

# 2 D4 - TEKKOM B

## PRAKTIKUM 3



Nama : Septian Bagus Jumanoro  
Kelas : 2 – D4 Teknik Komputer B  
NRP : 3221600039  
Dosen : Maretha Ruswiansari S.ST.,M.T  
Mata Kuliah : Praktikum Bahasa Pemrograman  
Hari/Tgl. Praktikum : Kamis, 27 Oktober 202

## Praktikum 3

### Aplikasi Android Dasar (*Methods*)

#### I. Tujuan Pembelajaran

- Mahasiswa dapat menerapkan dan menjelaskan penggunaan *Linear Layout*, *Image View*, *Edit Text*, *Button*, dan *Text View* pada aplikasi Android
- Mahasiswa dapat menerapkan dan menjelaskan penggunaan *Methods*
- Mahasiswa dapat menerapkan *Toast alert* pada suatu *conditional logic*

#### II. Perangkat Praktikum

- PC atau Laptop
- Android Studio
- Android Device

#### III. Prosedur Percobaan

1. Buat *new project* > pilih *project template* > pilih *empty activity* > *next*
2. Lakukan konfigurasi *project* > *finish*
3. Buka app > res > layout > activity\_main.xml
4. Pada percobaan ini, kalian akan membuat aplikasi Android yang terdiri dari *one screen layout* menggunakan *Linear Layout*. Dalam aplikasi akan menampilkan gambar menggunakan *Image View*, menginputkan nilai menggunakan *Edit Text*, melakukan konversi nilai melalui *Button* dan menampilkan hasil menggunakan *Text View*
5. Pada activity\_main.xml, gantikan *root element* dengan *Linear Layout* sebagai orientasi vertical, kemudian *resource manager* tambahkan *icon* yang akan digunakan > klik tanda (+) > pilih *import drawables*

```
<?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">

    <ImageView
        android:layout_width="160dp"
        android:layout_height="145dp"
        android:src="@drawable/height" />

    <EditText
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/edit_text_tinggi"
        android:hint="Tinggi Badan [ cm ]"
        android:layout_margin="30dp"
        android:textAlignment="center"/>
```

```

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/button_hitung"
    android:text="Hitung"
    android:layout_margin="12dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/text_hasil"
    android:layout_margin="12dp"
    android:textSize="16pt"
/>

</LinearLayout>
LinearLayout

```

6. Kalian akan menerapkan tiga Java *Method*, yaitu `findViews()`, `hitungTinggi()`, dan `tampilHasil()`

```

private void findViews(){
    tinggiEdit = findViewById(R.id.edit_text_tinggi);
    hitungButton = findViewById(R.id.button_hitung);
    hasilText = findViewById(R.id.text_hasil);
}

private double hitungTinggi(String height){
    int input = Integer.parseInt(height);
    return input/30.48;
}

private void tampilHasil(double result){
    DecimalFormat koma = new DecimalFormat( pattern: "0.00");
    String hasil = koma.format(result);
    hasilText.setText(hasil + " ft");
}
}

```

7. Penggunaan *conditional logic* dilakukan untuk mengecek apakah input nilai telah dilakukan atau kosong, bila kosong maka akan menampilkan *Toast alert*

```
package com.example.introjava;

import ...

public class MainActivity extends AppCompatActivity {

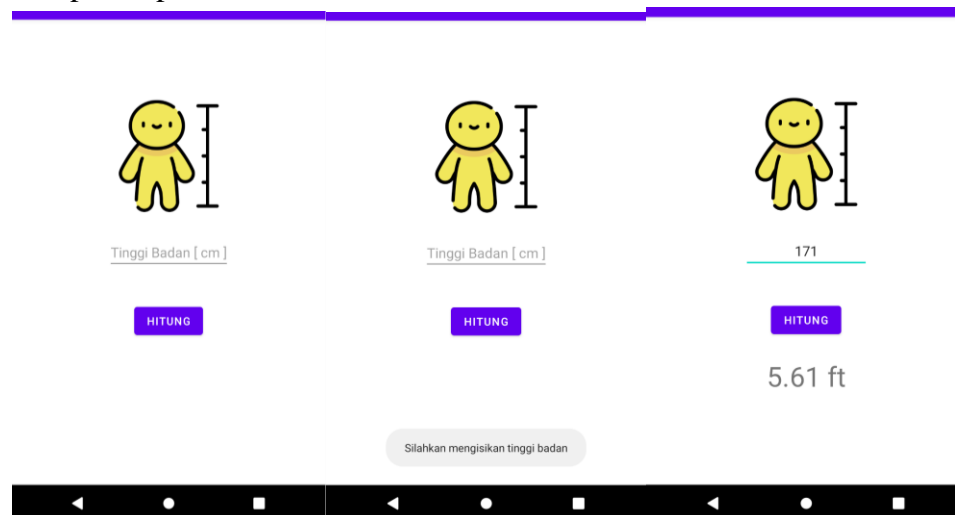
    EditText tinggiEdit;
    Button hitungButton;
    TextView hasilText;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        findViews();
        hitungButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String height = tinggiEdit.getText().toString();
                if(height.isEmpty()){
                    Toast.makeText(context: MainActivity.this, text: "Silahkan mengisikan tinggi badan", Toast.LENGTH_SHORT).show();
                }else {
                    double result = hitungTinggi(height);
                    tampilHasil(result);
                }
            }
        });
    }

    private void findViews(){
```

8. Tampilan aplikasi



#### IV. Hasil Praktikum

- **Source Code**
  - Code XML

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="180dp"
        android:layout_height="300dp"
        android:layout_marginBottom="68dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.234"
        app:srcCompat="@drawable/gambar_tinggibadan" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hitung"
        app:layout_constraintBottom_toTopOf="@+id/textView"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/etx" />

    <EditText
        android:id="@+id/etx"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Tinggi Badan [cm]"
        android:inputType="number"
        android:textAlignment="center"
        app:layout_constraintBottom_toTopOf="@+id/button"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/imageView"
        app:layout_constraintVertical_bias="0.144" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="148dp"
        android:layout_height="wrap_content"
```

```

        android:hint="Hasil"
        android:textAlignment="center"
        android:textSize="20pt"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.776" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

## ➤ Code Activity JAVA

```

package com.example.praktikum3;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import java.text.DecimalFormat;

public class MainActivity extends AppCompatActivity {

    EditText tinggiEdit;
    Button hitungButton;
    TextView hasilText;

    private void findViews(){
        tinggiEdit = findViewById(R.id.etx);
        hitungButton = findViewById(R.id.button);
        hasilText = findViewById(R.id.textView);
    }

    private double hitungTinggi(String height){
        int input = Integer.parseInt(height);
        return input/30.48;
    }

    private void tampilHasil(double result){
        DecimalFormat koma = new DecimalFormat("0.00");
        String hasil = koma.format(result);
        hasilText.setText(hasil + " ft");
    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        findViews();
    }
}

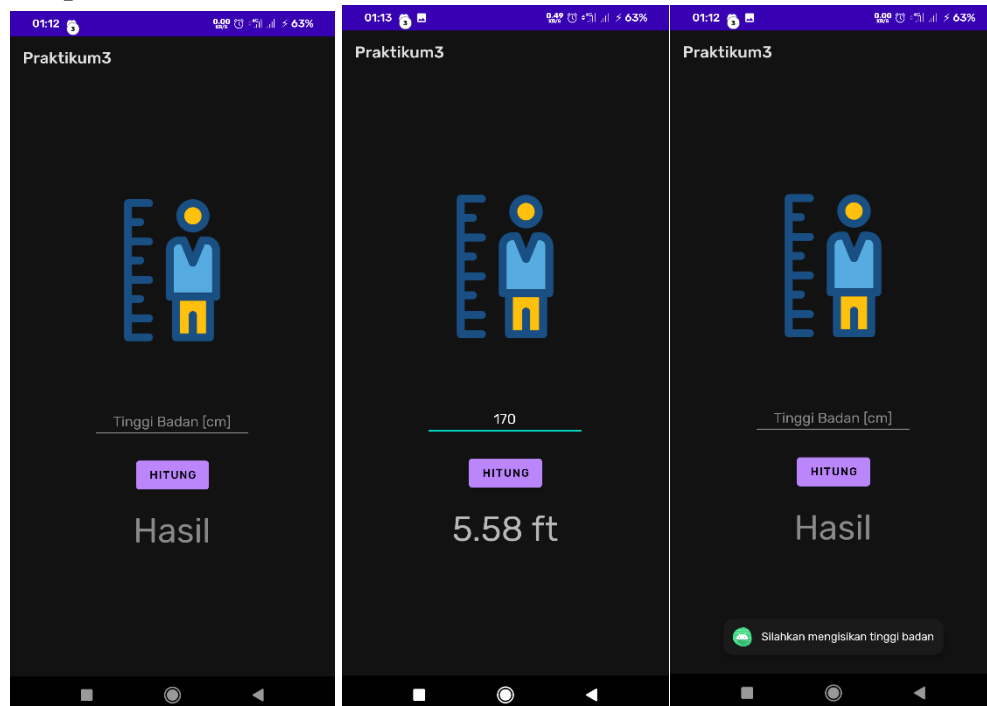
```

```

        hitungButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String height = tinggiEdit.getText().toString();
                if(height.isEmpty()){
                    Toast.makeText(MainActivity.this, "Silahkan mengisi tinggi badan", Toast.LENGTH_SHORT).show();
                }
                else{
                    double result = hitungTinggi(height);
                    tampilHasil(result);
                }
            }
        });
    }
}

```

- **Output**

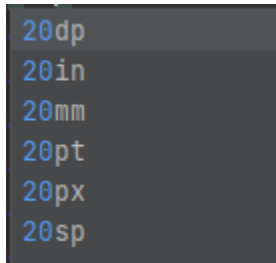


- **Analisa**

Pada praktikum tersebut dapat diketahui bahwa ketika nilai input yang berbentuk cm, akan memberikan nilai output dalam konversi menjadi ft ketika button ditekan. Pada kode program, digunakan sebuah method atau fungsi untuk mengurangi

penulisan yang berulang. Method yang dibuat adalah findViews(), hitungTinggi(), tampilHasil().

Pada file xml terdapat fungsi linearLayout, yang berguna untuk menata komponen yang berada didalamnya disusun secara vertical. ImageView menampilkan gambar yang diambil dari resource drawable. Digunakan untuk memasukkan nilai input. Button digunakan untuk memproses nilai input dan mengeluarkan output. TextView berguna untuk menampilkan hasil output menjadi sebuah teks.



Ukuran terbagi menjadi beberapa jenis satuan seperti:

- dp (Density-independent Pixels)  
Satuan ukuran yang didasarkan pada kepadatan fisik layar. Dimana 1 dp adalah 1 pixel pada layar 160 dpi.
- in (Inches)  
Satuan ukuran yang didasarkan pada ukuran fisik layar dalam inchi
- mm (Milimeters)  
Satuan ukuran berdasarkan ukuran fisik layar dalam milimeter
- pt (Points)  
Satuan ukuran dengan perhitungan  $1/72$  dari satu inci yang didasarkan pada ukuran fisik layar
- px (Pixels)  
Satuan ukuran yang berkorespondensi dengan pixels yang sebenarnya pada layar
- sp (Scale-independent Pixels)  
Hampir sama dengan dp, namun sp memiliki skala dengan preferensi ukuran huruf.

Pada kode program, method yang digunakan yaitu findViews(), hitungTinggi(), tampilHasil(). Setiap method memiliki fungsinya sendiri yaitu:

- findViews()



Digunakan untuk mewadahi deklarasi semua komponen yang ada dalam aplikasi didalam satu fungsi.

- `hitungTinggi()`

Digunakan untuk menghitung nilai cm menjadi ft

- `tampilHasil()`

Digunakan untuk memberikan output hasil perhitungan menjadi bentuk String

`findViews()` dan `tampilHasil()` memiliki bentuk void, yang artinya keluaran dari fungsi tersebut adalah isi dari fungsinya. Sedangkan `hitungTinggi()` memiliki bentuk double, yang artinya akan memberikan nilai balik berbentuk double ketika diberi masukan.

```
private void findViews() {  
    tinggiEdit = findViewById(R.id.etx);  
    hitungButton = findViewById(R.id.button);  
    hasilText = findViewById(R.id.textView);  
}
```

`findViews()` berfungsi untuk mewadahi semua deklarasi komponen menjadi satu fungsi `findViews()` agar ketika di bagian `Oncreate()` tidak memenuhi deklarasi dan menjadi lebih rapi.

```
private double hitungTinggi(String height) {  
    int input = Integer.parseInt(height);  
    return input/30.48;  
}
```

`hitungTinggi()` berfungsi untuk menghitung cm menjadi ft. Nilai cm didapat dari input editText yang diubah menjadi int dengan fungsi `parseInt(height)`. Height adalah nilai string yang didapat dari editText. Nilai tersebut dimasukan kedalam input lalu dikeluarkan dengan `return input/30.48`.

```
private void tampilHasil(double result) {  
    DecimalFormat koma = new DecimalFormat("0.00");  
    String hasil = koma.format(result);  
    hasilText.setText(hasil + " ft");  
}
```

`tampilHasil()` berfungsi untuk mengubah dan menampilkan hasil dari perhitungan tadi menjadi bentuk string lalu dikeluarkan outputnya ke textView. Digunakan

DecimalFormat untuk memberikan nilai keluaran yang memiliki koma dengan 2 nol dibelakang koma.

```
hitungButton.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View view) {  
        String height = tinggiEdit.getText().toString();  
        if(height.isEmpty()){  
            Toast.makeText(MainActivity.this, "Silahkan mengisi tinggi  
badan", Toast.LENGTH_SHORT).show();  
        }  
        else{  
            double result = hitungTinggi(height);  
            tampilHasil(result);  
        }  
    }  
});
```

Ketika button di click, button akan memproses nilai yang diinputkan dan akan mengeluarkan nilai tersebut menjadi output. Ketika nilai input kosong, maka akan keluar sebuah notice untuk mengisi nilai input, dan ketika nilai tidak kosong maka akan diproses nilai tersebut menggunakan fungsi hitungTinggi() dan output dikeluarkan dengan tampilHasil()

## V. Tugas

- **Source Code**
  - Code XML

```
<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout  
xmlns:android="http://schemas.android.com/apk/res/android"  
xmlns:app="http://schemas.android.com/apk/res-auto"  
xmlns:tools="http://schemas.android.com/tools"  
android:layout_width="match_parent"  
android:layout_height="match_parent"  
tools:context=".MainActivity">  
  
    <ImageView  
        android:id="@+id/imageView"  
        android:layout_width="180dp"  
        android:layout_height="276dp"
```

```
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.497"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.0"
app:srcCompat="@drawable/asepstore" />
```

#### <Spinner

```
android:id="@+id/spinner"
android:layout_width="300dp"
android:layout_height="wrap_content"
android:layout_marginStart="55dp"
android:layout_marginLeft="55dp"
android:layout_marginEnd="56dp"
android:layout_marginRight="56dp"
android:spinnerMode="dropdown"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.0"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.356" />
```

#### <EditText

```
android:id="@+id/etxNama"
android:layout_width="309dp"
android:layout_height="wrap_content"
android:layout_marginStart="103dp"
android:layout_marginLeft="103dp"
android:layout_marginTop="16dp"
android:layout_marginEnd="99dp"
android:layout_marginRight="99dp"
android:layout_marginBottom="26dp"
android:ems="10"
android:hint="Masukkan Nama"
android:inputType="textPersonName"
app:layout_constraintBottom_toTopOf="@+id/etxHari"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.52"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/imageView" />
```

#### <EditText

```
android:id="@+id/etxHari"
android:layout_width="309dp"
android:layout_height="wrap_content"
android:layout_marginStart="103dp"
android:layout_marginLeft="103dp"
android:layout_marginEnd="99dp"
android:layout_marginRight="99dp"
android:layout_marginBottom="30dp"
android:ems="10"
android:hint="Total Kehadiran"
android:inputType="number"
app:layout_constraintBottom_toTopOf="@+id/etxLembur"
app:layout_constraintEnd_toEndOf="parent"
```

```

app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/etxNama" />

<EditText
    android:id="@+id/etxLembur"
    android:layout_width="309dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="103dp"
    android:layout_marginLeft="103dp"
    android:layout_marginEnd="99dp"
    android:layout_marginRight="99dp"
    android:ems="10"
    android:hint="Total Jam Lembur"
    android:inputType="number"
    app:layout_constraintBottom_toTopOf="@+id/button"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/etxHari" />

<Button
    android:id="@+id/button"
    android:layout_width="309dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="51dp"
    android:layout_marginLeft="51dp"
    android:layout_marginTop="18dp"
    android:layout_marginEnd="51dp"
    android:layout_marginRight="51dp"
    android:layout_marginBottom="182dp"
    android:text="Button"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/etxLembur" />

<TextView
    android:id="@+id/txtView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:hint="Hasil"
    android:textSize="20sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.139"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.875" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

## ➤ Code Activity JAVA

```

package com.example.tugas;

import androidx.appcompat.app.AppCompatActivity;

```

```

import android.os.Bundle;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

import java.text.SimpleDateFormat;
import java.text.DecimalFormat;
import java.text.DateFormat;
import java.util.Calendar;
import java.util.Date;

public class MainActivity extends AppCompatActivity {

    String[] divisi = {"JOBDESK", "Manager", "Coach", "Team Leader",
"Player"};

    double gajiPokok, tunjangan;

    Spinner jobdesk;
    EditText Nama, Hari, Lembur;
    Button tombol;
    TextView hasil;

    private void findViews(){
        jobdesk = findViewById(R.id.spinner);
        Nama = findViewById(R.id.etxNama);
        Hari = findViewById(R.id.etxHari);
        Lembur = findViewById(R.id.etxLembur);
        hasil = findViewById(R.id.txtView);
        tombol = findViewById(R.id.button);
    }

    private void upah(int divisi, String hariKerja, String jamLembur){
        int nilaiHari = Integer.parseInt(hariKerja);
        int nilaiLembur = Integer.parseInt(jamLembur);

        double persen;

        if(nilaiHari>=20)
            persen = 1;
        else persen = 0.9;

        if(nilaiLembur>=20)
            persen +=0.1;

        switch(divisi){
            case 0:
                gajiPokok = 0;
                tunjangan = 0;
                break;
            case 1:

```

```

        gajiPokok = 5000000 * persen;
        tunjangan= 2000000 * persen;
        break;
    case 2:
        gajiPokok = 3000000 * persen;
        tunjangan= 1500000 * persen;
        break;
    case 3:
        gajiPokok = 2000000 * persen;
        tunjangan= 1000000 * persen;
        break;
    case 4:
        gajiPokok = 1000000 * persen;
        tunjangan= 500000 * persen;
        break;
    }
}

private double totalGaji(double gajiPokok, double tunjangan){

    return gajiPokok+tunjangan;
}

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    findViews();

    ArrayAdapter<String> adapter = new
ArrayAdapter<String>(MainActivity.this,
android.R.layout.simple_spinner_dropdown_item,divisi);
    jobdesk.setAdapter(adapter);

    tombol.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            DateFormat tanggal = new SimpleDateFormat("dd-MM-yyyy");
            String tanggalSekarang = tanggal.format(new Date());

            String nama = Nama.getText().toString();
            String hariKerja = Hari.getText().toString();
            String jamLembur = Lembur.getText().toString();
            int divisi = jobdesk.getSelectedItemPosition();

            if(hariKerja.isEmpty())
                hariKerja = "0";
            if(jamLembur.isEmpty())
                jamLembur = "0";

            upah(divisi, hariKerja, jamLembur);

            if(divisi==0){
                Toast.makeText(MainActivity.this, "Silahkan Pilih Jobdesk
Anda", Toast.LENGTH_SHORT).show();
            }
        }
    });
}

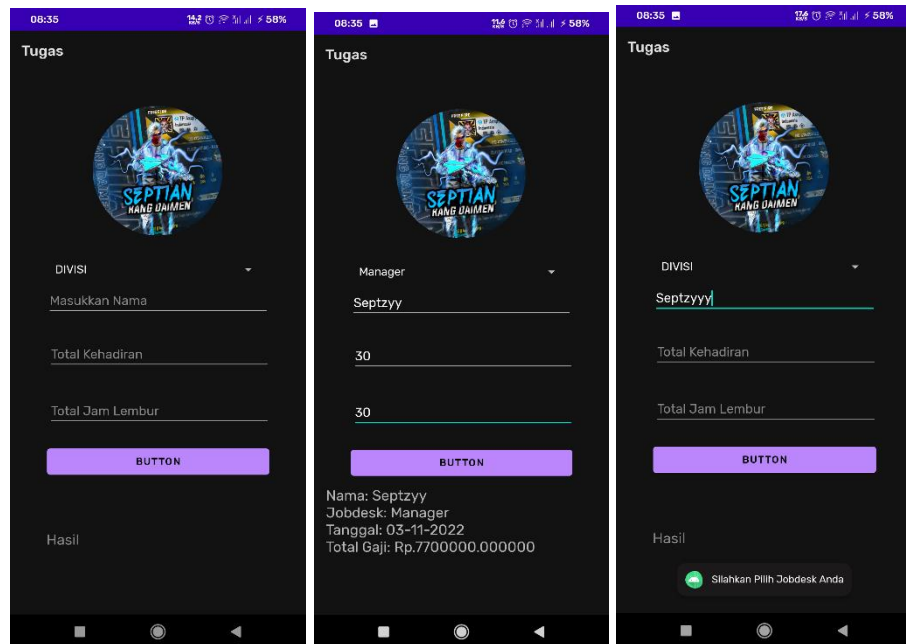
```

```

        else{
            hasil.setText(String.format("Nama: %s\nJobdesk:
%s\nTanggal: %s\nTotal Gaji: Rp.%.2f", nama,jobdesk.getSelectedItemAt(),
tanggalSekarang, totalGaji(gajiPokok, tunjangan)));
        }
    }
}
});
}
}
}

```

- **Output**



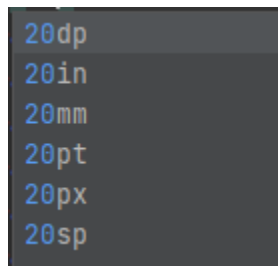
- **Analisa**

Pada praktikum tersebut dapat diketahui bahwa ketika divisi dipilih dan input yang ada diisi, maka akan memproses hasil dan mengeluarkan ouputnya pada textview ketika button ditekan.

Pada file xml terdapat fungsi linearLayout yang berguna untuk menata komponen yang berada didalamnya disusun secara vertical. imageView menampilkan gambar yang diambil dari resource drawable. Spinner sebagai dropdown list. editText digunakan untuk memasukkan nilai input. Button digunakan untuk memproses nilai input dan mengeluarkan output. textView berguna untuk menampilkan hasil output menjadi sebuah teks.

Pada komponen tersebut memiliki kodenya tersendiri yang disimpan dalam bentuk XML. Kode tersebut memiliki fungsinya masing-masing yaitu:

- `android:id="@+id/imageView13"`  
dimiliki pada hampir setiap komponen yang ada, digunakan untuk mendefinisikan id dari komponennya untuk dipanggil pada kode program.
- `app:srcCompat="@drawable/height"`  
ada pada `imageView`, digunakan untuk memilih gambar yang terdapat pada `drawable`
- `android:gravity="center"`  
ada pada komponen yang memiliki teks, digunakan untuk menengahkan teks
- `android:inputType="number"`  
ada pada `editText`, digunakan untuk memberikan sifat pada input `editText`
- `android:hint="hari masuk dalam 1 bulan"`  
ada pada `editText`, digunakan untuk memberikan petunjuk atau tulisan abu-abu pada teks.
- `android:text="Button"`  
ada pada komponen yang memiliki teks, digunakan untuk memberikan tulisan pada komponen



Ukuran terbagi menjadi beberapa jenis satuan seperti:

- **dp (Density-independent Pixels)**  
Satuan ukuran yang didasarkan pada keadaan fisik layar. Dimana 1 dp adalah 1 pixel pada layar 160 dpi.
- **in (Inches)**  
Satuan ukuran yang didasarkan pada ukuran fisik layar dalam inchi
- **mm (Milimeters)**  
Satuan ukuran berdasarakan ukuran fisik layar dalam milimeter



- pt (Points)  
Satuan ukuran dengan perhitungan  $1/72$  dari satu inci yang didasarkan pada ukuran fisik layar
- px (Pixels)  
Satuan ukuran yang berkorespondensi dengan pixels yang sebenarnya pada layar
- sp (Scalae-independent Pixels)  
Hampir sama dengan dp, namun sp memiliki skala dengan preferensi ukuran huruf

Pada kode program, method yang digunakan yaitu findViews(), hitungTinggi(), tampilHasil(). Setiap method memiliki fungsinya sendiri yaitu:

- findViews()  
Digunakan untuk mewadahi deklarasi semua komponen yang ada dalam aplikasi didalam satu fungsi.
  - hitungTinggi()  
Digunakan untuk menghitung nilai cm menjadi ft
  - tampilHasil()  
Digunakan untuk memberikan output hasil perhitungan menjadi bentuk String
- findViews() dan tampilHasil() memiliki bentuk void, yang artinya keluaran dari fungsi tersebut adalah isi dari fungsinya. Sedangkan hitungTinggi() memiliki bentuk double, yang artinya akan memberikan nilai balik berbentuk double ketika diberi masukan.

```
private void findViews() {
    jobdesk = findViewById(R.id.spinner);
    Nama     = findViewById(R.id.etxNama);
    Hari     = findViewById(R.id.etxHari);
    Lembur   = findViewById(R.id.etxLembur);
    hasil    = findViewById(R.id.txtView);
    tombol   = findViewById(R.id.button);
}
```

findViews() memiliki fungsi untuk mewadahi semua deklarasi komponen menjadi satu fungsi findViews() agar ketika di bagian Oncreate() tidak memenuhi deklarasi dan menjadi lebih rapi.

```

private void upah(int divisi, String hariKerja, String jamLembur){
    int nilaiHari = Integer.parseInt(hariKerja);
    int nilaiLembur = Integer.parseInt(jamLembur);

    double persen;

    if(nilaiHari>=20)
        persen = 1;
    else persen = 0.9;

    if(nilaiLembur>=20)
        persen +=0.1;

    switch(divisi){
        case 0:
            gajiPokok = 0;
            tunjangan = 0;
            break;
        case 1:
            gajiPokok = 5000000 * persen;
            tunjangan= 2000000 * persen;
            break;
        case 2:
            gajiPokok = 3000000 * persen;
            tunjangan= 1500000 * persen;
            break;
        case 3:
            gajiPokok = 2000000 * persen;
            tunjangan= 1000000 * persen;
            break;
        case 4:
            gajiPokok = 1000000 * persen;
            tunjangan= 500000 * persen;
            break;
    }
}

```

hitungGaji() memiliki fungsi untuk menghitung gaji dengan ketentuan divisi, hari kerja, dan jam lembur. Nilai hariKerja dan jamLembur didapat dari input editText yang diubah menjadi int dengan fungsi parseInt(height). Divisi didapat dari spinner yang listnya dipilih salah satu dan diambil posisinya. gajiPokok dan tunjangan didapat dari hasil proses hitungGaji yang nilainya telah ditentukan.

```

private double totalGaji(double gajiPokok, double tunjangan){

    return gajiPokok+tunjangan;
}

```

totalGaji() memiliki fungsi untuk menghitung total gajipokok+tunjangan

```
if (divisi==0) {  
    Toast.makeText(MainActivity.this, "Silahkan Pilih Jobdesk Anda",  
    Toast.LENGTH_SHORT).show();  
}  
else{  
    hasil.setText(String.format("Nama: %s\nJobdesk: %s\nTanggal:  
%s\nTotal Gaji: Rp.%.2f", nama, jobdesk.getSelectedItem(), tanggalSekarang,  
    totalGaji(gajiPokok, tunjangan)));  
}  
}
```

Ketika button di click, button akan memproses nilai yang diinputkan dan akan menampilkan nilai tersebut. Ketika Jobdesk tidak dipilih, maka akan keluar sebuah PopUp untuk memilih jobdesk terlebih dahulu, dan ketika jobdesk dipilih maka akan diproses nilai tersebut menggunakan fungsi hitungGaji dan output nilai dikeluarkan pada textViewnya.