

## LAMPIRAN

### 1. Konsultasi.java

```
package com.example.saksermu.dentalexsys;

import android.app.Activity;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.TextView;
import android.view.View.OnClickListener;

public class Konsultasi extends Activity implements
OnClickListener {
    CheckBox
G01, G02, G03, G04, G05, G06, G07, G08, G09, G10, G11, G12, G13, G14, G15, G16
, G17, G18, G19, G20, G21, G22,
        G23, G24, G25, G26, G27, G28, G29, G30, G31, G32, G33, G34;
    TextView hsl, solusi;
    Button AnalisaGejala;
    String Penyakit;
    String Solusi;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_konsultasi);
        AnalisaGejala =
        (Button) findViewById(R.id.buttonAnalisa);

        G01=(CheckBox) findViewById(R.id.Gejala1);
        G02=(CheckBox) findViewById(R.id.Gejala2);
        G03=(CheckBox) findViewById(R.id.Gejala3);
        G04=(CheckBox) findViewById(R.id.Gejala4);
        G05=(CheckBox) findViewById(R.id.Gejala5);
        G06=(CheckBox) findViewById(R.id.Gejala6);
        G07=(CheckBox) findViewById(R.id.Gejala7);
        G08=(CheckBox) findViewById(R.id.Gejala8);
        G09=(CheckBox) findViewById(R.id.Gejala9);
        G10=(CheckBox) findViewById(R.id.Gejala10);
        G11=(CheckBox) findViewById(R.id.Gejala11);
        G12=(CheckBox) findViewById(R.id.Gejala12);
        G13=(CheckBox) findViewById(R.id.Gejala13);
        G14=(CheckBox) findViewById(R.id.Gejala14);
        G15=(CheckBox) findViewById(R.id.Gejala15);
        G16=(CheckBox) findViewById(R.id.Gejala16);
        G17=(CheckBox) findViewById(R.id.Gejala17);
        G18=(CheckBox) findViewById(R.id.Gejala18);
        G19=(CheckBox) findViewById(R.id.Gejala19);
        G20=(CheckBox) findViewById(R.id.Gejala20);
        G21=(CheckBox) findViewById(R.id.Gejala21);
        G22=(CheckBox) findViewById(R.id.Gejala22);
```

```

G23=(CheckBox) findViewById(R.id.Gejala23);
G24=(CheckBox) findViewById(R.id.Gejala24);
G25=(CheckBox) findViewById(R.id.Gejala25);
G26=(CheckBox) findViewById(R.id.Gejala26);
G27=(CheckBox) findViewById(R.id.Gejala27);
G28=(CheckBox) findViewById(R.id.Gejala28);
G29=(CheckBox) findViewById(R.id.Gejala29);
G30=(CheckBox) findViewById(R.id.Gejala30);
G31=(CheckBox) findViewById(R.id.Gejala31);
G32=(CheckBox) findViewById(R.id.Gejala32);
G33=(CheckBox) findViewById(R.id.Gejala33);
G34=(CheckBox) findViewById(R.id.Gejala34);

hsl=(TextView) findViewById(R.id.hasil);
solusi=(TextView) findViewById(R.id.solusi);

AnalisaGejala.setOnClickListener(this);
}

public void onClick(View v){
    //-----Penyakit Karies Gigi 1,2,3,4-----//
    if
(G01.isChecked() &&G02.isChecked() &&G03.isChecked() &&G04.isCheck
ed()){
        hsl.setText("Anda Terkena Penyakit : Karies Gigi");
        solusi.setText("Solusi : Restorasi, Pembersihan dan
Sterilisasi saluran akar");
    }else if (G02.isChecked() && G03.isChecked()){
        hsl.setText("Anda Terkena Penyakit : Karies Gigi");
        solusi.setText("Solusi : Restorasi, Pembersihan dan
Sterilisasi saluran akar");
    }else if (G01.isChecked() && G03.isChecked() &&
G02.isChecked()){
        hsl.setText("Anda Terkena Penyakit : Karies Gigi");
        solusi.setText("Solusi : Restorasi, Pembersihan dan
Sterilisasi saluran akar");
    }else if (G03.isChecked() && G04.isChecked() &&
G01.isChecked()){
        hsl.setText("Anda Terkena Penyakit : Karies Gigi");
        solusi.setText("Solusi : Restorasi, Pembersihan dan
Sterilisasi saluran akar");
    }else if (G01.isChecked() && G02.isChecked() &&
G04.isChecked()){
        hsl.setText("Anda Terkena Penyakit : Karies Gigi");
        solusi.setText("Solusi : Restorasi, Pembersihan dan
Sterilisasi saluran akar");
    }else if (G01.isChecked() && G04.isChecked()){
        hsl.setText("Anda Terkena Penyakit : Karies Gigi");
        solusi.setText("Solusi : Restorasi, Pembersihan dan
Sterilisasi saluran akar");
    }else if (G04.isChecked()){
        hsl.setText("Anda Terkena Penyakit : Karies Gigi");
        solusi.setText("Solusi : Restorasi, Pembersihan dan
Sterilisasi saluran akar");
    }
}

```

```

}else if (G03.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Karies Gigi");
    solusi.setText("Solusi : Restorasi, Pembersihan dan
Sterilisasi saluran akar");
}else if (G02.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Karies Gigi");
    solusi.setText("Solusi : Restorasi, Pembersihan dan
Sterilisasi saluran akar");
}else if (G01.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Karies Gigi");
    solusi.setText("Solusi : Restorasi, Pembersihan dan
Sterilisasi saluran akar");
}else if (G01.isChecked() && G02.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Karies Gigi");
    solusi.setText("Solusi : Restorasi, Pembersihan dan
Sterilisasi saluran akar");
}

//-----Pulpitis Reversible 1,2,5-----//
if (G01.isChecked() && G02.isChecked() && G05.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis Reversible");
    solusi.setText("Solusi : Dipulpotomi (pemotongan pulpa),
restorasi, pembersihan dan sterilisasi saluran akar dan
pemberian obat anti nyeri");
}else if (G02.isChecked() && G05.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis Reversible");
    solusi.setText("Solusi : Dipulpotomi (pemotongan pulpa),
restorasi, pembersihan dan sterilisasi saluran akar dan
pemberian obat anti nyeri");
}else if (G01.isChecked() && G05.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis Reversible");
    solusi.setText("Solusi : Dipulpotomi (pemotongan pulpa),
restorasi, pembersihan dan sterilisasi saluran akar dan
pemberian obat anti nyeri");
}else if (G05.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis Reversible");
    solusi.setText("Solusi : Dipulpotomi (pemotongan pulpa),
restorasi, pembersihan dan sterilisasi saluran akar dan
pemberian obat anti nyeri");
}

//-----Pulpitis Ireversible 6,7,8,9,10-----//
if (G06.isChecked() && G07.isChecked() && G08.isChecked() &&
G09.isChecked() && G10.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis
Ireversible");
    solusi.setText("Solusi : Pemberian obat pereda nyeri,
devitalisasi (syaraf dimatikan) dan dilakukan perawatan saluran
akar");
}else if (G06.isChecked() && G07.isChecked() && G10.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis
Ireversible");
    solusi.setText("Solusi : Pemberian obat pereda nyeri,
devitalisasi (syaraf dimatikan) dan dilakukan perawatan saluran
akar");
}

```

```

}else if (G07.isChecked() && G09.isChecked() && G10.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis
Ireversible");
    solusi.setText("Solusi : Pemberian obat pereda nyeri,
devitalisasi (syaraf dimatikan) dan dilakukan perawatan saluran
akar");
}else if (G07.isChecked() && G10.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis
Ireversible");
    solusi.setText("Solusi : Pemberian obat pereda nyeri,
devitalisasi (syaraf dimatikan) dan dilakukan perawatan saluran
akar");
}else if (G06.isChecked() && G09.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis
Ireversible");
    solusi.setText("Solusi : Pemberian obat pereda nyeri,
devitalisasi (syaraf dimatikan) dan dilakukan perawatan saluran
akar");
}else if (G06.isChecked() && G08.isChecked() && G10.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis
Ireversible");
    solusi.setText("Solusi : Pemberian obat pereda nyeri,
devitalisasi (syaraf dimatikan) dan dilakukan perawatan saluran
akar");
}else if (G07.isChecked() && G08.isChecked() && G09.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis
Ireversible");
    solusi.setText("Solusi : Pemberian obat pereda nyeri,
devitalisasi (syaraf dimatikan) dan dilakukan perawatan saluran
akar");
}else if (G08.isChecked() && G10.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis
Ireversible");
    solusi.setText("Solusi : Pemberian obat pereda nyeri,
devitalisasi (syaraf dimatikan) dan dilakukan perawatan saluran
akar");
}else if (G07.isChecked() && G08.isChecked() && G10.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis
Ireversible");
    solusi.setText("Solusi : Pemberian obat pereda nyeri,
devitalisasi (syaraf dimatikan) dan dilakukan perawatan saluran
akar");
}else if (G07.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis
Ireversible");
    solusi.setText("Solusi : Pemberian obat pereda nyeri,
devitalisasi (syaraf dimatikan) dan dilakukan perawatan saluran
akar");
}else if (G08.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis
Ireversible");
    solusi.setText("Solusi : Pemberian obat pereda nyeri,
devitalisasi (syaraf dimatikan) dan dilakukan perawatan saluran
akar");
}else if (G09.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Pulpitis
Ireversible");

```

```

//-----Abses Periapikal 7,12,13,14,15-----//
if (G07.isChecked() && G12.isChecked() && G13.isChecked() &&
G14.isChecked() && G15.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Abses Periapikal");
    solusi.setText("Solusi : Dilakukan pembedahan untuk
mengeluarkan nanah atau pengobatan saluran akar");
} else if (G12.isChecked() && G13.isChecked() && G15.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Abses Periapikal");
    solusi.setText("Solusi : Dilakukan pembedahan untuk
mengeluarkan nanah atau pengobatan saluran akar");
} else if (G07.isChecked() && G12.isChecked() && G14.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Abses Periapikal");
    solusi.setText("Solusi : Dilakukan pembedahan untuk
mengeluarkan nanah atau pengobatan saluran akar");
} else if (G12.isChecked() && G14.isChecked() && G15.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Abses Periapikal");
    solusi.setText("Solusi : Dilakukan pembedahan untuk
mengeluarkan nanah atau pengobatan saluran akar");
} else if (G12.isChecked() && G15.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Abses Periapikal");
    solusi.setText("Solusi : Dilakukan pembedahan untuk
mengeluarkan nanah atau pengobatan saluran akar");
} else if (G07.isChecked() && G12.isChecked() && G14.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Abses Periapikal");
    solusi.setText("Solusi : Dilakukan pembedahan untuk
mengeluarkan nanah atau pengobatan saluran akar");
} else if (G12.isChecked() && G13.isChecked() && G15.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Abses Periapikal");
    solusi.setText("Solusi : Dilakukan pembedahan untuk
mengeluarkan nanah atau pengobatan saluran akar");
}
}
//-----Gingivitis 11,12,16,17,18-----//
if (G11.isChecked() && G12.isChecked() && G16.isChecked() &&
G17.isChecked() && G18.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Gingivitis");
    solusi.setText("Solusi : Melakukan Scalling dan
meningkatkan DHE");
} else if (G12.isChecked() && G16.isChecked() && G17.isChecked() &&
G18.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Gingivitis");
    solusi.setText("Solusi : Melakukan Scalling dan
meningkatkan DHE");
} else if (G12.isChecked() && G16.isChecked() && G17.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Gingivitis");
    solusi.setText("Solusi : Melakukan Scalling dan
meningkatkan DHE");
} else if (G17.isChecked() && G18.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Gingivitis");
    solusi.setText("Solusi : Melakukan Scalling dan
meningkatkan DHE");
} else if (G17.isChecked() && G18.isChecked() && G16.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Gingivitis");
    solusi.setText("Solusi : Melakukan Scalling dan
meningkatkan DHE");
} else if (G12.isChecked() && G17.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Gingivitis");
    solusi.setText("Solusi : Melakukan Scalling dan

```

```

}else if (G12.isChecked() && G16.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Gingivitis");
    solusi.setText("Solusi : Melakukan Scalling dan
meningkatkan DHE");
}else if (G16.isChecked() && G18.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Gingivitis");
    solusi.setText("Solusi : Melakukan Scalling dan
meningkatkan DHE");
}else if (G12.isChecked() && G18.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Gingivitis");
    solusi.setText("Solusi : Melakukan Scalling dan
meningkatkan DHE");
}else if (G12.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Gingivitis");
    solusi.setText("Solusi : Melakukan Scalling dan
meningkatkan DHE");
}else if (G16.isChecked() && G17.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Gingivitis");
    solusi.setText("Solusi : Melakukan Scalling dan
meningkatkan DHE");
}else if (G16.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Gingivitis");
    solusi.setText("Solusi : Melakukan Scalling dan
meningkatkan DHE");
}else if (G17.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Gingivitis");
    solusi.setText("Solusi : Melakukan Scalling dan
meningkatkan DHE");
}else if (G18.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Gingivitis");
    solusi.setText("Solusi : Melakukan Scalling dan
meningkatkan DHE");
}
//-----Periodontitis 11,12,16,17,18,19,20-----//
if (G18.isChecked() && G19.isChecked() && G20.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Periodontitis");
    solusi.setText("Solusi : Disesuaikan dengan tingkat
keparahannya. Pemberian antibiotik jika sudah parah");
}else if (G18.isChecked() && G19.isChecked() && G20.isChecked() &&
G11.isChecked() && G12.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Periodontitis");
    solusi.setText("Solusi : Disesuaikan dengan tingkat
keparahannya. Pemberian antibiotik jika sudah parah");
}else if (G11.isChecked() && G19.isChecked() && G20.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Periodontitis");
    solusi.setText("Solusi : Disesuaikan dengan tingkat
keparahannya. Pemberian antibiotik jika sudah parah");
}else if (G16.isChecked() && G17.isChecked() && G20.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Periodontitis");
    solusi.setText("Solusi : Disesuaikan dengan tingkat
keparahannya. Pemberian antibiotik jika sudah parah");
}else if (G18.isChecked() && G19.isChecked() && G20.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Periodontitis");
    solusi.setText("Solusi : Disesuaikan dengan tingkat
keparahannya. Pemberian antibiotik jika sudah parah");
}

```

```

}else if (G11.isChecked() && G17.isChecked() && G19.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Periodontitis");
    solusi.setText("Solusi : Disesuaikan dengan tingkat
keparahannya. Pemberian antibiotik jika sudah parah");
}else if (G17.isChecked() && G19.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Periodontitis");
    solusi.setText("Solusi : Disesuaikan dengan tingkat
keparahannya. Pemberian antibiotik jika sudah parah");
}else if (G17.isChecked() && G19.isChecked() && G20.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Periodontitis");
    solusi.setText("Solusi : Disesuaikan dengan tingkat
keparahannya. Pemberian antibiotik jika sudah parah");
}else if (G19.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Periodontitis");
    solusi.setText("Solusi : Disesuaikan dengan tingkat
keparahannya. Pemberian antibiotik jika sudah parah");
}

//-----Herpes Labialis 14,15,17,21,34-----//
if (G14.isChecked() && G15.isChecked() && G17.isChecked() &&
G21.isChecked() && G34.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Herpes Labialis");
    solusi.setText("Solusi : Menggunakan obat kumur antiseptik,
perbanyak konsumsi Vitamin B kompleks dan Vitamin C, atau
dengan salep Acyclovir");
}else if (G21.isChecked() && G34.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Herpes Labialis");
    solusi.setText("Solusi : Menggunakan obat kumur antiseptik,
perbanyak konsumsi Vitamin B kompleks dan Vitamin C, atau
dengan salep Acyclovir");
}else if (G14.isChecked() && G21.isChecked() && G34.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Herpes Labialis");
    solusi.setText("Solusi : Menggunakan obat kumur antiseptik,
perbanyak konsumsi Vitamin B kompleks dan Vitamin C, atau
dengan salep Acyclovir");
}else if (G14.isChecked() && G15.isChecked() && G34.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Herpes Labialis");
    solusi.setText("Solusi : Menggunakan obat kumur antiseptik,
perbanyak konsumsi Vitamin B kompleks dan Vitamin C, atau
dengan salep Acyclovir");
}else if (G15.isChecked() && G17.isChecked() && G21.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Herpes Labialis");
    solusi.setText("Solusi : Menggunakan obat kumur antiseptik,
perbanyak konsumsi Vitamin B kompleks dan Vitamin C, atau
dengan salep Acyclovir");
}else if (G21.isChecked() && G14.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Herpes Labialis");
    solusi.setText("Solusi : Menggunakan obat kumur antiseptik,
perbanyak konsumsi Vitamin B kompleks dan Vitamin C, atau
dengan salep Acyclovir");
}else if (G21.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Herpes Labialis");
    solusi.setText("Solusi : Menggunakan obat kumur antiseptik,
perbanyak konsumsi Vitamin B kompleks dan Vitamin C, atau
dengan salep Acyclovir");
}else if (G34.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Herpes Labialis");
}

```

```

//-----Stomatitis/Sariawan 11,22,23,24,25-----//
if (G11.isChecked() && G22.isChecked() && G23.isChecked() &&
G24.isChecked() &&G25.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Stomatitis");
    solusi.setText("Solusi : Peningkatan asupan nutrisi,
vitamin dan air, serta pemberian minosep");
}else if (G11.isChecked() && G22.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Stomatitis");
    solusi.setText("Solusi : Peningkatan asupan nutrisi,
vitamin dan air, serta pemberian minosep");
}else if (G22.isChecked() && G23.isChecked() && G24.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Stomatitis");
    solusi.setText("Solusi : Peningkatan asupan nutrisi,
vitamin dan air, serta pemberian minosep");
}else if (G11.isChecked() && G23.isChecked() && G24.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Stomatitis");
    solusi.setText("Solusi : Peningkatan asupan nutrisi,
vitamin dan air, serta pemberian minosep");
}else if (G23.isChecked() && G25.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Stomatitis");
    solusi.setText("Solusi : Peningkatan asupan nutrisi,
vitamin dan air, serta pemberian minosep");
}else if (G11.isChecked() && G22.isChecked() && G24.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Stomatitis");
    solusi.setText("Solusi : Peningkatan asupan nutrisi,
vitamin dan air, serta pemberian minosep");
}else if (G22.isChecked() && G25.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Stomatitis");
    solusi.setText("Solusi : Peningkatan asupan nutrisi,
vitamin dan air, serta pemberian minosep");
}else if (G22.isChecked() && G23.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Stomatitis");
    solusi.setText("Solusi : Peningkatan asupan nutrisi,
vitamin dan air, serta pemberian minosep");
}else if (G22.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Stomatitis");
    solusi.setText("Solusi : Peningkatan asupan nutrisi,
vitamin dan air, serta pemberian minosep");
}else if (G11.isChecked() && G23.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Stomatitis");
    solusi.setText("Solusi : Peningkatan asupan nutrisi,
vitamin dan air, serta pemberian minosep");
}else if (G11.isChecked() && G25.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Stomatitis");
    solusi.setText("Solusi : Peningkatan asupan nutrisi,
vitamin dan air, serta pemberian minosep");
}
}
//-----Glositis 26,27,28,29,30,31-----//
if (G26.isChecked() && G27.isChecked() && G28.isChecked() &&
G29.isChecked() && G30.isChecked() && G31.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Glositis");
    solusi.setText("Solusi : Pengobatan tergantung pada
penyebabnya. Gunakan antibiotik untuk pengobatan infeksi
bakteri");
}else if (G27.isChecked() && G29.isChecked() && G31.isChecked()) {
    hsl.setText("Anda Terkena Penyakit : Glositis");
    solusi.setText("Solusi : Pengobatan tergantung pada

```



```

}else if (G26.isChecked() && G29.isChecked() && G30.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Glositis");
    solusi.setText("Solusi : Pengobatan tergantung pada
penyebabnya. Gunakan antibiotik untuk pengobatan infeksi
bakteri");
}else if (G28.isChecked() && G30.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Glositis");
    solusi.setText("Solusi : Pengobatan tergantung pada
penyebabnya. Gunakan antibiotik untuk pengobatan infeksi
bakteri");
}else if (G27.isChecked() && G28.isChecked() && G30.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Glositis");
    solusi.setText("Solusi : Pengobatan tergantung pada
penyebabnya. Gunakan antibiotik untuk pengobatan infeksi
bakteri");
}else if (G30.isChecked() && G31.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Glositis");
    solusi.setText("Solusi : Pengobatan tergantung pada
penyebabnya. Gunakan antibiotik untuk pengobatan infeksi
bakteri");
}else if (G26.isChecked() && G30.isChecked() && G31.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Glositis");
    solusi.setText("Solusi : Pengobatan tergantung pada
penyebabnya. Gunakan antibiotik untuk pengobatan infeksi
bakteri");
}else if (G26.isChecked() && G31.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Glositis");
    solusi.setText("Solusi : Pengobatan tergantung pada
penyebabnya. Gunakan antibiotik untuk pengobatan infeksi
bakteri");
}else if (G26.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Glositis");
    solusi.setText("Solusi : Pengobatan tergantung pada
penyebabnya. Gunakan antibiotik untuk pengobatan infeksi
bakteri");
}else if (G26.isChecked() && G28.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Glositis");
    solusi.setText("Solusi : Pengobatan tergantung pada
penyebabnya. Gunakan antibiotik untuk pengobatan infeksi
bakteri");
}else if (G27.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Glositis");
    solusi.setText("Solusi : Pengobatan tergantung pada
penyebabnya. Gunakan antibiotik untuk pengobatan infeksi
bakteri");
}else if (G28.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Glositis");
    solusi.setText("Solusi : Pengobatan tergantung pada
penyebabnya. Gunakan antibiotik untuk pengobatan infeksi
bakteri");
}else if (G29.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Glositis");
    solusi.setText("Solusi : Pengobatan tergantung pada
penyebabnya. Gunakan antibiotik untuk pengobatan infeksi
bakteri");
penyebabnya. Gunakan antibiotik untuk pengobatan infeksi
bakteri");

```

```

}else if (G31.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Glositis");
    solusi.setText("Solusi : Pengobatan tergantung pada
//-----Fisure Tongue 27,28,11-----//
if (G11.isChecked() && G27.isChecked() && G28.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Fissured Tongue");
    solusi.setText("Solusi : Menyikat gigi dan menjaga
kebersihan mulut yang baik untuk mengurangi inflamasi");
}else if (G27.isChecked() && G28.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Fissured Tongue");
    solusi.setText("Solusi : Menyikat gigi dan menjaga
kebersihan mulut yang baik untuk mengurangi inflamasi");
}else if (G11.isChecked() && G28.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Fissured Tongue");
    solusi.setText("Solusi : Menyikat gigi dan menjaga
kebersihan mulut yang baik untuk mengurangi inflamasi");
}

//-----Black Hairy Tongue 11,27,28,32-----//
if (G27.isChecked() && G28.isChecked() && G32.isChecked() &&
G11.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Black Hairy Tongue");
    solusi.setText("Solusi : Penyikatan dorsum lidah
menggunakan tongue scaper (pembersihan lidah) 2x sehari");
}else if (G11.isChecked() && G32.isChecked() && G27.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Black Hairy Tongue");
    solusi.setText("Solusi : Penyikatan dorsum lidah
menggunakan tongue scaper (pembersihan lidah) 2x sehari");
}else if (G27.isChecked() && G32.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Black Hairy Tongue");
    solusi.setText("Solusi : Penyikatan dorsum lidah
menggunakan tongue scaper (pembersihan lidah) 2x sehari");
}else if (G11.isChecked() && G32.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Black Hairy Tongue");
    solusi.setText("Solusi : Penyikatan dorsum lidah
menggunakan tongue scaper (pembersihan lidah) 2x sehari");
}else if (G32.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Black Hairy Tongue");
    solusi.setText("Solusi : Penyikatan dorsum lidah
menggunakan tongue scaper (pembersihan lidah) 2x sehari");
}

//-----Georaphic Tongue 23,33-----//
if (G23.isChecked() && G33.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Geographic Tongue");
    solusi.setText("Solusi : Menggunakan obat penghilang nyeri
(Analgesik), larutan kumur yang mengandung antihistamin");
}else if (G33.isChecked()){
    hsl.setText("Anda Terkena Penyakit : Geographic Tongue");
    solusi.setText("Solusi : Menggunakan obat penghilang nyeri
(Analgesik), larutan kumur yang mengandung antihistamin");

    G01.setChecked(false);
    G02.setChecked(false);
    G03.setChecked(false);
    G04.setChecked(false);
    G05.setChecked(false);

```

```

G06.setChecked(false);
G07.setChecked(false);
G08.setChecked(false);
G09.setChecked(false);
G10.setChecked(false);
G11.setChecked(false);
G12.setChecked(false);
G13.setChecked(false);
G14.setChecked(false);
G15.setChecked(false);
G16.setChecked(false);
G17.setChecked(false);
G18.setChecked(false);
G19.setChecked(false);
G20.setChecked(false);
G21.setChecked(false);
G22.setChecked(false);
G23.setChecked(false);
G24.setChecked(false);
G25.setChecked(false);
G26.setChecked(false);
G27.setChecked(false);
G28.setChecked(false);
G29.setChecked(false);
G30.setChecked(false);
G31.setChecked(false);
G32.setChecked(false);
G33.setChecked(false);
G34.setChecked(false);
}
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if
    it is present.
    getMenuInflater().inflate(R.menu.menu_konsultasi, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so
    long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();
    //noinspection SimplifiableIfStatement
    if (id == R.id.action_settings) {
        return true;
    }

    return super.onOptionsItemSelected(item);
}
}

```

## 2. SQLiteHelper.java

```
package com.example.saksermu.dentalexsys;

import java.util.ArrayList;
import java.util.HashMap;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

/**
 * Created by sak sermu on 23/08/2016.
 */
public class SQLiteHelper extends SQLiteOpenHelper {

    private static final String nama_database =
"database_gilut.db";
    private static final int versi_database = 1;
    private static final String query_buat_tabel_gejala =
"CREATE TABLE IF NOT EXISTS tabel_gejala(kd_gejala INTEGER
PRIMARY KEY AUTOINCREMENT, nama_gejala TEXT)";
    private static final String query_buat_tabel_penyakit =
"CREATE TABLE IF NOT EXISTS tabel_penyakit(kd_penyakit INTEGER
PRIMARY KEY AUTOINCREMENT, nama_penyakit TEXT, definisi TEXT)";
    private static final String query_buat_tabel_solusi
="CREATE TABLE IF NOT EXISTS tabel_solusi(kd_solusi INTEGER
PRIMARY KEY AUTOINCREMENT, solusi TEXT)";
    private static final String query_buat_tabel_rule="CREATE
TABLE IF NOT EXISTS tabel_rule(kd_rule INTEGER PRIMARY KEY
AUTOINCREMENT, kd_penyakit TEXT, nama_penyakit TEXT, kd_gejala
TEXT, kd_solusi TEXT)";
    private static final String query_hapus_tabel_gejala =
"DROP TABLE IF EXISTS query_buat_tabel_gejala ";
    private static final String query_hapus_tabel_penyakit =
"DROP TABLE IF EXISTS query_buat_tabel_penyakit ";
    private static final String query_hapus_tabel_solusi =
"DROP TABLE IF EXISTS query_buat_tabel_solusi ";
    private static final String query_hapus_tabel_rule = "DROP
TABLE IF EXISTS query_buat_tabel_rule ";
    public SQLiteHelper(Context context){super(context,
nama_database, null, versi_database);}

    @Override
    public void onCreate(SQLiteDatabase sqLiteDatabase){
        sqLiteDatabase.execSQL(query_buat_tabel_gejala);
        System.out.println("tabel_gejala sudah dibuat");
        sqLiteDatabase.execSQL(query_buat_tabel_penyakit);
        System.out.println("tabel_penyakit sudah dibuat");
        sqLiteDatabase.execSQL(query_buat_tabel_solusi);
        System.out.println("tabel_solusi sudah dibuat");
        sqLiteDatabase.execSQL(query_buat_tabel_rule);
        System.out.println("tabel_rule sudah dibuat");
    }
}
```

```

@Override
public void onUpgrade(SQLiteDatabase database, int versi_lama,
int versi_baru) {
    database.execSQL(query_hapus_tabel_gejala);
    database.execSQL(query_hapus_tabel_penyakit);
    database.execSQL(query_hapus_tabel_solusi);
    database.execSQL(query_hapus_tabel_rule);
    onCreate(database);
}

public void tambah_gejala(String nama_gejala) {
    SQLiteDatabase database = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put("nama_gejala", nama_gejala);
    database.insert("tabel_gejala", null, values);
    database.close();
}

public void tambah_penyakit(String nama_penyakit, String
definisi) {
    SQLiteDatabase database = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put("nama_penyakit", nama_penyakit);
    values.put("definisi", definisi);
    database.insert("tabel_penyakit", null, values);
    database.close();
}

public void tambah_solusi(String solusi) {
    SQLiteDatabase database = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put("solusi", solusi);
    database.insert("tabel_solusi", null, values);
    database.close();
}

public void tambah_rule(String kd_penyakit, String
nama_penyakit, String kd_gejala, String kd_solusi ) {
    SQLiteDatabase database = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put("kd_penyakit", kd_penyakit);
    values.put("nama_penyakit", nama_penyakit);
    values.put("kd_gejala", kd_gejala);
    values.put("kd_solusi", kd_solusi);
    database.insert("tabel_rule", null, values);
    database.close();
}

public ArrayList<HashMap<String, String>>
tampil_semua_gejala() {
    ArrayList<HashMap<String, String>> arrayListGejala = new
ArrayList<HashMap<String, String>>();

    SQLiteDatabase database = this.getWritableDatabase();

    Cursor cursor = database.rawQuery("SELECT * FROM
tabel_gejala", null);

```

```

if (cursor.moveToFirst()){
    do {
        // deklarasikan sebuah hashmap,
        HashMap<String, String> hashMapGejala = new
HashMap<String, String>();

        // masukkan masing-masing field dari tabel_gejala
ke dalam hashMapGejala
        //pastikan id_biodata, nama dan alamat sama persis
dengan field yang ada pada tabel gejala
        hashMapGejala.put("kd_gejala",
cursor.getString(0));
        hashMapGejala.put("nama_gejala",
cursor.getString(1));
        // masukkan hashMapBiodata ke dalam arrayListGejala
        arrayListGejala.add(hashMapGejala);

    }while (cursor.moveToNext());

    }
    return arrayListGejala;
}

public ArrayList<HashMap<String, String>>
tampil_semua_penyakit(){
    ArrayList<HashMap<String,String>> arrayListPenyakit = new
ArrayList<HashMap<String, String>>();

    SQLiteDatabase database = this.getWritableDatabase();

    Cursor cursor = database.rawQuery("SELECT * FROM
tabel_penyakit",null);

    if (cursor.moveToFirst()){
        do {
            // deklarasikan sebuah hashmap,
            HashMap<String, String> hashMapPenyakit = new
HashMap<String, String>();

            // masukkan masing-masing field dari tabel_penakit
ke dalam hashMapPenyakit
            //pastikan id_biodata, nama dan alamat sama persis
dengan field yang ada pada tabel_penakit
            hashMapPenyakit.put("kd_penyakit",
cursor.getString(0));
            hashMapPenyakit.put("nama_penyakit",
cursor.getString(1));
            hashMapPenyakit.put("definisi",
cursor.getString(2));
            // masukkan hashMapBiodata ke dalam
arrayListPenyakit
            arrayListPenyakit.add(hashMapPenyakit);

        }while (cursor.moveToNext());

    }
    return arrayListPenyakit;
}

```

```

public ArrayList<HashMap<String, String>>
tampil_semua_solusi(){
    ArrayList<HashMap<String,String>> arrayListSolusi = new
ArrayList<HashMap<String, String>>();

    SQLiteDatabase database = this.getWritableDatabase();

    Cursor cursor = database.rawQuery("SELECT * FROM
tabel_solusi",null);

    if (cursor.moveToFirst()){
        do {
            // deklarasikan sebuah hashmap, yang bisa menamp
HashMap<String, String> hashMapSolusi = new
HashMap<String, String>();

            // masukkan masing-masing field dari tabel_gejala
ke dalam hashMapGejala
            //pastikan id_biodata, nama dan alamat sama persis
dengan field yang ada pada tabel_gejala
            hashMapSolusi.put("kd_solusi",
cursor.getString(0));
            hashMapSolusi.put("solusi", cursor.getString(1));
            // masukkan hashMapBiodata ke dalam arrayListSolusi
arrayListSolusi.add(hashMapSolusi);

        }while (cursor.moveToNext());

    }
    return arrayListSolusi;
}

public ArrayList<HashMap<String, String>> tampil_semua_rule(){
    ArrayList<HashMap<String,String>> arrayListRule = new
ArrayList<HashMap<String, String>>();

    SQLiteDatabase database = this.getWritableDatabase();

    Cursor cursor = database.rawQuery("SELECT * FROM
tabel_rule",null);

    if (cursor.moveToFirst()){
        do {
            // deklarasikan sebuah hashmap, yang bisa menamp
HashMap<String, String> hashMapRule = new
HashMap<String, String>();

            // masukkan masing-masing field dari tabel_gejala
ke dalam hashMapGejala
            //pastikan id_biodata, nama dan alamat sama persis
dengan field yang ada pada tabel_gejala
            hashMapRule.put("kd_rule", cursor.getString(0));
            hashMapRule.put("kd_penyakit",
cursor.getString(1));
            hashMapRule.put("nama_penyakit",
cursor.getString(2));
            hashMapRule.put("kd_gejala", cursor.getString(3));

```

```

public int update_gejala(int kd_gejala, String nama_gejala){
    SQLiteDatabase database = this.getWritableDatabase();
    ContentValues recordGejala = new ContentValues();
    recordGejala.put("nama_gejala",nama_gejala);
    return database.update("tabel_gejala", recordGejala,
"kd_gejala=" +kd_gejala, null);
}

public int update_penyakit(int kd_penyakit, String
nama_penyakit, String definisi){
    SQLiteDatabase database = this.getWritableDatabase();
    ContentValues recordPenyakit = new ContentValues();
    recordPenyakit.put("nama_penyakit",nama_penyakit);
    recordPenyakit.put("definisi", definisi);
    return database.update("tabel_penyakit", recordPenyakit,
"kd_penyakit=" +kd_penyakit, null);
}

public int update_solusi(int kd_solusi, String solusi){
    SQLiteDatabase database = this.getWritableDatabase();
    ContentValues recordSolusi = new ContentValues();
    recordSolusi.put("solusi",solusi);
    return database.update("tabel_solusi", recordSolusi,
"kd_solusi=" +kd_solusi, null);
}

public int update_rule(int kd_rule, String kd_penyakit, String
nama_penyakit, String kd_gejala, String kd_solusi){
    SQLiteDatabase database = this.getWritableDatabase();
    ContentValues recordRule = new ContentValues();
    recordRule.put("kd_penyakit",kd_penyakit);
    recordRule.put("nama_penyakit", nama_penyakit );
    recordRule.put("kd_gejala", kd_gejala);
    recordRule.put("kd_solusi", kd_solusi);
    return database.update("tabel_rule", recordRule, "kd_rule="
+kd_rule, null);
}

public void hapus_gejala(int kd_gejala){
    SQLiteDatabase database = this.getWritableDatabase();
    database.execSQL("DELETE FROM tabel_gejala WHERE
kd_gejala='" + kd_gejala + "'");
    database.close();
}

public void hapus_penyakit(int kd_penyakit){
    SQLiteDatabase database = this.getWritableDatabase();
    database.execSQL("DELETE FROM tabel_penyakit WHERE
kd_penyakit='" +kd_penyakit+ "'");
    database.close();
}

public void hapus_solusi(int kd_solusi){
    SQLiteDatabase database = this.getWritableDatabase();
    database.execSQL("DELETE FROM tabel_solusi WHERE
kd_solusi='" +kd_solusi+ "'");
    database.close();
}

```



```

public void hapus_rule(int kd_rule){
    SQLiteDatabase database = this.getWritableDatabase();
    database.execSQL("DELETE FROM tabel_rule WHERE kd_rule='"
+kd_rule+ "'");
    database.close();
}

public HashMap<String, String> tampil_gejala_berdasarkan_kd(int
kd_gejala) {

    SQLiteDatabase database = this.getReadableDatabase();

    HashMap<String, String> hashMapGejala = new HashMap<String,
String>();

    Cursor cursor = database.rawQuery("SELECT * FROM
tabel_gejala WHERE kd_gejala=" + kd_gejala + "", null);

    if (cursor.moveToFirst()) {
        do {
            hashMapGejala.put("kd_gejala",
cursor.getString(0));
            hashMapGejala.put("nama_gejala",
cursor.getString(1));
        } while (cursor.moveToNext());
    }

    return hashMapGejala;
}

public HashMap<String, String>
tampil_penyakit_berdasarkan_kd(int kd_penyakit) {

    SQLiteDatabase database = this.getReadableDatabase();

    HashMap<String, String> hashMapPenyakit = new
HashMap<String, String>();

    Cursor cursor = database.rawQuery("SELECT * FROM
tabel_penyakit WHERE kd_penyakit=" + kd_penyakit + "", null);

    if (cursor.moveToFirst()) {
        do {
            hashMapPenyakit.put("kd_penyakit",
cursor.getString(0));
            hashMapPenyakit.put("nama_penyakit",
cursor.getString(1));
            hashMapPenyakit.put("definisi",
cursor.getString(2));
        } while (cursor.moveToNext());
    }

    return hashMapPenyakit;
}

```

```

public HashMap<String, String> tampil_solusi_berdasarkan_kd(int
kd_solusi) {

    SQLiteDatabase database = this.getReadableDatabase();

    HashMap<String, String> hashMapSolusi = new
HashMap<String, String>();

    Cursor cursor = database.rawQuery("SELECT * FROM
tabel_solusi WHERE kd_solusi=" + kd_solusi + "", null);

    if (cursor.moveToFirst()) {
        do {
            hashMapSolusi.put("kd_solusi",
cursor.getString(0));
            hashMapSolusi.put("solusi",
cursor.getString(1));
        } while (cursor.moveToNext());
    }

    return hashMapSolusi;
}

public HashMap<String, String>
tampil_rule_berdasarkan_kd(int kd_rule) {

    SQLiteDatabase database = this.getReadableDatabase();

    HashMap<String, String> hashMapRule = new
HashMap<String, String>();

    Cursor cursor = database.rawQuery("SELECT * FROM
tabel_rule WHERE kd_rule=" + kd_rule + "", null);

    if (cursor.moveToFirst()) {
        do {
            hashMapRule.put("kd_rule",
cursor.getString(0));
            hashMapRule.put("kd_penyakit",
cursor.getString(1));
            hashMapRule.put("nama_penyakit",
cursor.getString(2));
            hashMapRule.put("kd_gejala",
cursor.getString(3));
            hashMapRule.put("kd_solusi",
cursor.getString(4));
        } while (cursor.moveToNext());
    }

    return hashMapRule;
}

}

```