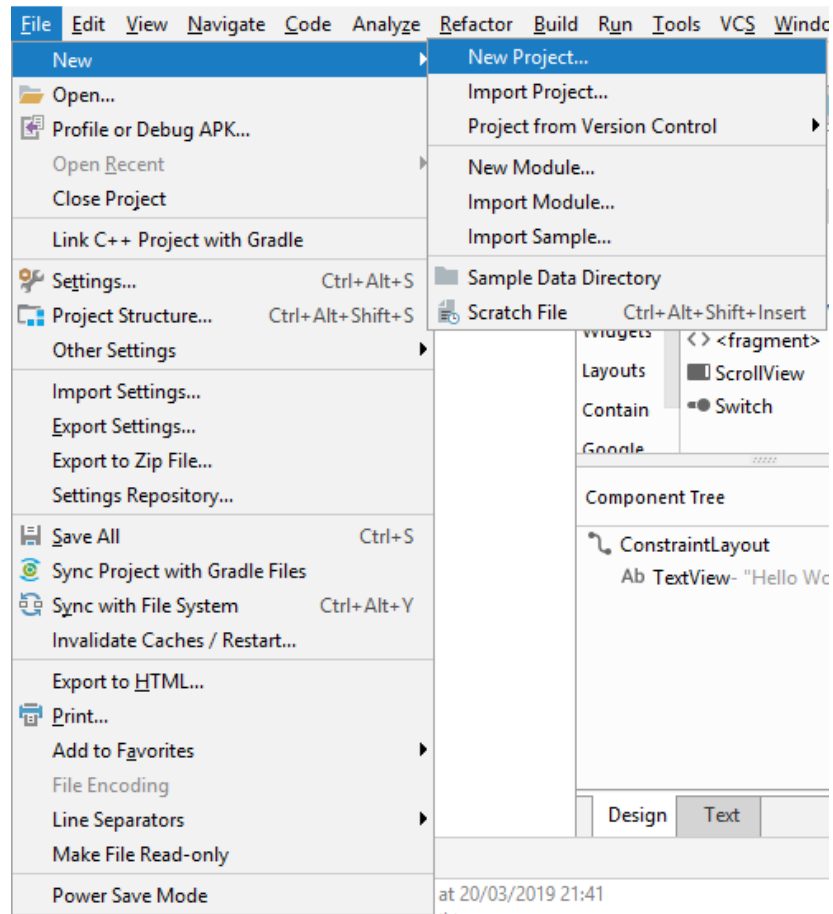
- f. Switch (Tombol yang berbentuk seperti saklar on and of)



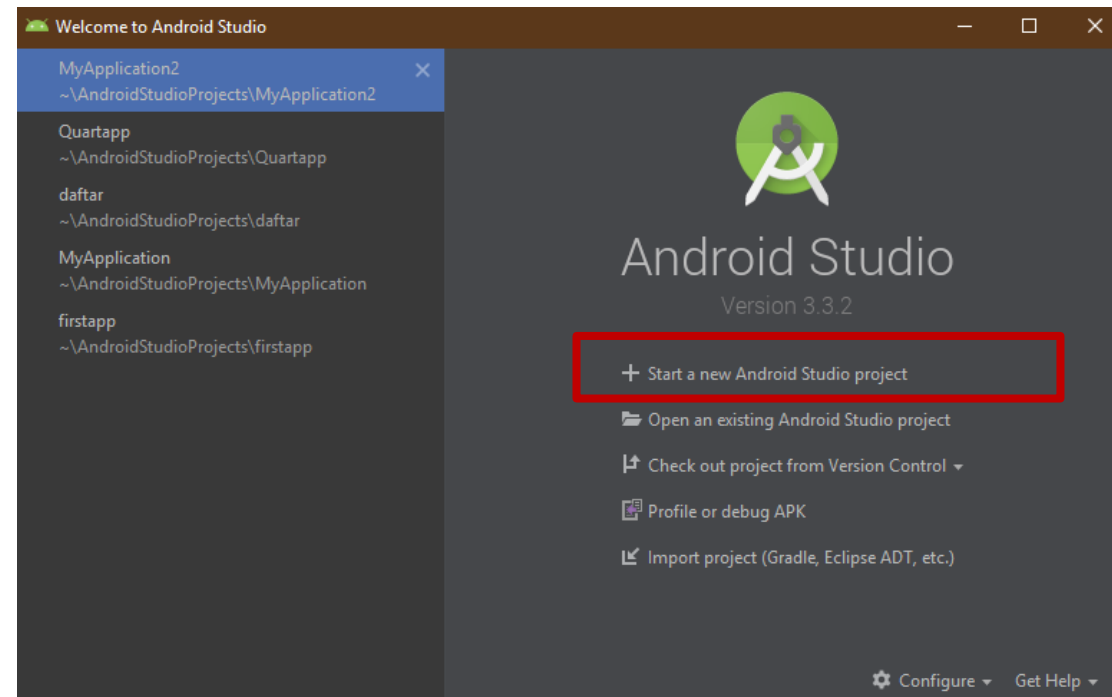
## Praktek Menambahkan Komponen Tombol (3.1)

## Pelatihan

- Buatlah sebuah project dengan nama HelooWorld dengan langkah seperti berikut

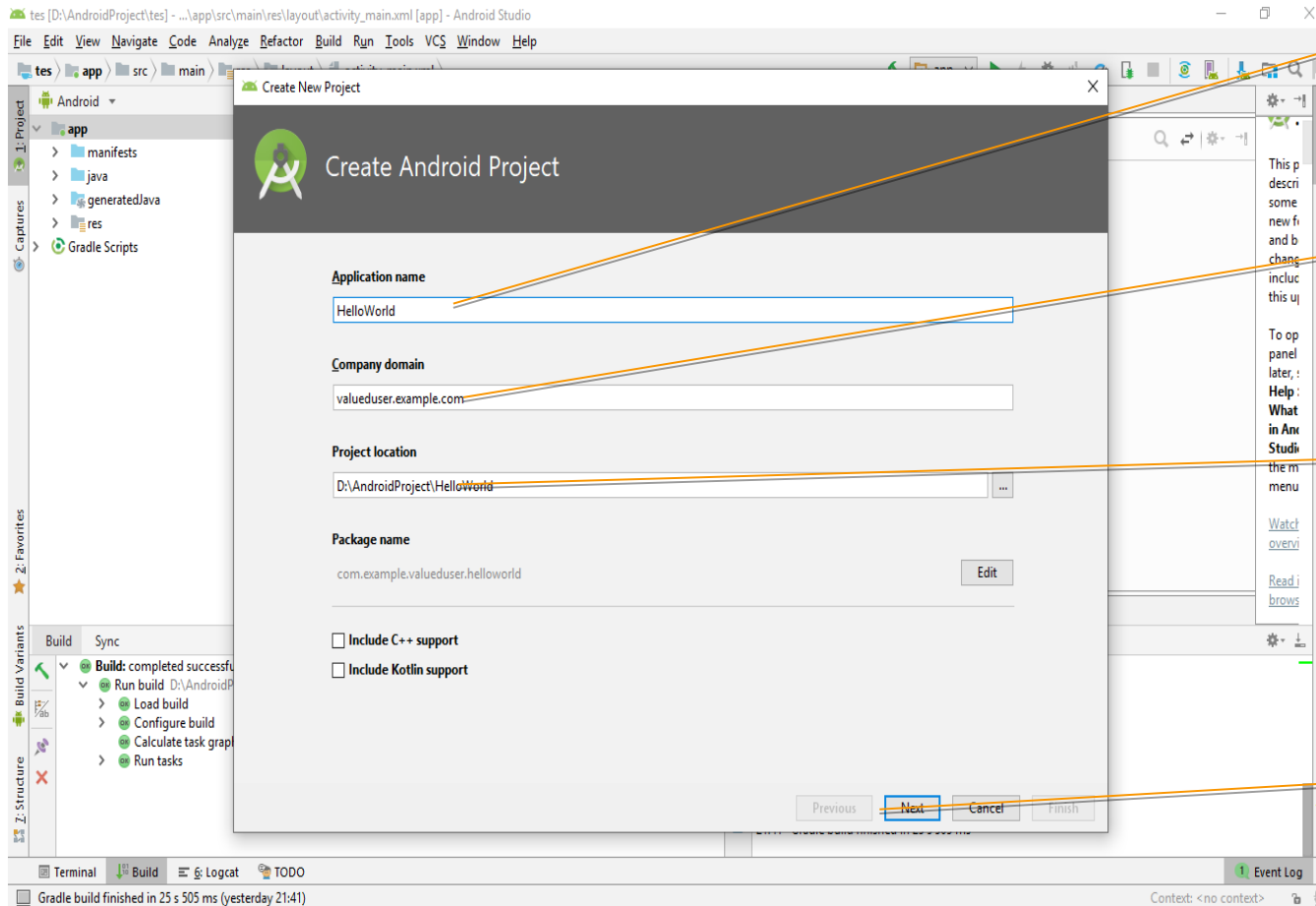


Buka Android Studio, Pilih menu:  
**File ▾ New ▾ New Project**



## Praktek Menambahkan Komponen Tombol (3.2)

## Pelatihan



Application Name, diisi dengan : "Hello World"

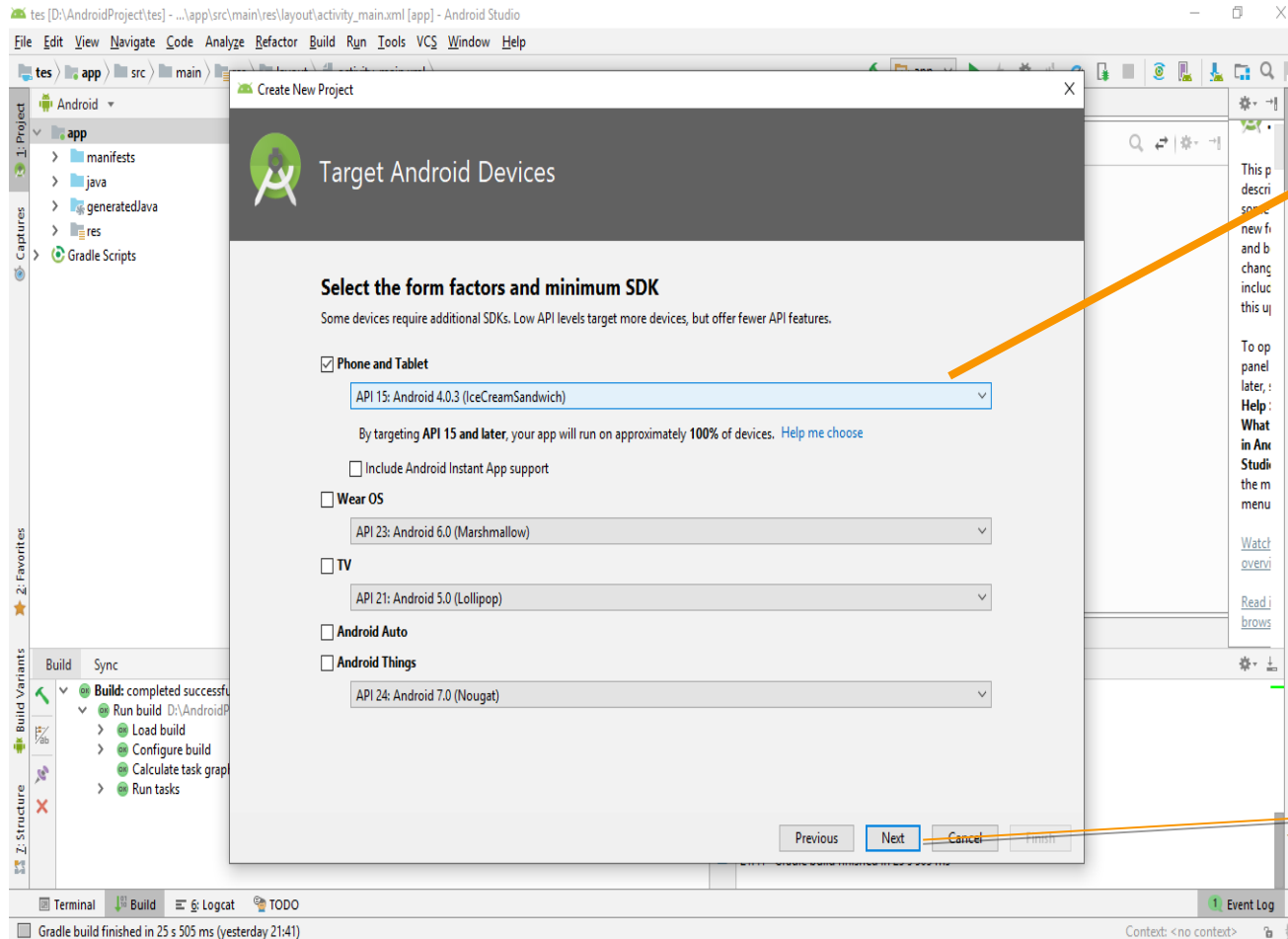
Company domain biarkan default, atau diisi kalaupunya domain

Project Location diisi dengan lokasi folder penyimpanan project

Lanjutkan proses dengan menekan tombol next

## Praktek Menambahkan Komponen Tombol (3.3)

## Pelatihan

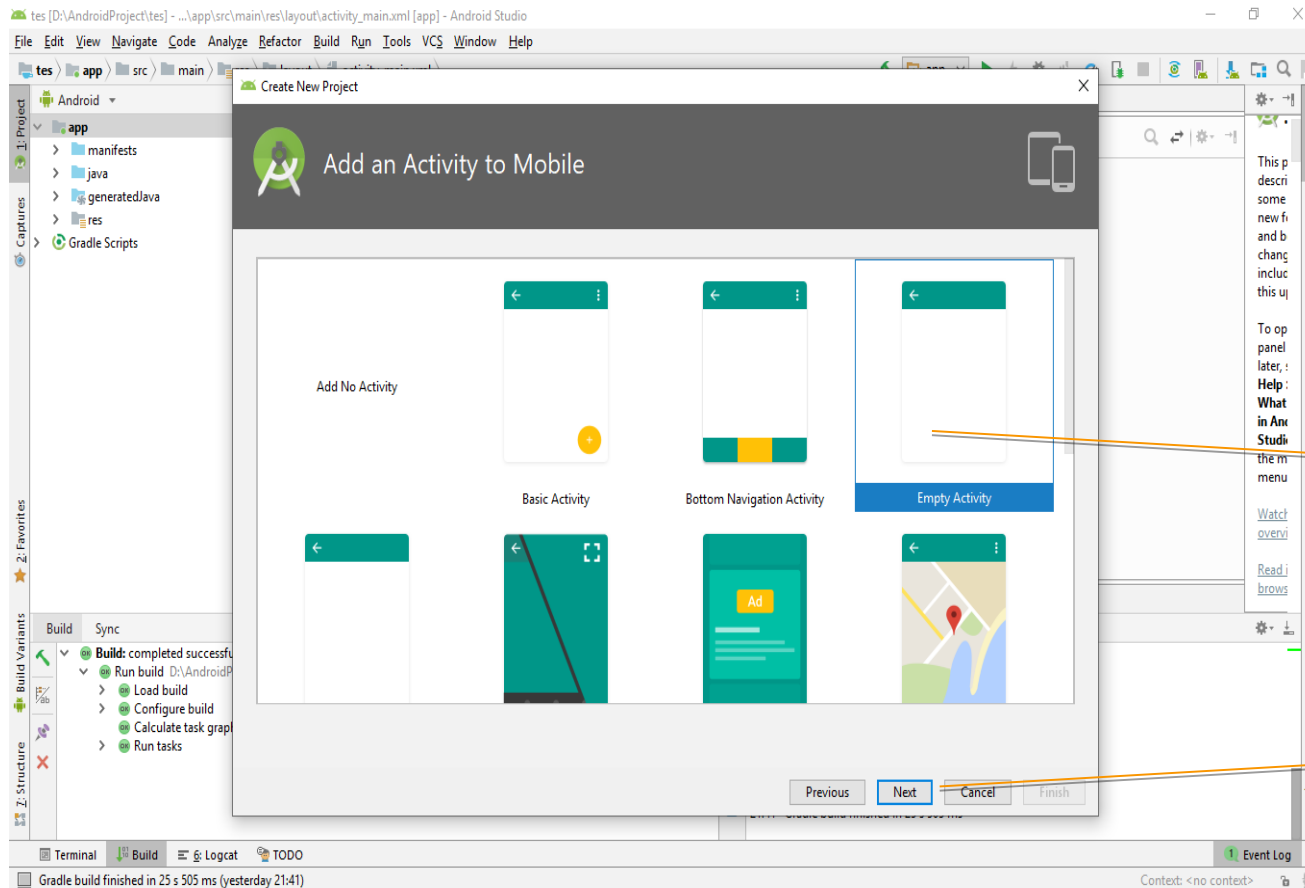


- Jendela Target Android Device.
- Jika proyek Anda memerlukan komponen tambahan untuk SDK target terpilih, Android Studio akan memasangnya secara otomatis. Klik Next.

Lanjutkan proses dengan menekan tombol next

## Praktek Menambahkan Komponen Tombol (3.4)

## Pelatihan



- Jendela Customize the Activity.
- Setiap aplikasi membutuhkan paling tidak satu aktivitas. Aktivitas mewakili satu layar dengan satu antarmuka pengguna

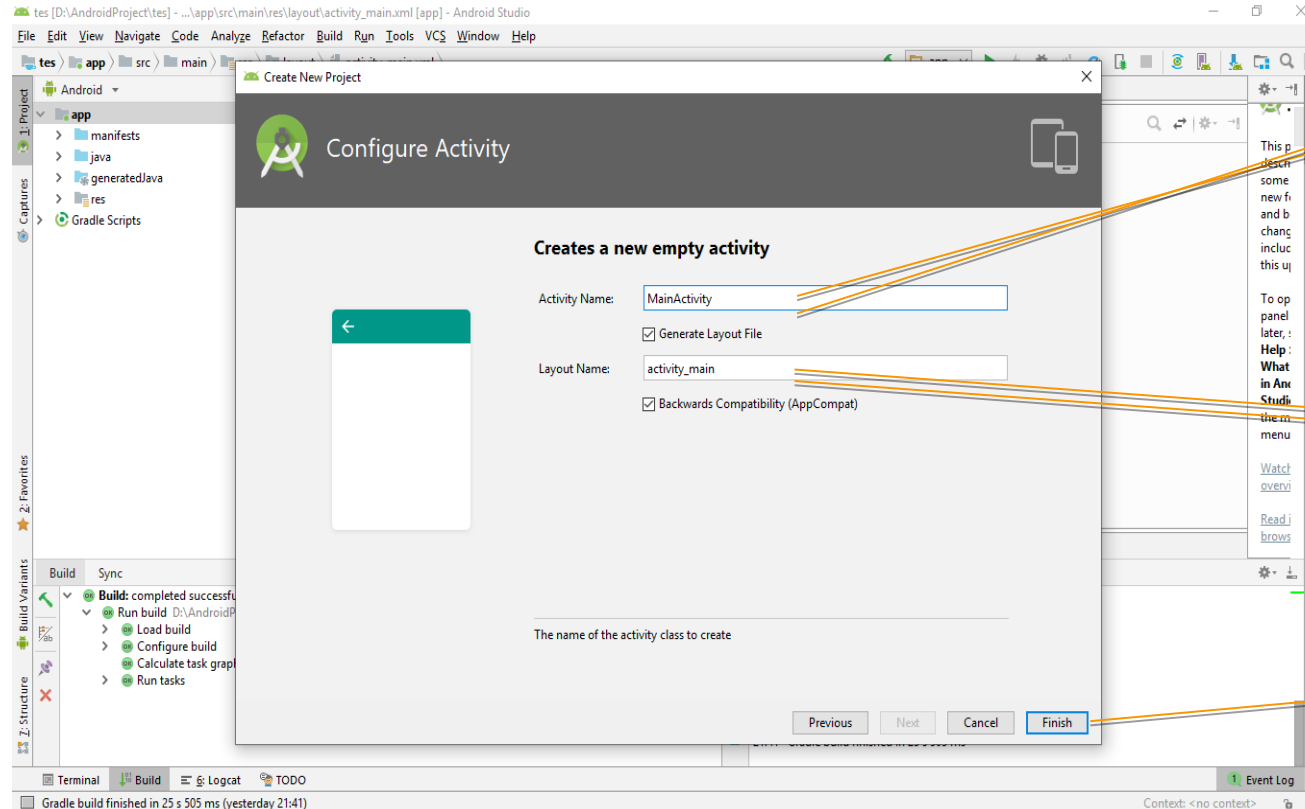
Pilih empty activity

Lanjutkan proses dengan menekan tombol next



# Praktek Menambahkan Komponen Tombol (3.5)

## Pelatihan



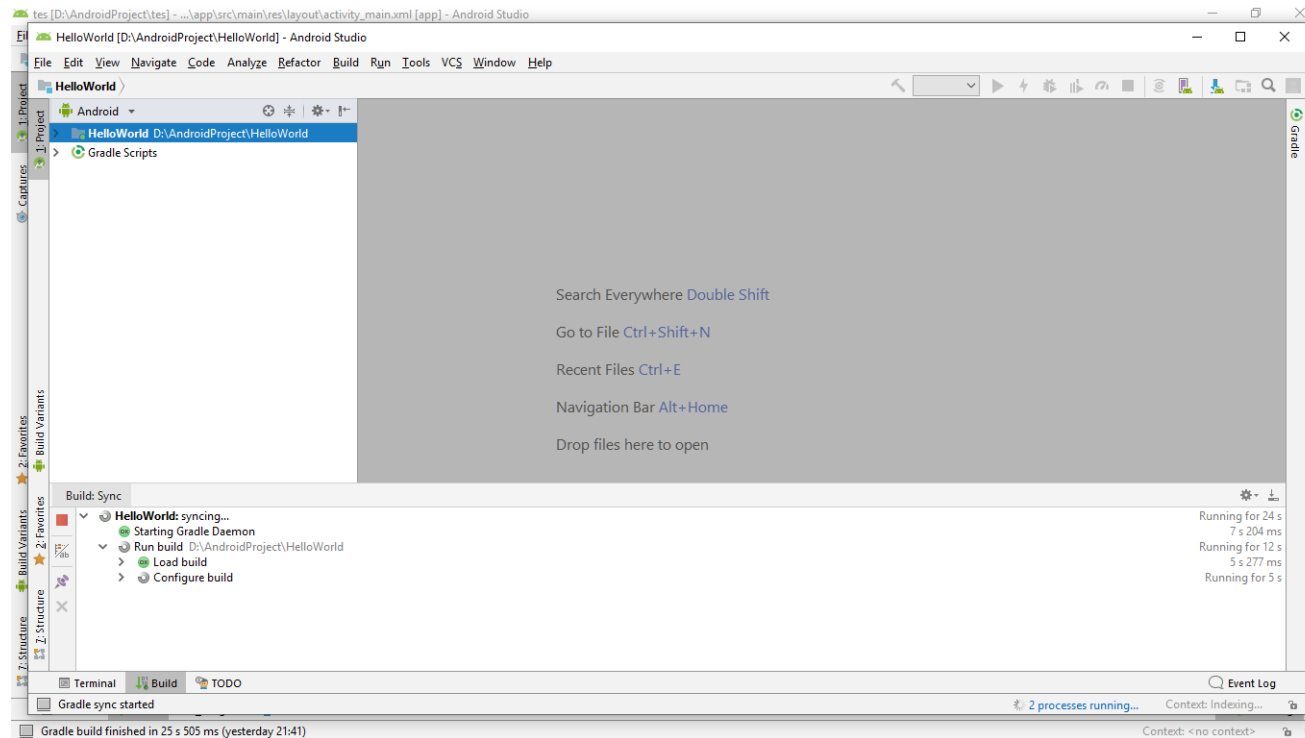
Activity Main, nama ini akan digunakan sebagai nama class main dari project. (biarkan default)

activity\_main sebagai nama layoutnya, biasanya layout name ini diberi nama sesuai dengan aktivitasnya.

Tekan tombol Finish untuk membuat project

## Praktek Menambahkan Komponen Tombol (3.6)

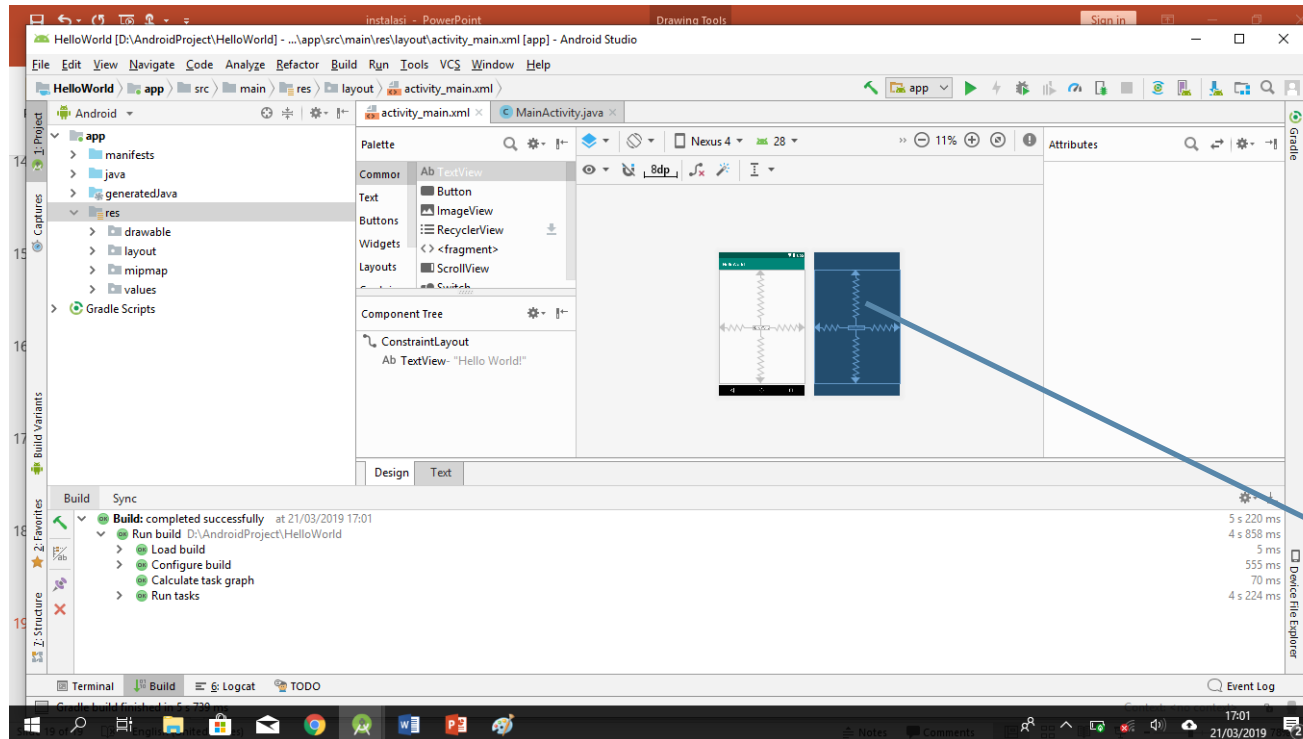
## Pelatihan



- Tunggu sampai proses selesai.
- Proses akan membangun modul-modul yang dibutuhkan oleh project termasuk memperbaharui dan mendownload beberapa gradle jika diperlukan

## Praktek Menambahkan Komponen Tombol (3.7)

## Pelatihan



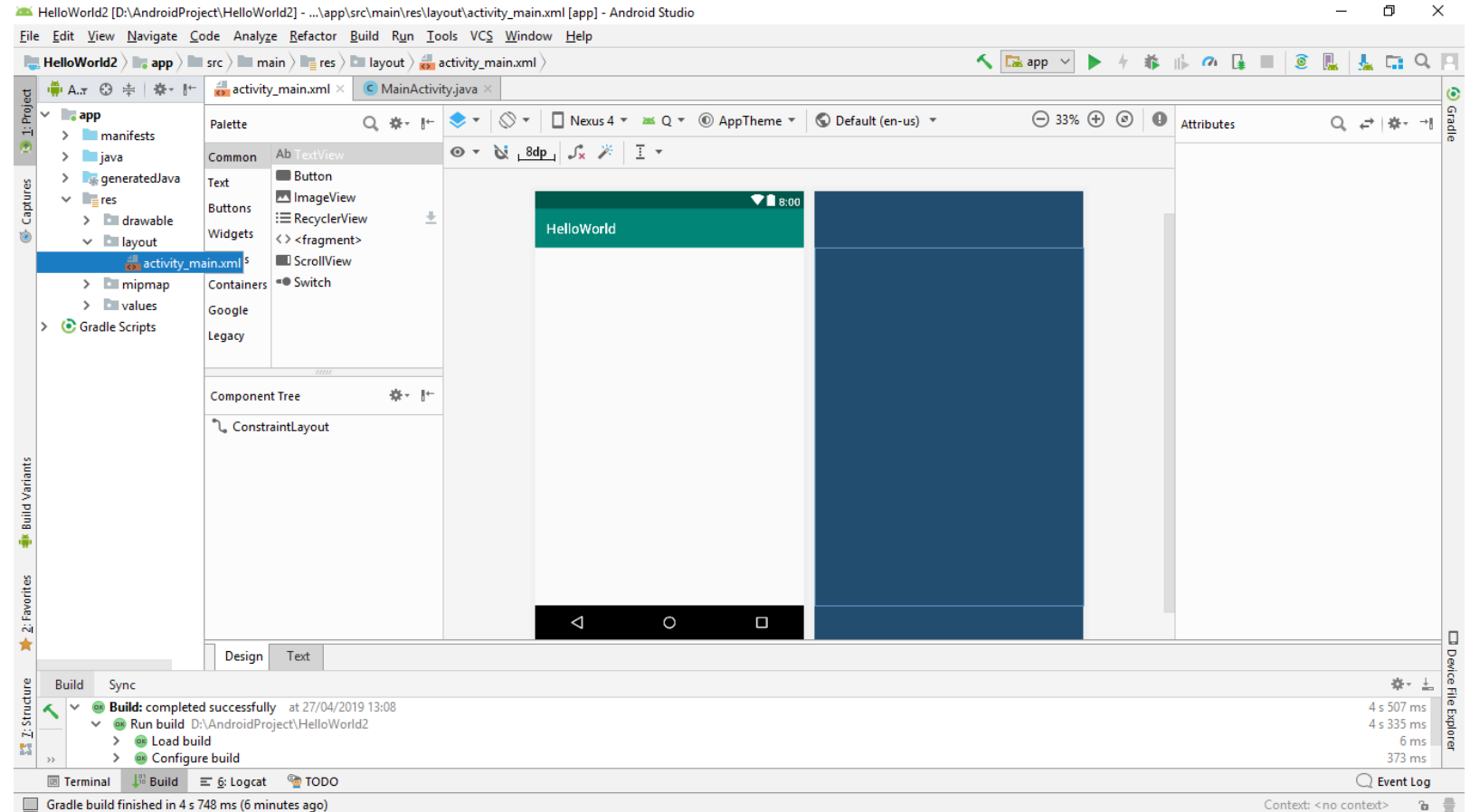
- Jika proses pembuatan project berhasil, maka akan muncul tampilan layout project seperti pada gambar (Klik pada tab **activity\_main.xml**)

Pada bagian tengah dari tampilan akan terdapat sebuah text view dengan tulisan “Hello World”

## Praktek Menambahkan Komponen Tombol (3.8)

## Pelatihan

- Hapus text dengan tulisan “Hello World” dengan cara mengklik pada komponen text tersebut kemudian tekan tombol delete pada keyboard
- Maka Tampilan akan menjadi seperti berikut



## Praktek Menambahkan Komponen Tombol (3.9)

## Pelatihan

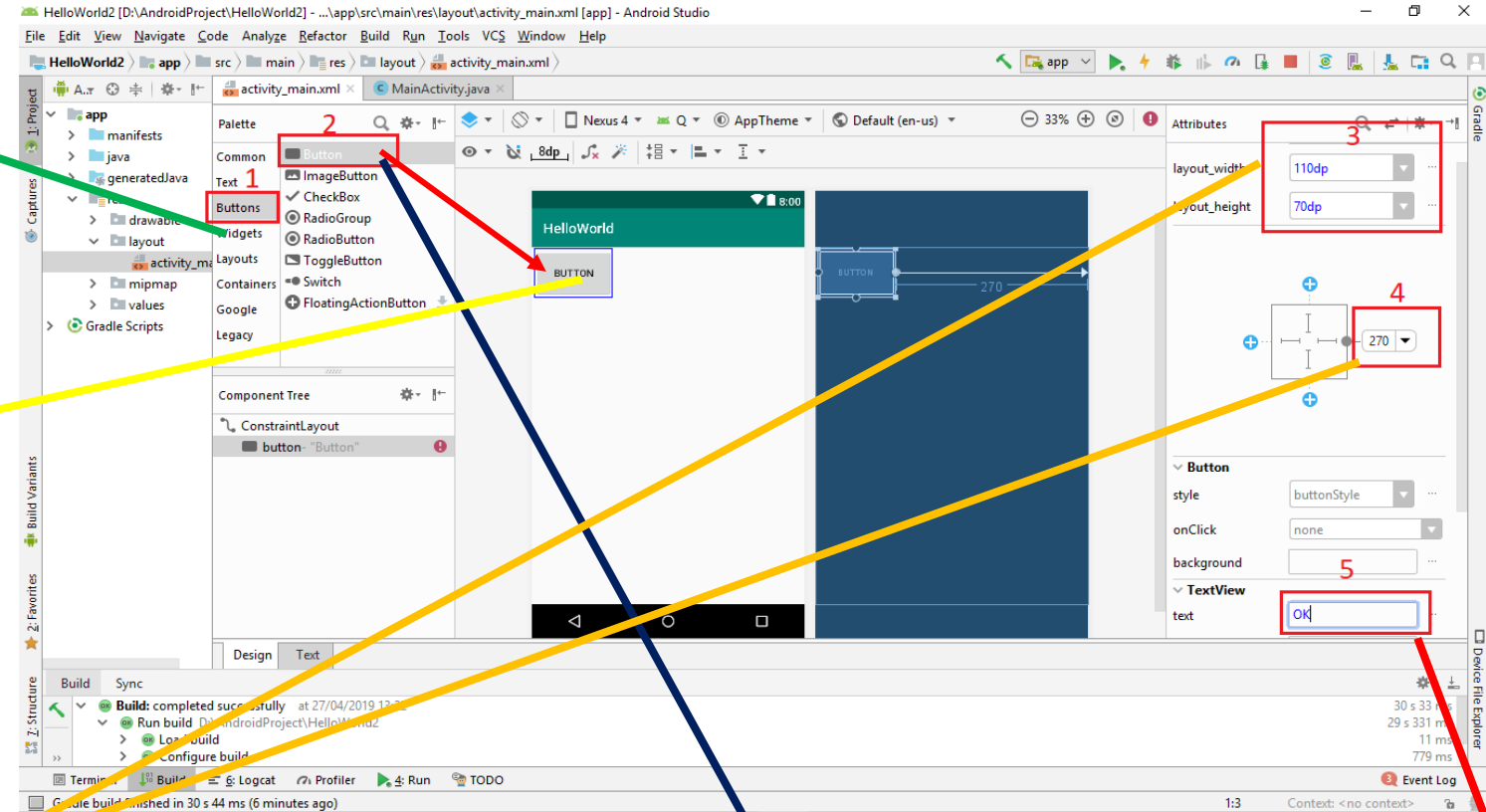
Pada bagian komponen pilih Buttons (no 1)

Klik button tersebut lalu Tarik ke kotak tampilan Hello World, setelah drag kemudian lepas tombol mouse, maka button akan terpasang pada tempat kita menaruh cursor mouse

Untuk mengubah ukuran dapat dilakukan dengan mengklik pada tombol yang dipasang lalu seret kekanan dan kebawah, atau dengan mengubah value pada property (bagian 3 dan 4)

Kemudian pilih Button (no 2) pada komponen palette

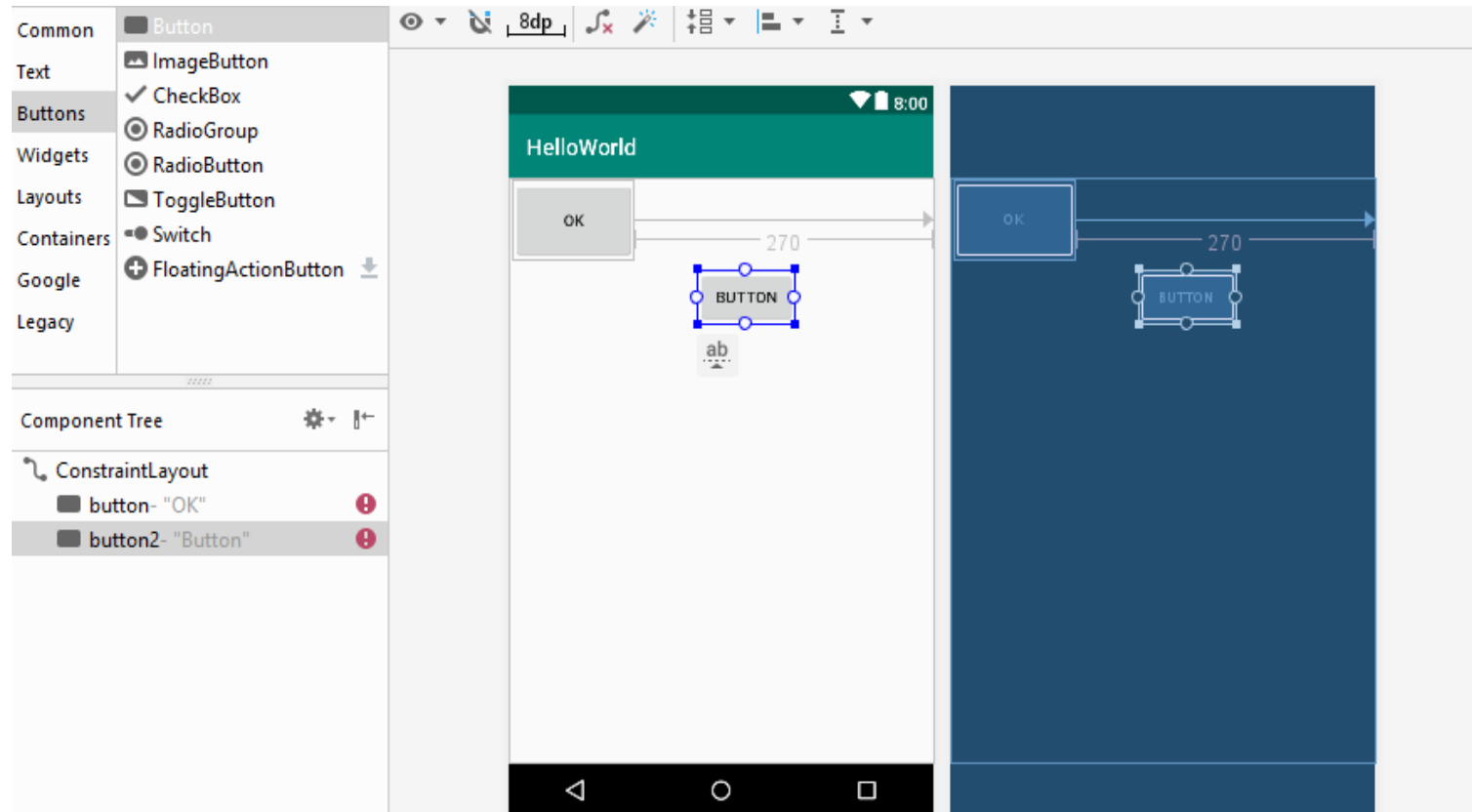
Untuk mengubah text yang tampil pada button dapat dilakukan dengan mengetikkan pada kolom text (no 5)



## Praktek Menambahkan Komponen Tombol (3.10)

## Pelatihan

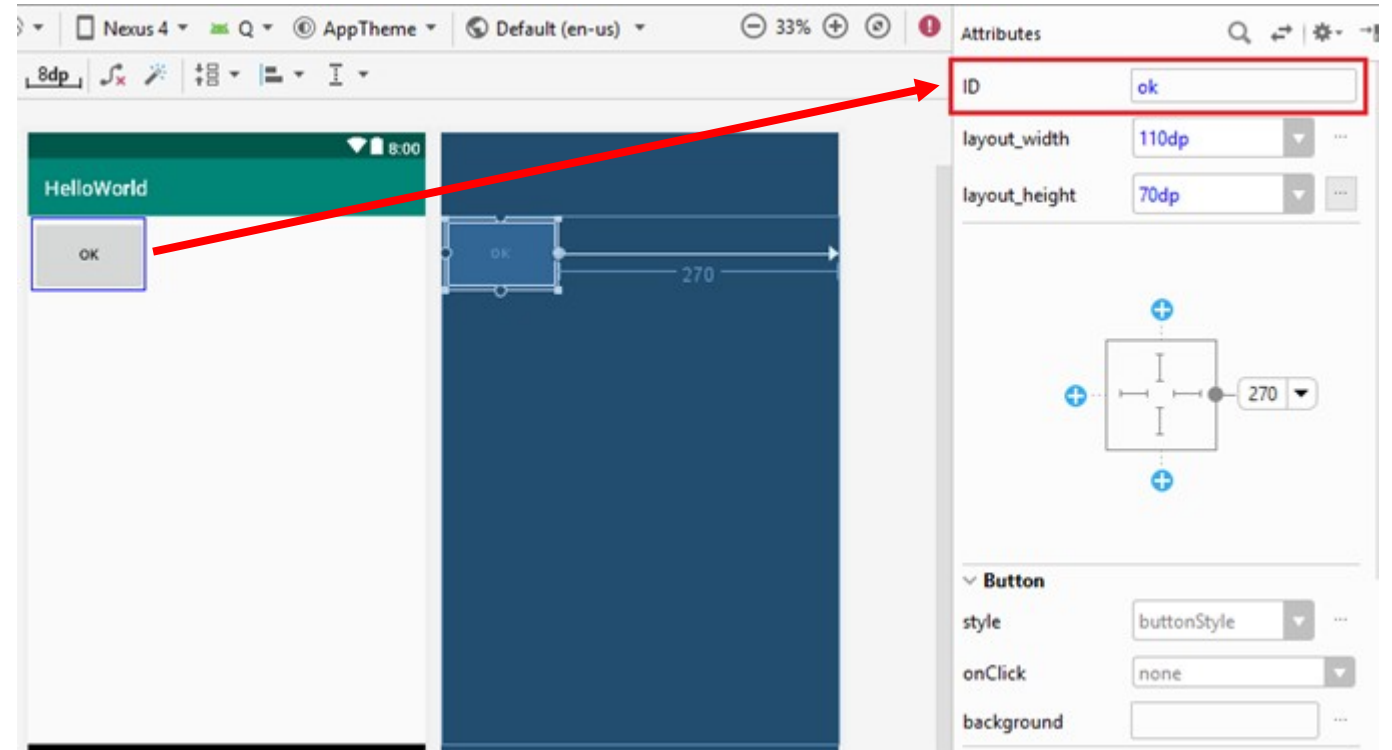
- Untuk menambahkan tombol lain dapat dilakukan dengan cara mengulangi langkah sebelumnya,
- Tombol dapat diletakkan dimana saja karena menggunakan layout relative dan absolute



## Memberi perintah / aksi pada tombol (3.11)

## Pelatihan

- Ubah `id` dari tombol yang akan kita beri aksi
- ID ini sangat penting untuk mengingat tombol yang akan kita beri perintah, karena pada classnya, tombol ini akan dikenali melalui ID nya
- Ubahlah `id` dari button OK dengan mengetikkan `ok` pada kotak ID





## Memberi perintah / aksi pada tombol (3.12)

## Pelatihan

Untuk melihat kode XML dari tampilan layout dapat dilakukan dengan mengklik pada tab text di bagian bawah dari design view

Catatan :

Jika kita telah menguasai XML maka tampilan boleh dikoreksi dengan memodifikasi xml tersebut

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     tools:context=".MainActivity">
8
9     <Button
10         android:id="@+id/ok"
11         android:layout_width="110dp"
12         android:layout_height="70dp"
13         android:layout_marginEnd="270dp"
14         android:layout_marginRight="270dp"
15         android:text="OK"
16         app:layout_constraintEnd_toEndOf="parent"
17         tools:layout_editor_absoluteY="2dp" />
18
19     <Button
20         android:id="@+id/button2"
21         android:layout_width="wrap_content"
22         android:layout_height="wrap_content"
23         android:text="Button"
24         tools:layout_editor_absoluteX="171dp"
25         tools:layout_editor_absoluteY="80dp" />
26 </android.support.constraint.ConstraintLayout>
```

Id dari button paling atas akan berubah menjadi  
`android:id="@+id/ok"`

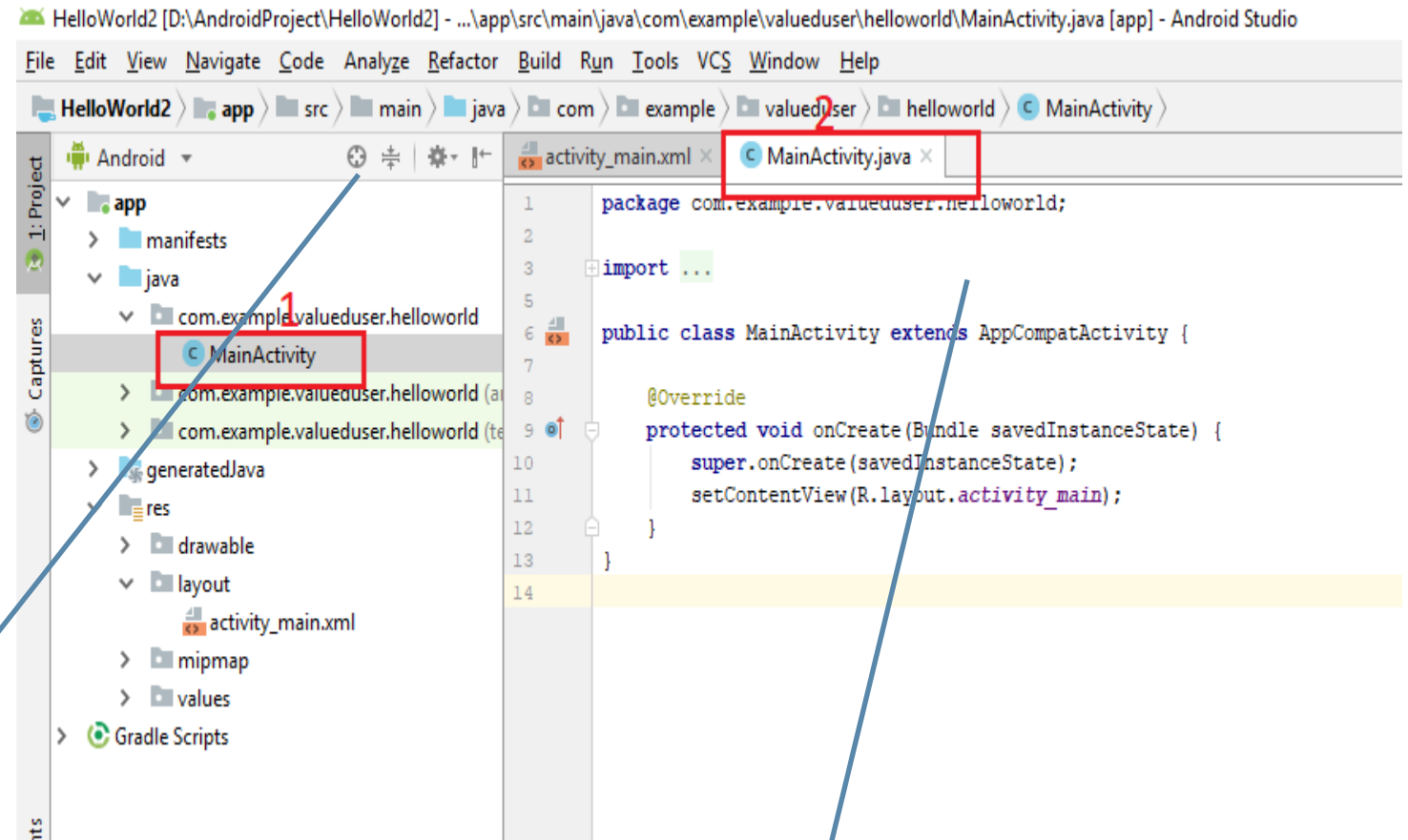


## Memberi perintah / aksi pada tombol (3.13)

## Pelatihan

- Langkah selanjutnya adalah memodifikasi kode program pada class main (yang ditulis dalam Bahasa pemrograman java)
- Klik pada tree view MainActivity (1) atau pada tab MainActivity (2) maka jendela akan pindah ke kode program java

MainActivity  
Class



Code program  
java

## Memberi perintah / aksi pada tombol (3.14)

## Pelatihan

```
package com.example.valueduser.helloworld;
import android.graphics.Color;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        final Button btnOk = findViewById(R.id.ok);
        btnOk.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                btnOk.setBackgroundColor(Color.RED);
            }
        });
    }
}
```

Membuat variable btnOk untuk mewakili button OK pada view

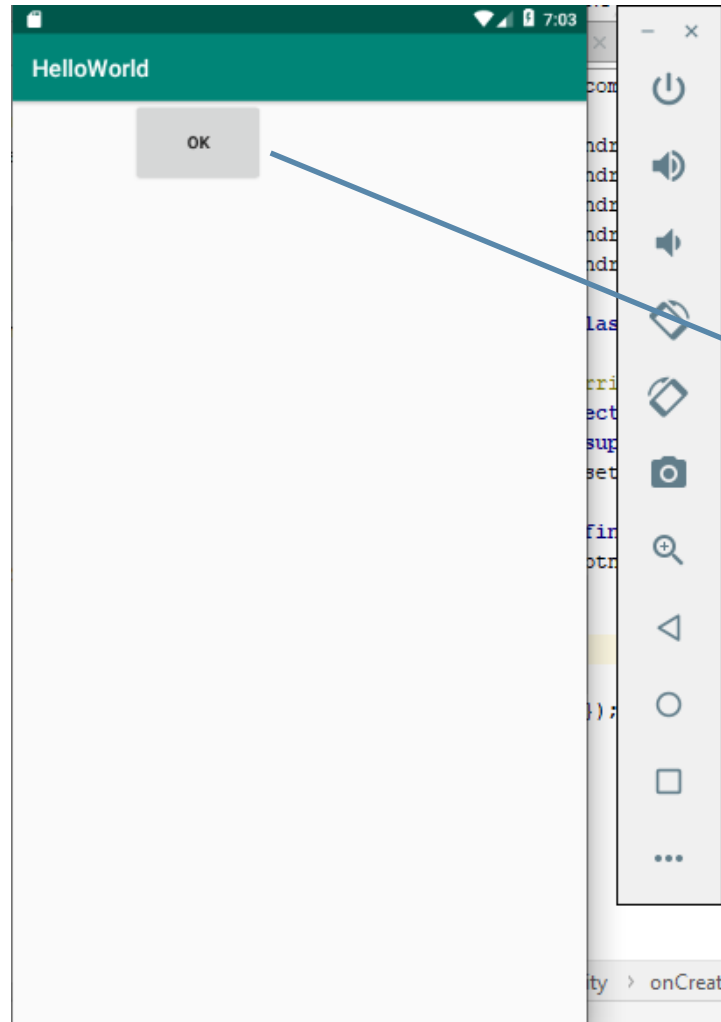
Perintah  
findViewById(R.id.ok)  
berfungsi untuk mencari dan mengenali button dengan id ok yang dibuat pada view xml

Memberi aksi pada tombol Ok ketika di click oleh user

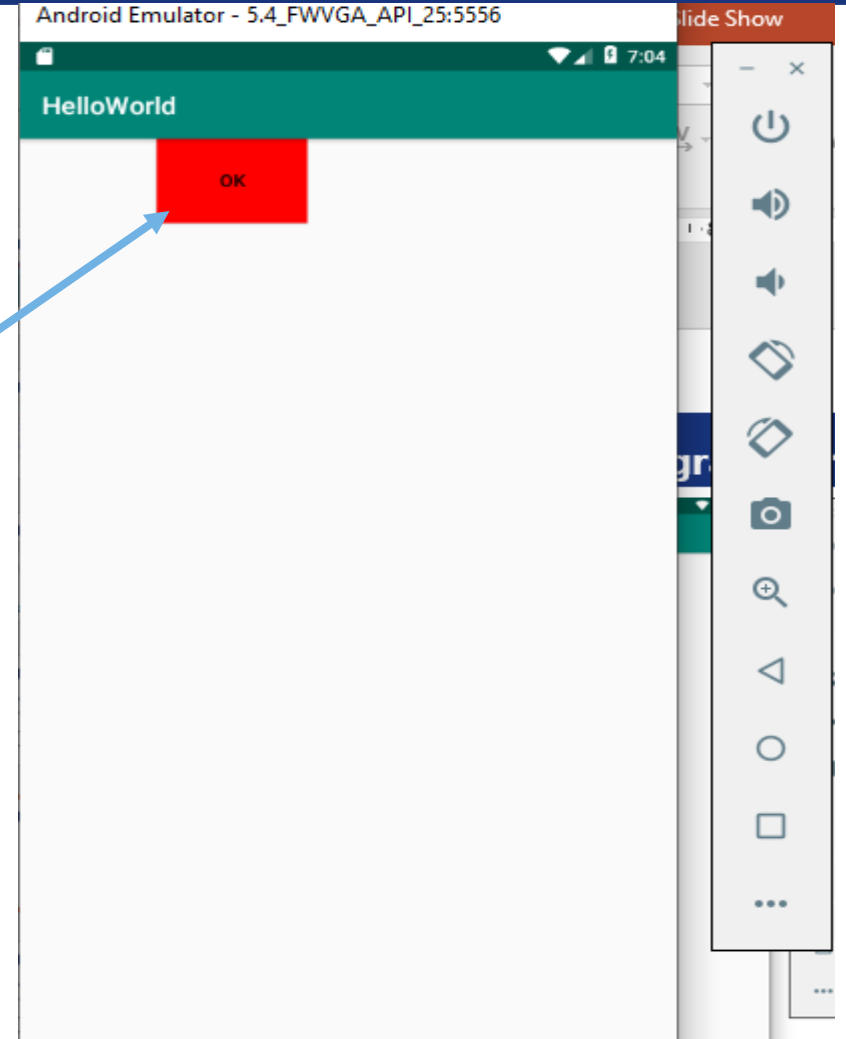
Mengganti warna background button menjadi merah saat di-klik

## Jalankan Program (3.15)

## Pelatihan

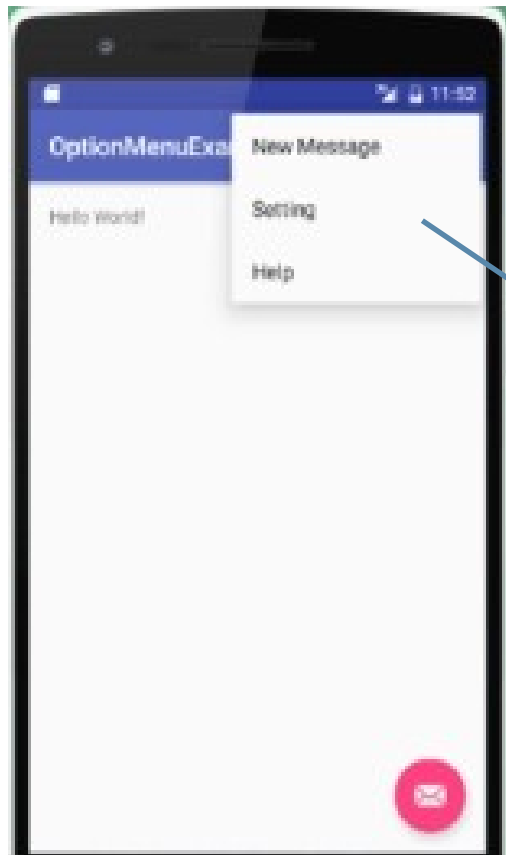


Klik tombol OK  
maka tombol akan  
berubah menjadi  
merah



## 4. Menambahkan Komponen Menu Pelatihan

Menu atau option menu biasanya ditampilkan pada sisi kanan aplikasi android dengan lambing garis atau titik sebanyak tiga buah. Ketika di klik maka akan muncul menu drop down dengan beberapa list menu



Menu / Option Menu

# Menambahkan Komponen Menu (4.1)

## Pelatihan

Buatlah sebuah project baru dengan nama “ProjectMenu”

Create New Project

Create Android Project

Application name  
ProjectMenu

Company domain  
valueduser.example.com

Project location  
D:\AndroidProject\ProjectMenu

Package name  
com.example.valueduser.projectmenu

☐ Include C++ support  
☐ Include Kotlin support

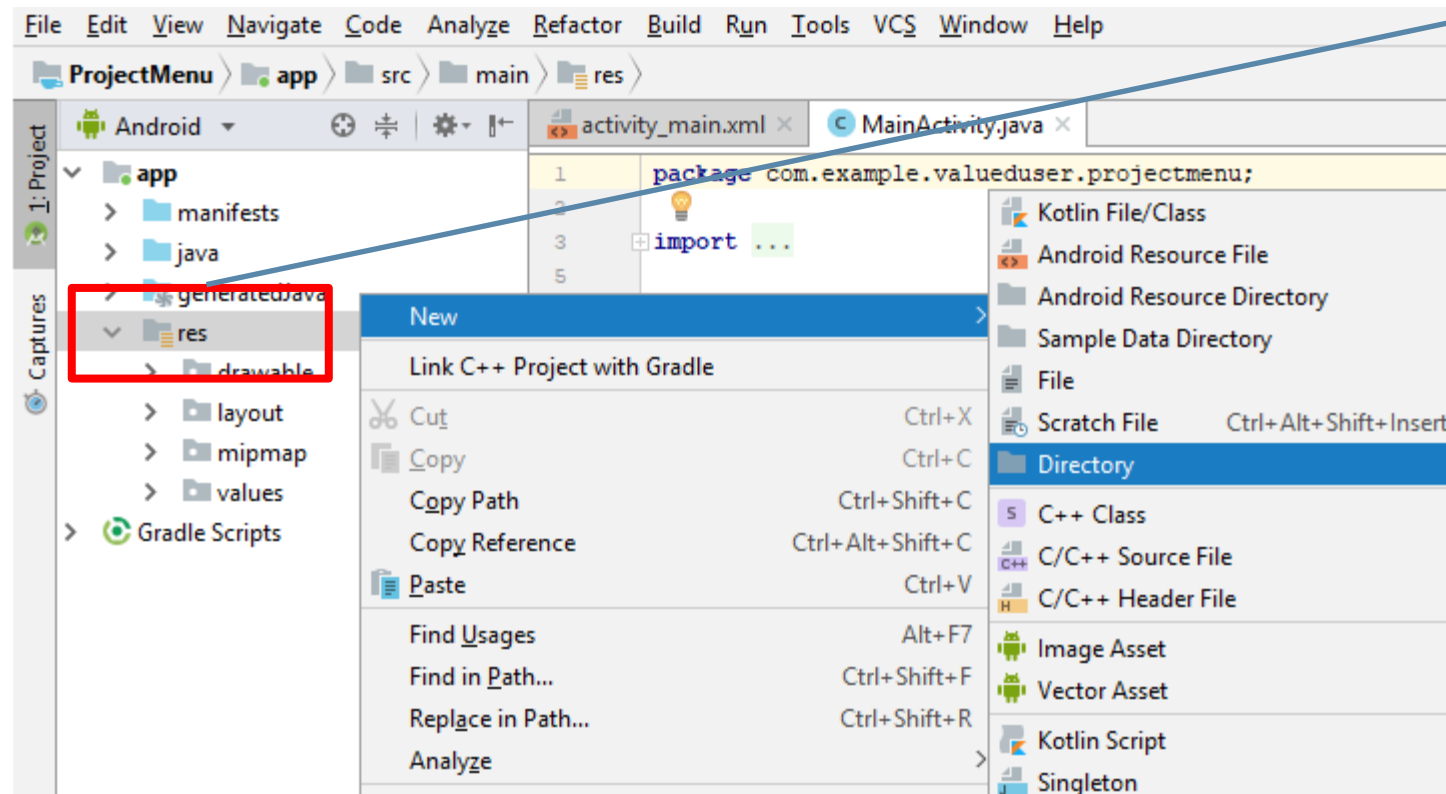
Previous Next Cancel Finish

ProjectMenu

## Menambahkan Komponen Menu (4.2)

## Pelatihan

Buatlah folder menu di dalam folder “res”



Klik kanan pada folder res

Pilih menu New

Kemudian pilih Directory

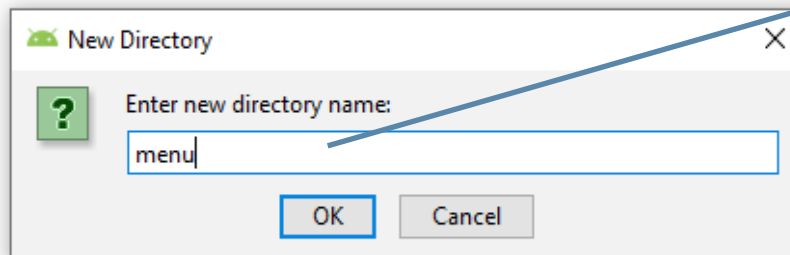
Kemudian akan muncul  
jendela new Directory

## Menambahkan Komponen Menu (4.3)

## Pelatihan

Buatlah folder menu

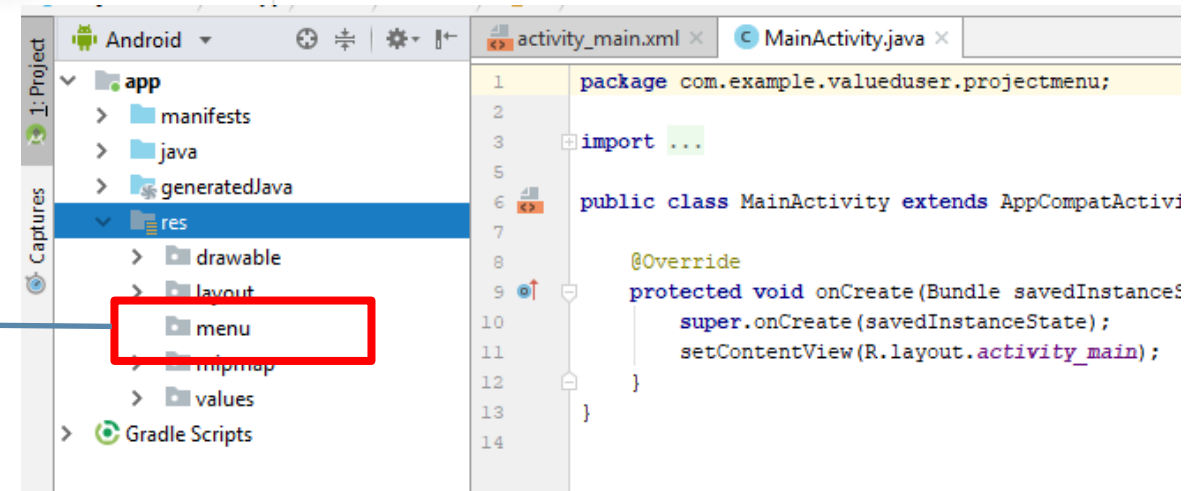
```
setContentView(R.layout.activity_main);  
}
```



Ketikan **menu** pada kotak  
name

Kemudian klik OK

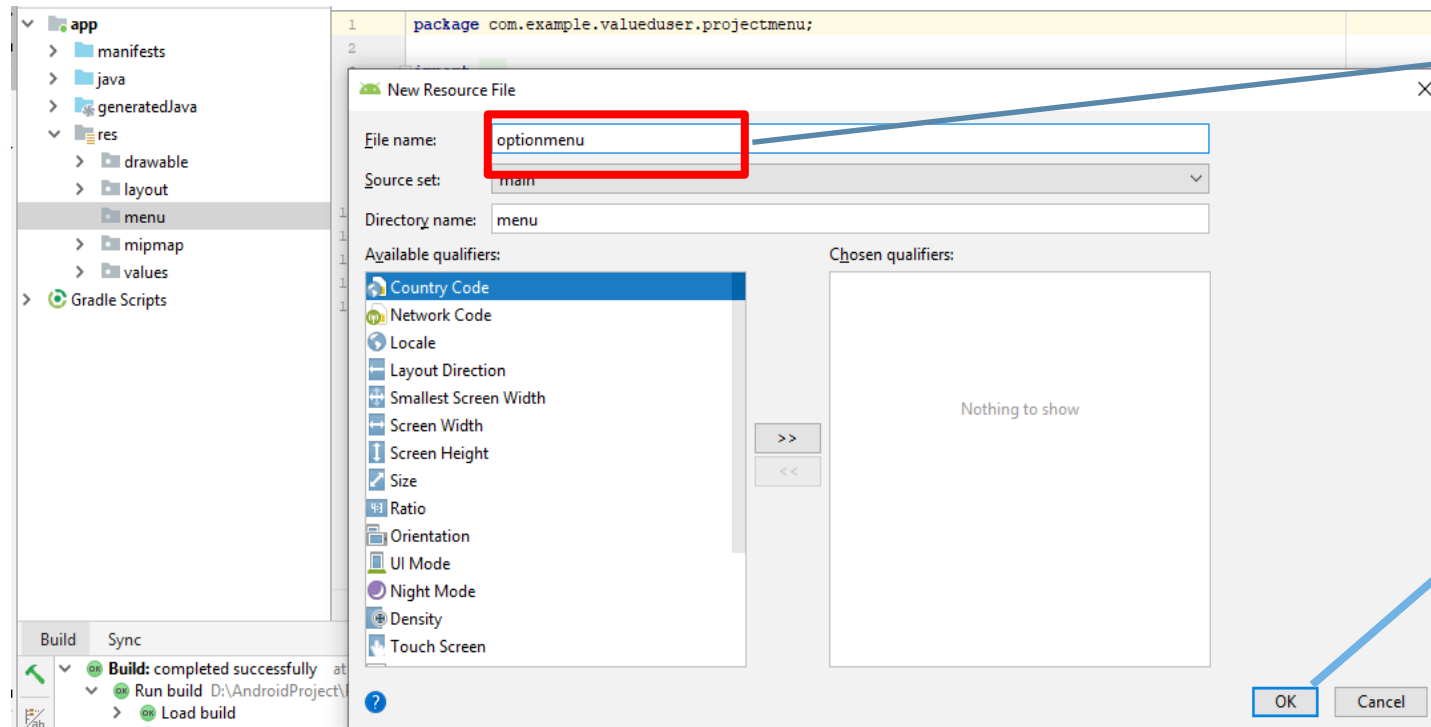
Folder **menu** akan terbentuk



# Menambahkan Komponen Menu (4.4)

## Pelatihan

Buatlah Menu Resource File



Ketikan "optionmenu"  
dengan huruf kecil

Biarkan option yang lain  
seperti apa adanya

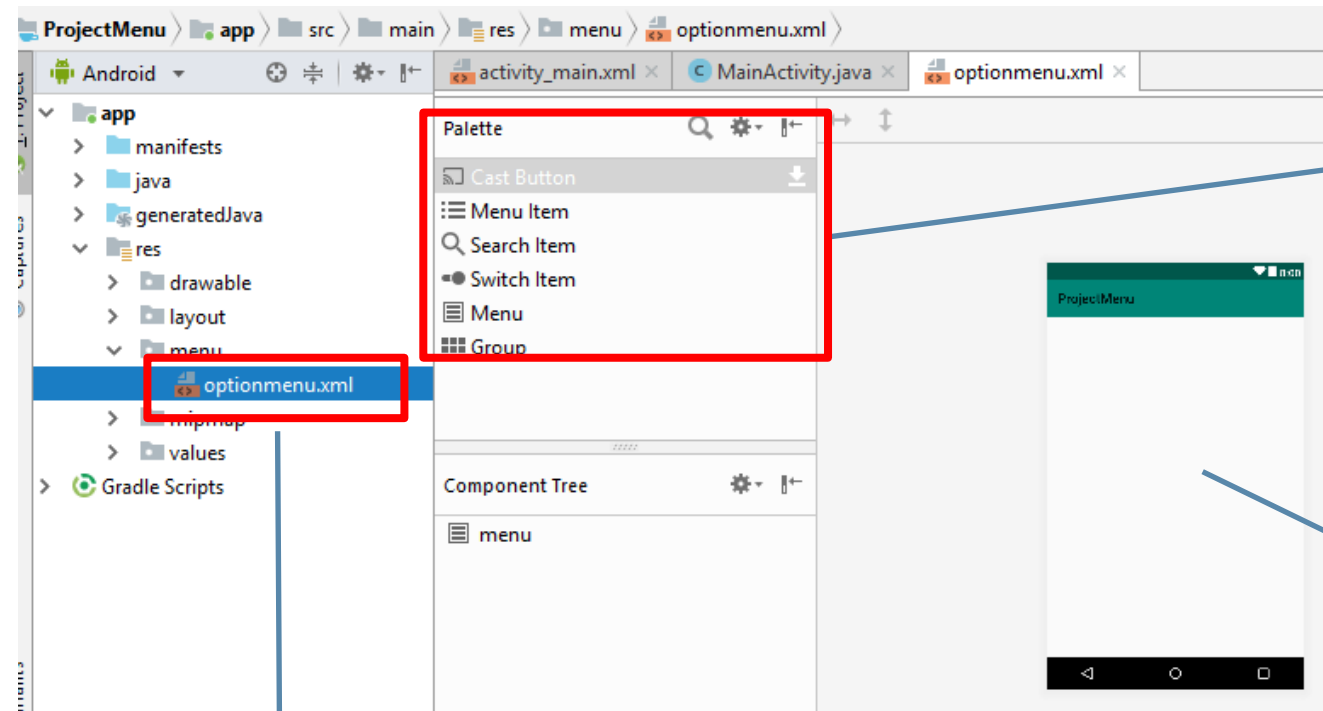
Klik Ok untuk membuat  
resource menu ini



# Menambahkan Komponen Menu (4.5)

## Pelatihan

Buatlah Menu Resource File



Palette menu yang tersedia (palette ini otomatis muncul ketika resource menu ditampilkan)

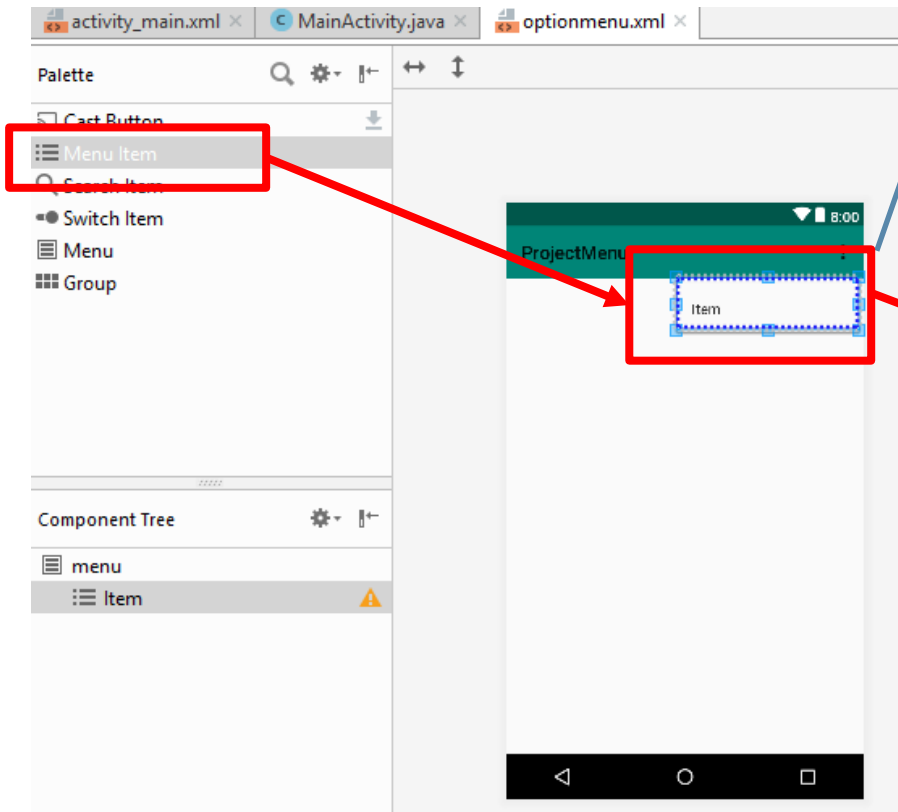
Tampilan desain menu tempat meletakkan menu yang akan dibuat

“optionmenu” yang merupakan resource file yang dibuat

# Menambahkan Komponen Menu (4.6)

## Pelatihan

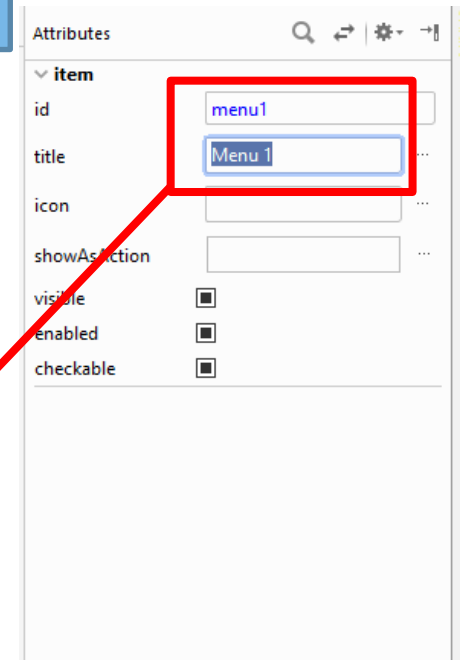
### Menambahkan menu option



Pilih Palette “Menu Item” kemudian drag ke desain tampilan, lalu lepas tombol pada mouse

Menu item akan otomatis muncul pada bagian kanan atas layer desain

Pada attributes, Ubahlah value id menjadi “menu1” dan title menjadi “Menu 1”



# Menambahkan Komponen Menu (4.7)

## Pelatihan

Menambahkan menu option

Dengan cara yang sama, tambahkanlah 3 buah menu item dengan id berurutan dari menu1, menu2, dan menu3

Palette

- Cast Button
- Menu Item**
- Search Item
- Switch Item
- Menu
- Group

Component Tree

- menu
  - menu1
  - menu2
  - menu3**

ProjectMenu

- Menu 1
- Menu 2
- Menu 3

item

- id: menu3
- title: Menu 3
- icon
- showAsAction
- visible
- enabled
- checkable

View all attributes

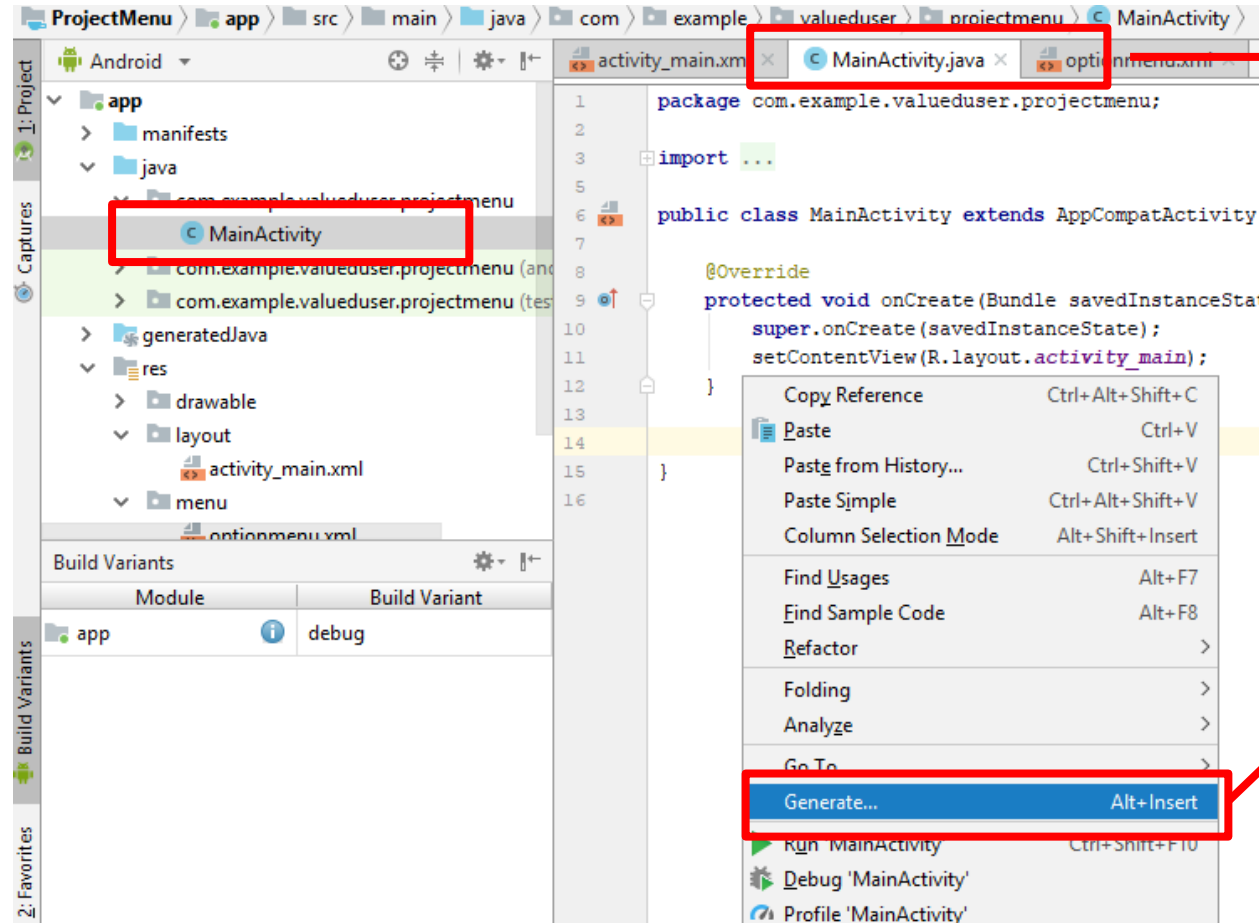
Untuk mengatur penampilan menu pada action bar

Pada component tree akan ditampilkan nama / id dari ke 3 menu yang dibuat

# Menambahkan Komponen Menu (4.8)

## Pelatihan

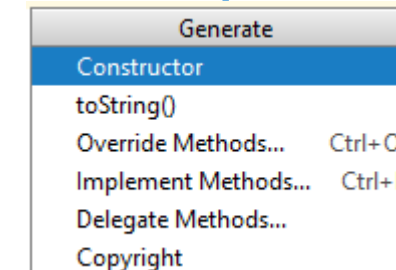
Menampilkan menu option



Klik pada class MainActivity  
untuk pindah ke layer kode  
program

Klik kanan pada bagian yang  
kosong, pada kode program  
(diatas tanda kurung } yang  
paling bawah

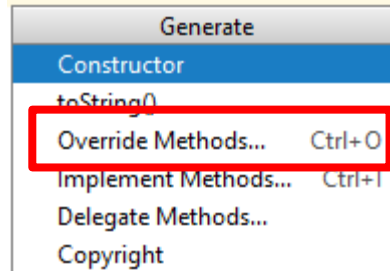
Pilih menu Generate...



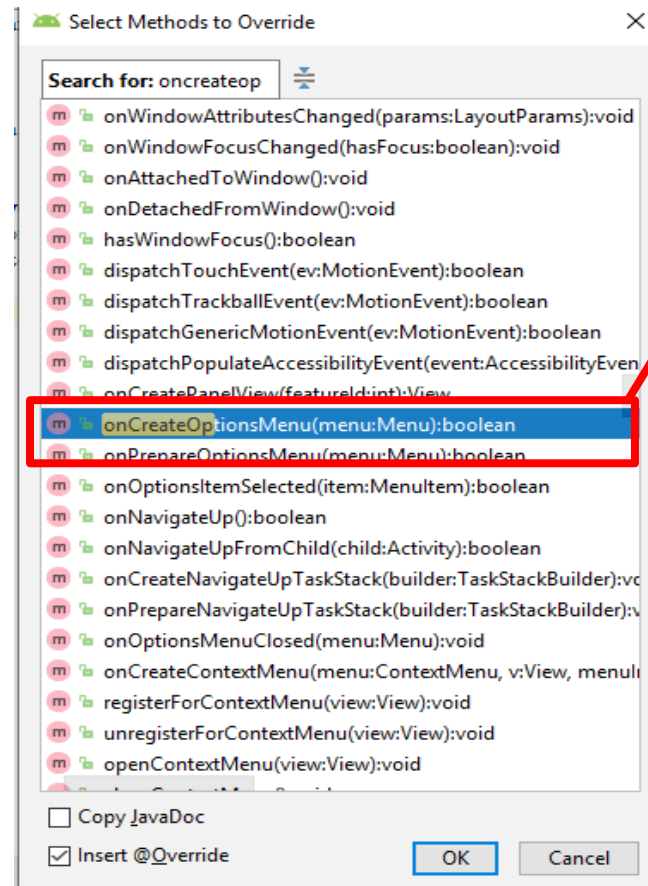
# Menambahkan Komponen Menu (4.9)

## Pelatihan

Menampilkan menu option



Pilih menu Override Methods..



Uncul menu select method to  
override

Cari method  
"onCreateOptionsMenu"  
dengan mengetikan pada  
keyboard

Seteah ketemu, pilih OK

```
@Override  
public boolean  
onCreateOptionsMenu(Menu menu) {  
    return  
    super.onCreateOptionsMenu(menu);  
}
```

# Menambahkan Komponen Menu (4.10)

## Pelatihan

Menampilkan menu option

```
1 package com.example.valueduser.projectmenu;
2
3 import ...
4
5
6
7 public class MainActivity extends AppCompatActivity {
8
9     @Override
10    protected void onCreate(Bundle savedInstanceState) {
11        super.onCreate(savedInstanceState);
12        setContentView(R.layout.activity_main);
13    }
14
15    @Override
16    public boolean onCreateOptionsMenu(Menu menu) {
17        return super.onCreateOptionsMenu(menu);
18    }
19 }
20
```

Rubahlah bagian kode tersebut menjadi

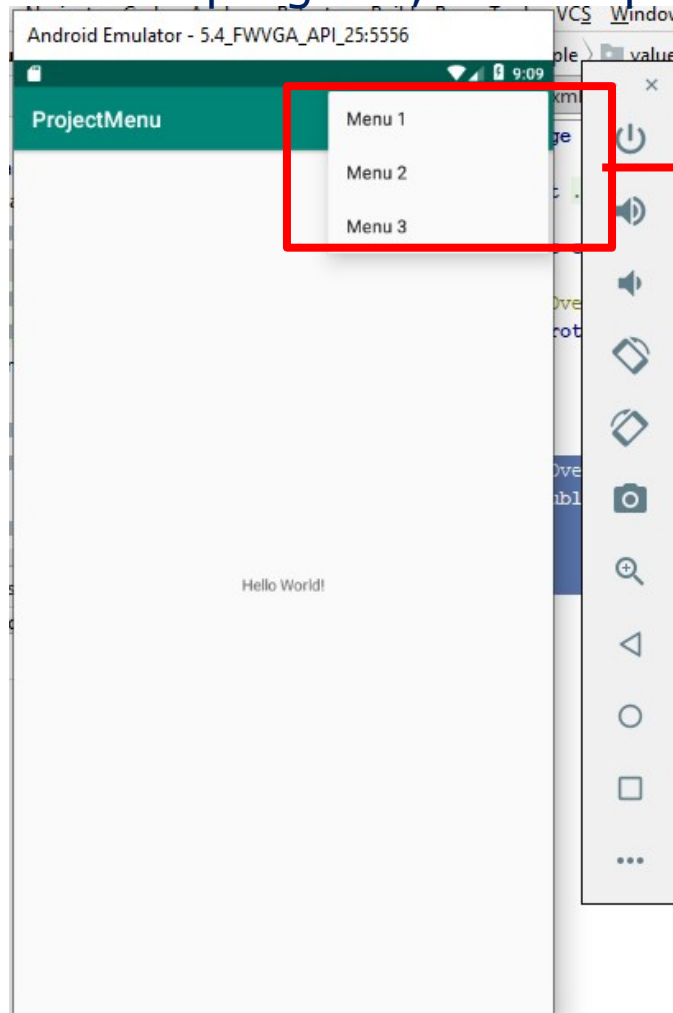
```
@Override
public boolean onCreateOptionsMenu (Menu
menu) {
    MenuInflater inflater =
getMenuInflater();
    inflater.inflate(R.menu.optionmenu,
menu);
    return true;
}
```

**MenuInflater** berfungsi untuk menampilkan menu yang sudah dibuat pada **optionmenu.xml**

## Menambahkan Komponen Menu (4.11)

## Pelatihan

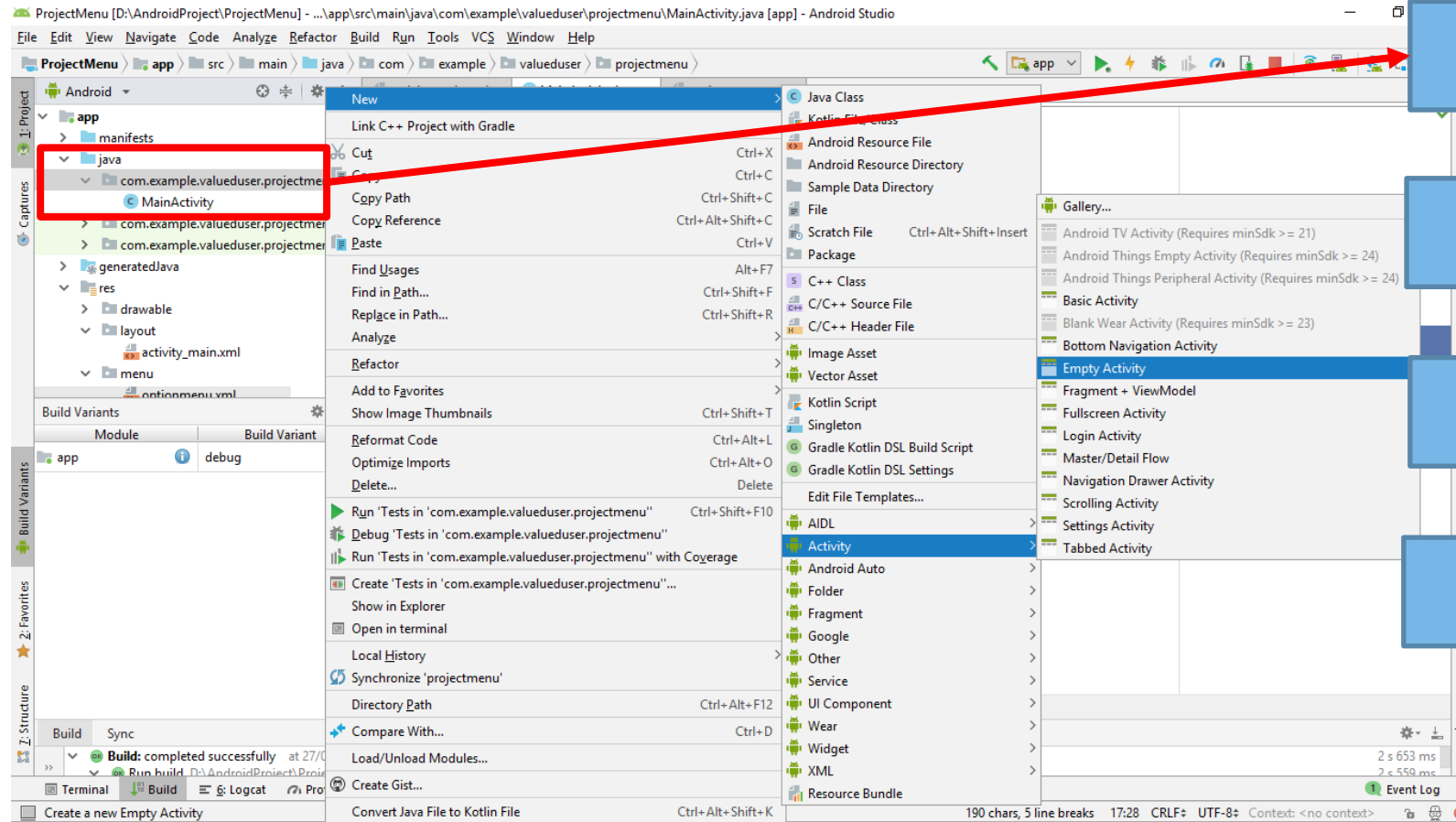
Jalankan program, maka option menu akan muncul pada pojok kanan atas aplikasi



Option menu yang telah  
berhasil dibuat

## 5. Memberi Perintah atau Aksi Pada Tombol dan Menu Pelatihan

Tambahkan 3 buah activity baru



Klik kanan pada folder / package  
"java" = "com example"

Pilih menu New

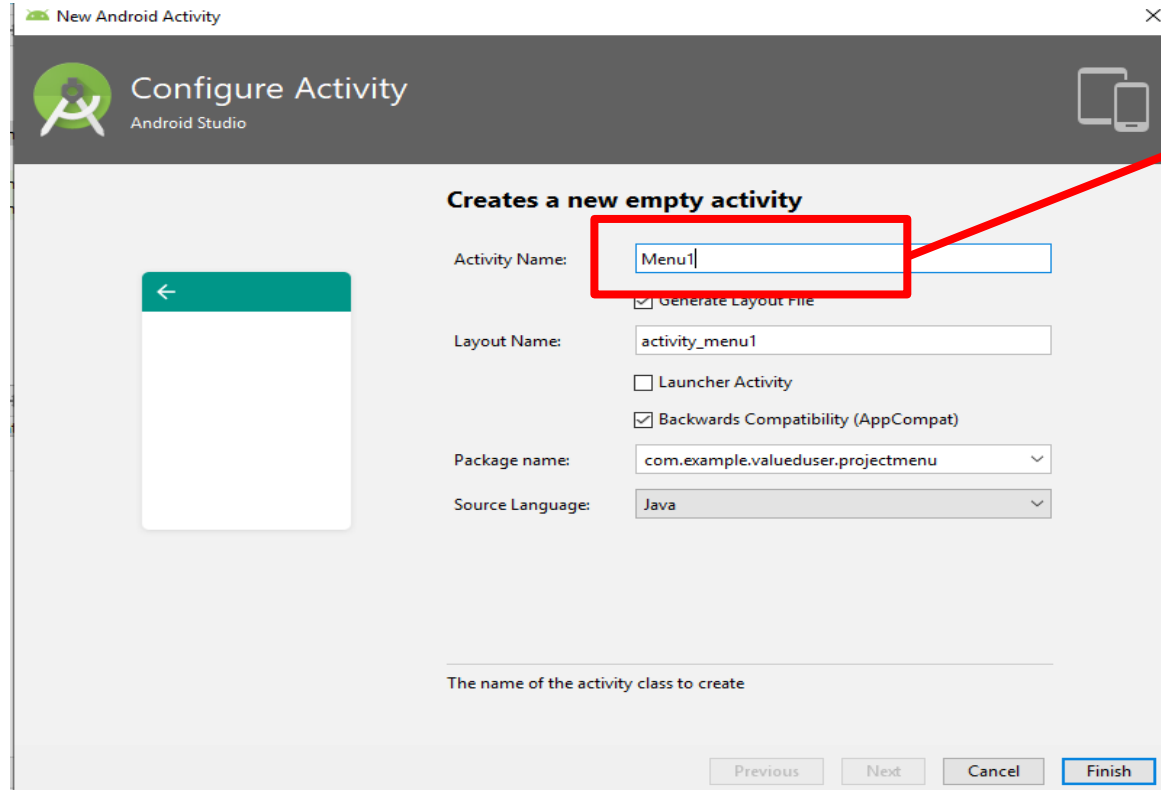
Pilih "Activity"

Pilih "Empty Activity"



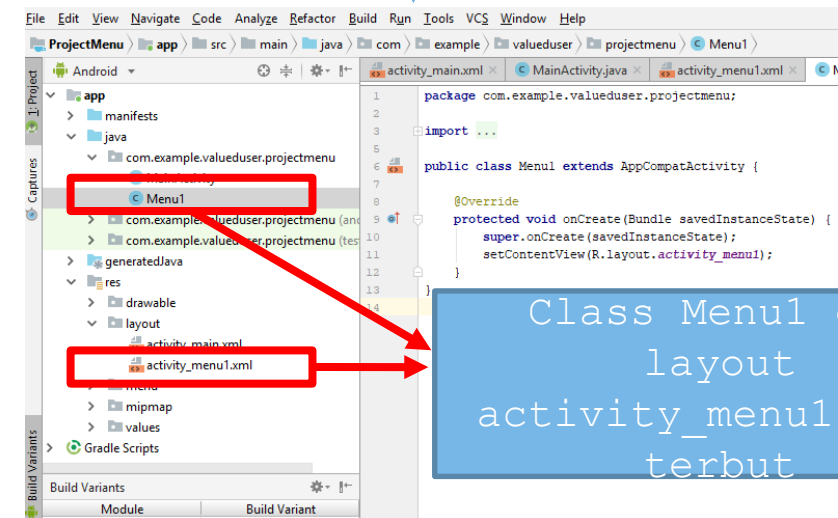
# Memberi Perintah atau Aksi Pada Tombol dan Menu (5.1) Pelatihan

## Tambahkan 3 buah activity baru



Beri Nama Menu1

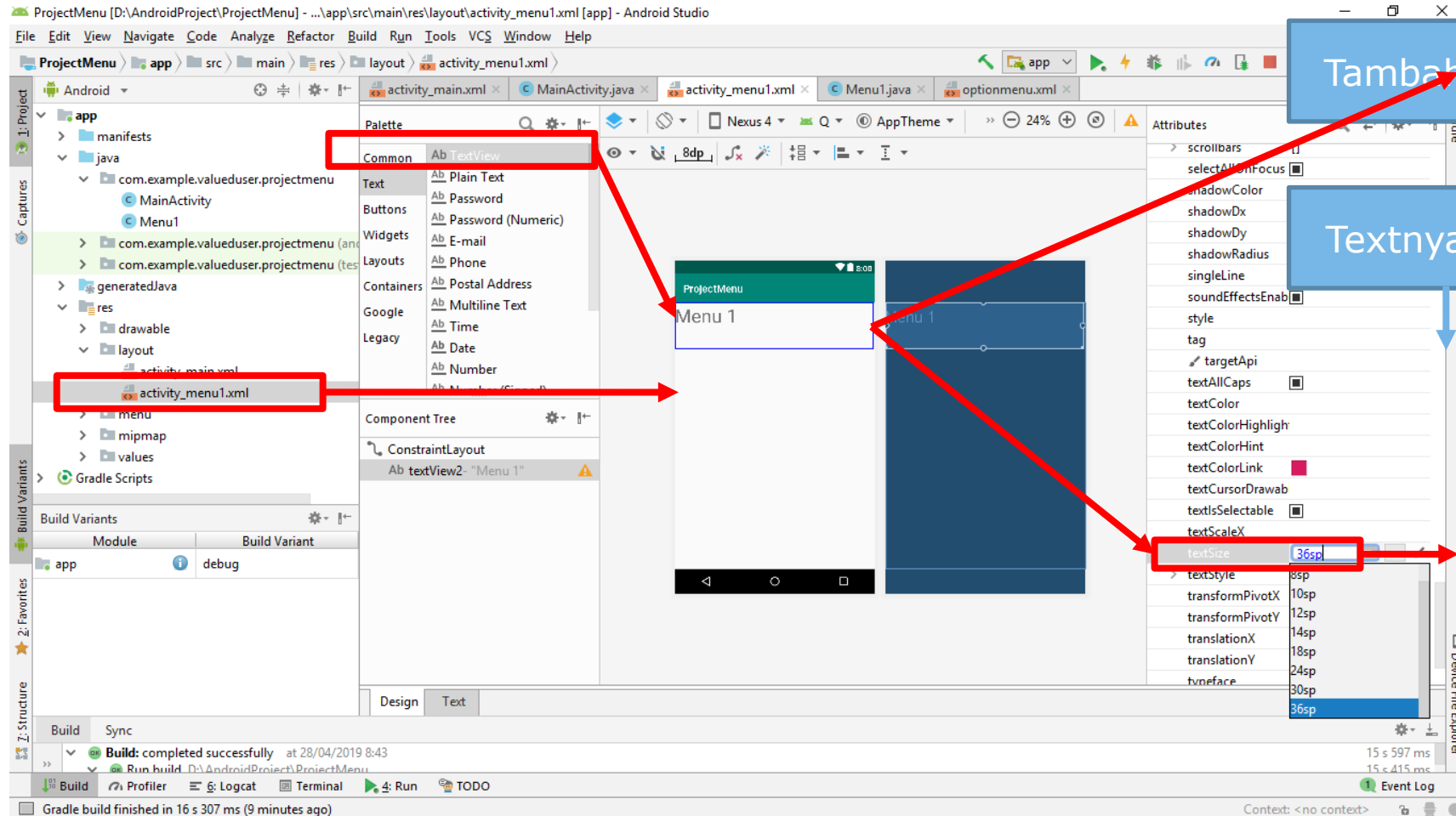
Alu pilih "Finish"



Class Menu1 dan  
layout  
activity\_menu1 akan  
terbut

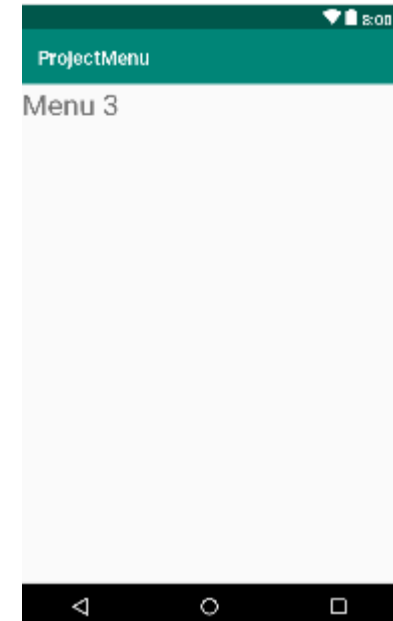
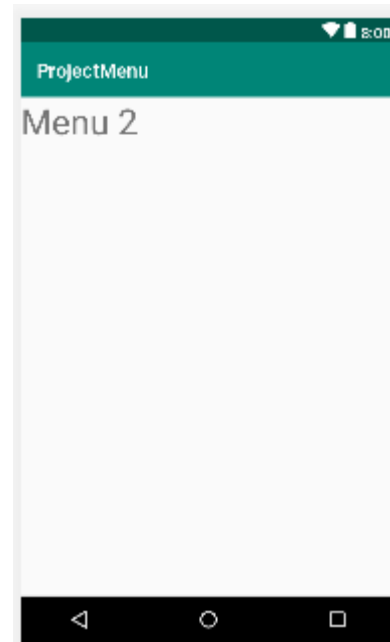
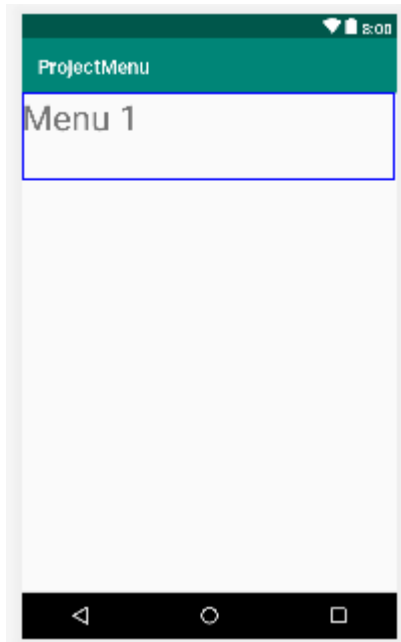
## Memberi Perintah atau Aksi Pada Tombol dan Menu (5.2) Pelatihan

Tambahkan text dengan tulisan "Menu 1" dan ukuran font 38



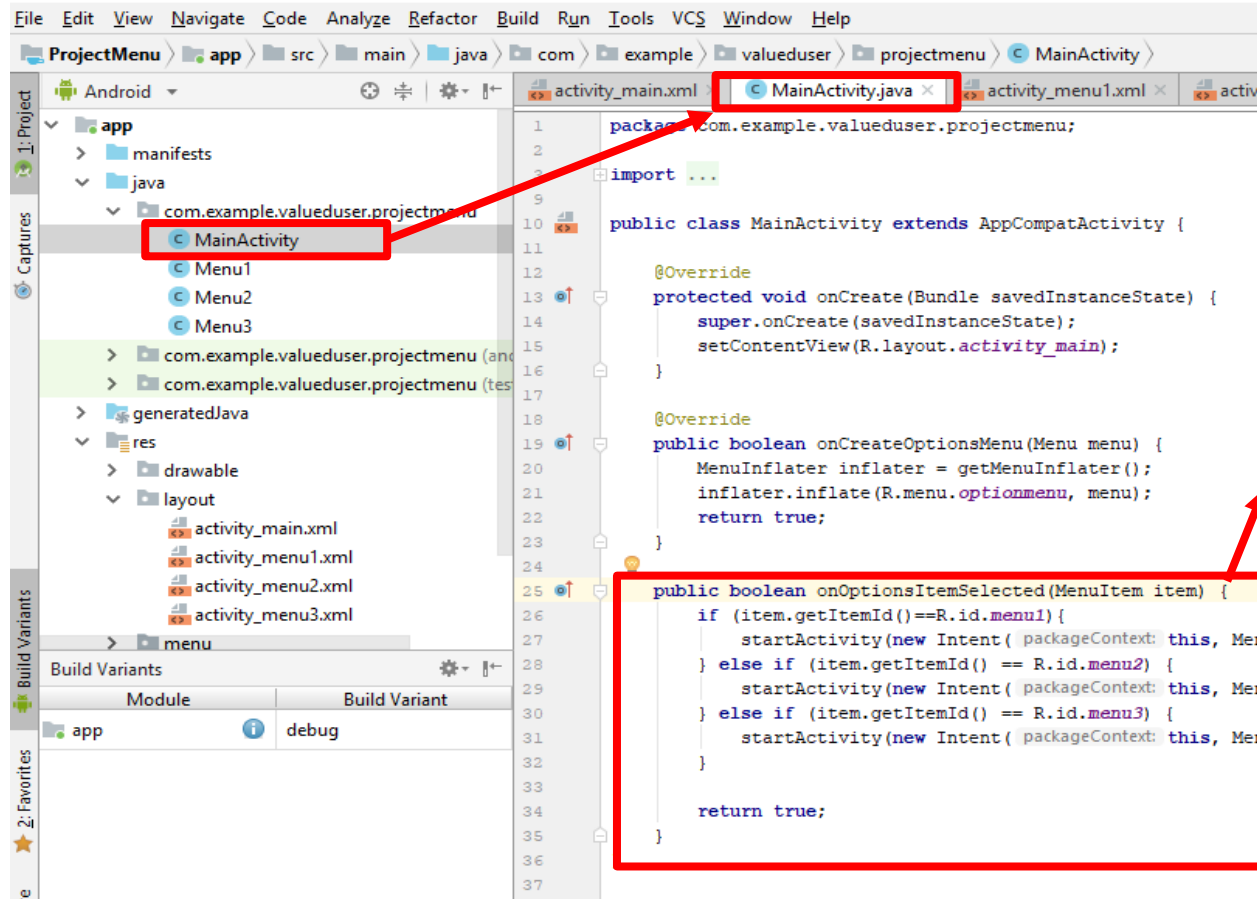
## Memberi Perintah atau Aksi Pada Tombol dan Menu (5.3) Pelatihan

Dengan langkah yang sama, tambahkan 2 buah activity baru, sehingga akan ada 3 buah activity dengan nama "Menu1", "Menu2" dan "Menu3"



## Memberi Perintah atau Aksi Pada Tombol dan Menu (5.4) Pelatihan

Untuk memberi perintah pada option menu, bukalah MainActivity.class kemudian ubahlah kode programnya menjadi



```
public boolean onOptionsItemSelected(MenuItem item) {
    if (item.getItemId() == R.id.menu1) {
        startActivity(new Intent(this, Menu1.class));
    } else if (item.getItemId() == R.id.menu2) {
        startActivity(new Intent(this, Menu2.class));
    } else if (item.getItemId() == R.id.menu3) {
        startActivity(new Intent(this, Menu3.class));
    }

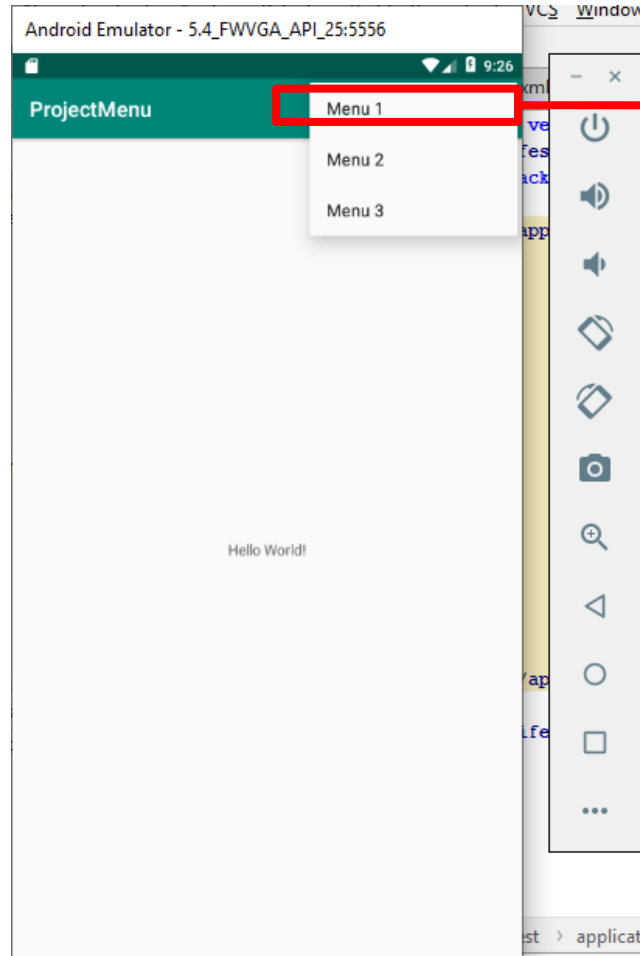
    return true;
}
```

**OptionsItemSelected** Berfungsi untuk memberi perintah ke menuoption yang telah dibuat

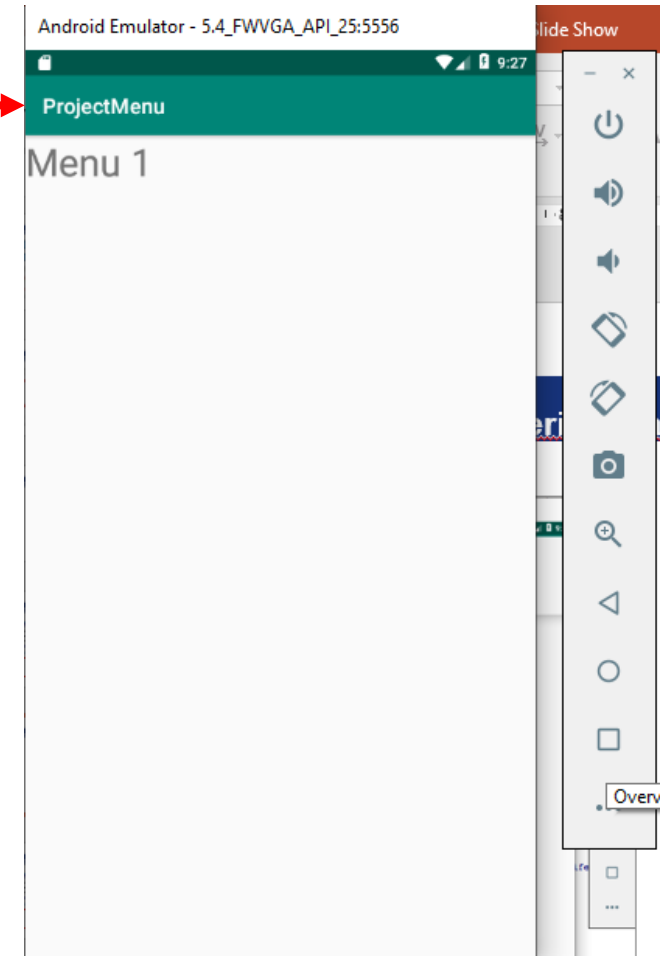
**getItemId** berfungsi untuk mendapatkan id dari menu item yang dibuat pada **optionmenu.xml**

**Intent** untuk menuju ke activity dari class yang akan ditampilkan

# Memberi Perintah atau Aksi Pada Tombol dan Menu (5.5) Pelatihan Jalankan Program



pilih maka layer akan  
berpindah ke activity



## Memberi Perintah atau Aksi Pada Tombol dan Menu (5.6) Pelatihan

Memberi caption/tittle pada setiap menu yang ditampilkan, tambahkanlah label pada "AndroidManifest.xml" tepat disebelah menu1, menu2, dan menu3 menjadi seperti berikut:

The screenshot shows the Android Studio interface with the `AndroidManifest.xml` file open. The file contains the following code:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.valueduser.projectmenu">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".Menu1" android:label="Latihan Menu 1"/>
        <activity android:name=".Menu2" android:label="Latihan Menu 2"/>
        <activity android:name=".Menu3" android:label="Latihan Menu 3"/>
    </application>
</manifest>
```

Annotations in the image include:

- A red box around the `AndroidManifest.xml` file in the Project view.
- A red box around the `<activity android:name=".Menu1" android:label="Latihan Menu 1"/>` line in the manifest file.
- A blue box containing the code snippets for `.Menu1`, `.Menu2`, and `.Menu3` with their respective labels.
- A blue box with the text: "Jalankan Program dan caption akan berubah ketika activity masing-masing menu ditampilkan".
- A blue box with the text: "Caption yang berubah menjadi 'Latihan Menu 1'".

The Android emulator at the bottom shows the app running with the caption "Latihan Menu 1" displayed on the screen.

## TUGAS

## Pelatihan

Buatlah sebuah game puzzle Huruf sederhana yang berisikan tombol dan menu seperti tampilan berikut





## Kesimpulan

## Pelatihan

### Kesimpulan

1. Peserta dapat memasang / membuat komponen tombol / button pada tampilan sesuai dengan kebutuhan
2. Peserta dapat memasang / membuat menu pada tampilan sesuai dengan kebutuhan
3. Peserta mampu merubah ukuran tombol dan menu sesuai dengan kebutuhan sistem
4. Peserta mampu memberikan perintah pada tombol / button dan menu yang dibuat



## < Topik\_Silabus >

## Pelatihan

### Referensi:

1. <referensi 1>
2. <referensi 2>

# Tim Penyusun:

- Alif Akbar Fitrawan, S.Pd, M. Kom (Politeknik Negeri Banyuwangi);
- Anwar, S.Si, MCs. (Politeknik Negeri Lhokseumawe);
- Eddo Fajar Nugroho (BPPTIK Cikarang);
- Eddy Tungadi, S.T., M.T. (Politeknik Negeri Ujung Pandang);
- Fitri Wibowo (Politeknik Negeri Pontianak);
- Ghifari Munawar (Politeknik Negeri Bandung);
- Hetty Meileni, S.Kom., M.T. (Politeknik Negeri Sriwijaya) ;
- I Wayan Candra Winetra, S.Kom., M.Kom (Politeknik Negeri Bali) ;
- Irkham Huda (Vokasi UGM) ;
- Josseano Amakora Koli Parera, S.Kom., M.T. (Politeknik Negeri Ambon) ;
- I Komang Sugiarta, S.Kom., MMSI (Universitas Gunadarma) ;
- Lucia Sri Istiyowati, M.Kom (Institut Perbanas) ;
- Maksy Sendiang, ST, MIT (Politeknik Negeri Manado) ;
- Medi Noviana (Universitas Gunadarma) ;
- Muhammad Nashrullah (Politeknik Negeri Batam) ;
- Nat. I Made Wiryana, S.Si., S.Kom., M.Sc. (Universitas Gunadarma) ;
- Rika Idmayanti, ST, M.Kom (Politeknik Negeri Padang) ;
- Rizky Yuniar Hakkun (Politeknik Elektronik Negeri Surabaya) ;
- Robinson A.Wadu, ST., MT (Politeknik Negeri Kupang) ;
- Roslina. M.IT (Politeknik Negeri Medan) ;
- Sukamto, SKom., MT. (Politeknik Negeri Semarang) ;
- Syamsi Dwi Cahya, M.Kom. (Politeknik Negeri Jakarta) ;
- Syamsul Arifin, S.Kom, M.Cs (Politeknik Negeri Jember) ;
- Usmanudin (Universitas Gunadarma) ;
- Wandy Alifha Saputra (Politeknik Negeri Banjarmasin) ;