

**O'ZBEKISTON RESPUBLIKASI ALOQA,
AXBOROTLASHTIRISH VA TELEKOMMUNIKATSIYA
TEXNOLOGIYALARI DAVLAT QO'MITASI
TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI**

“Dasturiy injiniring” fakulteti

Travel By Bus mavzusida

**KURS ISHI
XISOBOTI**

Topshirdi: 215-12 guruh talabasi

U.O. Mamajonov

Qabul qildi: N.A. Karaxanov

Toshkent 2015

MUNDARIJA

I. KIRISH	3
II. NAZARIY QISM	4
2.1. Java dasturlash tili haqida.....	4
2.2. Ma'lumotlar bazasi tuzilmasi.....	5
2.3. Android platformasi haqida.....	6
2.4. XML bilan ishlash	7
2.5. Eclipse platformasi haqida.....	8
III.AMALIY QISM	10
3.1. Dastur ma'lumotlar bazasi tuzilmasi.....	10
3.2. Dastur algoritmi.....	11
3.3. UML sxemasi.....	12
3.4. Dasturdan foydalanish	13
IV. XULOSA.....	17
V. FOYDALANILGAN ADABIYOTLAR.	18
VI. ILOVA	19
6.1. Dastur kodi.....	19

*Buyuk maqsadlarimizga, ezgu niyyatlarimizga
erishishimiz, jamiyatimizning yangilanishi, xayoti-
timiz taraqqiyoti va istiqboli, amalga oshirilayotgan
isloxotlarimiz va rejalarimizning samarali
taqdiri, avvalambor, davr talablariga javob
beradigan yuqori malakali, ongli, tafakkurga
ega bo'lgan mutaxassis kadrlar tayyorlash
muammosi bilan bog'liq...*

I.A.Karimov

I. KIRISH

Biz bugun mobil aloqa, internet, videotelefon tizimi, raqamli televideniya, zamonaviy bank xizmatlari, lizing, sug'urta, audit, injiniring va boshqa zamonaviy xizmat turlarini rivojlantirishga alohida e'tibor qaratmoqdamiz.

Bugungi kunda yurtimiz mobil aloqaning rivojlanish ko'rsatkichi bo'yicha dunyoda oldingi o'rinda turgan o'nta mamlakat qatoriga kiradi. O'zbekistonda mobil aloqa xizmatidan foydalanuvchilarning umumiy soni 2000 yilga nisbatan – 200 barobar oshib, 19 milliondan ortiq abonentni tashkil etmoqda.

Yuqoridagi ma'lumotlardan ko'rinib turibiki Android operatsion tizimida ishlab chiqarilgan dasturlar XXI asrdagi eng ko'p ishlatiladigan, ommalashib borayotgan dasturlar sarasiga kiradi. Demak, O'zbekiston uchun Android tizimida yaratilgan milliy - zamonaviy dasturlar ishlab chiqish informatika va axborot texnologiyalari sohasidagi barcha dasturlovchilar uchun birinchi darajali masalalardan hisoblanadi. Shu maqsadda men ham Android tizimida ishlovchi Toshkent avtobus, marshrut va metrolaridan foydalanish uchun off-line rejimida ishlovchi ma'lumot beruvchi dastur yaratishni lozim deb topdim.

Datur Toshkent shaxriga kelgan mehmonlar, talabalar va axoli uchun mo'ljallanadi. Foydalanuvchiga qulaylik yaratish, vaqtdan yutish imkonini beradi. Internet, GPS xizmatlari mavjud bo'lmagan hollarda ham foydalanish mumkin. Dastur kiritilgan manzillar orasida qatnovchi avtobus va marshrutlar ro'yxatini chiqarib beradi.

II. NAZARIY QISM

2.1. Java dasturlash tili haqida

Tarixi: Java dasturlash tili — eng yaxshi dasturlash tillaridan biri bo‘lib unda korporativ darajadagi mahsulotlarni(dasturlarni) yaratish mumkin. Bu dasturlash tili Oak dasturlash tili asosida paydo bo‘ldi. Oak dasturlash tili 90-yillarning boshida Sun Microsystems tomonidan platformaga(Operatsion tizimga) bog‘liq bo‘lmagan holda ishlovchi yangi avlod aqlli qurilmalarini yaratishni maqsad qilib harakat boshlagan edi. Bunga erishish uchun Sun hodimlari C++ ni ishlatishni rejalashtirdilar, lekin ba‘zi sabablarga ko‘ra bu fikridan voz kechishdi. Oak muvofaqiyatsiz chiqdi va 1995-yilda Sun uning nomini Java ga almashtirdi, va uni WWW rivojlanishiga hizmat qilishi uchun ma‘lum o‘zgarishlar qilishdi.

Java Obyektga Yo‘naltirilgan Dasturlash(OOP-object oriented programming) tili va u C++ ga ancha o‘xshash. Eng ko‘p yo‘l qo‘yildigan xatolarga sabab bo‘luvchi qismlari olib tashlanib, Java dasturlash tili ancha soddalashtirildi.

Java kod yozilgan fayllar(*.java bilan nihoyalayuvchi) kompilatsiyadan keyin bayt kod(bytecode) ga o‘tadi va bu bayt kod interpretator tomonidan o‘qib yurgizdiriladi.

Imkoniyatlari: Bugungi kunga kelib kompyuter va Internetda ishlovchi ko‘plab foydalanuvchilar Java haqida bir oz bo‘lsada tushunchaga ega bo‘lganlar.

Java dasturlash tilida C va C++ tilining konstruksiyalari, operatorlari va funksiyalaridan yetarlicha foydalanilgan bo‘lib, u yangicha uslubdagi ob‘yektga yo‘naltirilgan tamoyillar kiritilgan. Unda dastur tuzish va o‘rganishning osonligi, yaratilgan dasturlarning ixtiyoriy platformada ishlata olish, Internet uchun ixcham dasturlar tuzish, xotirani “chiqindi”lardan avtomatik tozalash, imkoniyatining mavjudligi uning ommabopligini oshirdi. Java dasturlari Java virtual mashinasi (JVM) deb ataluvchi mexanizmning bayt-kodlari orqali kompilyatsiya qilinganligi uchun ixtiyoriy operatsion tizimda ishlay oladi. Shu sababli, Java dasturlarining ishga tushirilishida vaqt ko‘proq talab qilinadi. Ko‘pchilik foydalanuvchilar uni shu tomonlama tanqid ostiga olishadi.

Java – tilining asosiy yutuqlaridan biri uning tez o'zgaruvchanligidir. Bu til dasturlash muhitiga va dasturlash yondashuvlariga tez moslasha oladi. Undan nafaqat ilovalar yaratishda, balki Internet uchun dasturlar yaratishda ham foydalanish mumkin.

Xavfsizlik (ishonch yuq kodni xavfsiz ishga tushirish).

Xotirani xavfsiz boshqarish (avtomat ravishda keraksiz ma'lumotlarni yig'adi)

Tarmoqda dasturlash

Ko'p oqimli (Multi-thread) dasturlash

Quyidagi farqlar bilan Java C/C++ dan ajralib turadi

- ✓ header fayllar yo'q
- ✓ Preprocessorlar yo'q
- ✓ Goto yo'q
- ✓ Unicode belgilar
- ✓ Avtomat musorlarni yig'adi
- ✓ Ko'rsatkich (pointer) yo'q
- ✓ Operatorlarni qayta yuklash qilish yo'q
- ✓ Dasturda barcha vazifalar class tushunchasi bilan boshlanadi
- ✓ Global o'zgaruvchi va funksiyalar yo'q

2.2.Ma'lumotlar bazasi tuzilmasi

Relyatsion ma'lumotlar bazasini boshqarish tizimi - o'zining ma'lumotlar strukturasi oddiyligi bilan, foydalanuvchi uchun jadval ko'rinishida joylashtirilishi bilan va ma'lumotlar ustidan oson hisob-kitob amallarini bajarish imkoniyati mavjudligi bilan ajralib turadi.

Hozirda relyatsion ma'lumot bazalari o'zining qulayligi tufayli keng miqyosda ishlatilmoqda. Relyatsion ma'lumot bazalari ma'lumotlarni jadvallarga joylashgan va jadvallar orasida mos bog'liqliklarni, ya'ni munosabatni (relyatsiyani) o'rnatishga asoslangan. Ular jadvallar orasidagi turli bog'liqliklarni o'rnatish, ma'lumot kiritish shakllarini yaratish, hisobot shakllarini chiqarish, turli so'rovlar (Zaproosi) tuzish imkonini beradi.

Relation model ma'lumotlarni ikki o'lchamli jadvalda tartiblashga asoslangan. Har bir relation jadval ikki o'lchamli massivdan iborat bo'ladi va quyidagi hususiyatlarga ega:

-Jadvalning bir ustunidagi barcha kataklar bir hil tipga mansub(misol uchun bir ustun barcha elementlari tipi simvolli yoki raqamli);

-Jadvalda bir xil qatorlar mavjud emas;

-Ustun va qatorlar ketmaketligi turlicha bo'lishi mumkin;

Relation ma'lumotlar bazasi boshqarish tizimining asosiy tushunchalari bu:

* Atribut;

* Relation;

* Kortej;

2.3. Android platformasi haqida

Android (yunoncha so'z) odam sifat robot ma'nosini ifoda etadi. Balki mana shuning uchun Android operatsion tizimi logotipida robot tasvirlangandir.

Android operatsion tizimining yaratilishi tarixi 2002-yillardan boshlangan. Mana shu davrda Google korporatsiyasi yaratuvchilari E. Rubinning dasturiy ishlanmalari to'plami bilan qiziqib qoladilar. Dastlab mobil qurilmalar uchun yangi operatsion tizimni yaratish loyihasi bilan katta maxfiylik ostida Android Inc. Kompaniyasi shug'ullangan, ushbu kompaniyani keyinchalik Google sotib oladi.

Android Linux yadrosiga asoslangan kommunikatorlar, planshetli kompyuterlar, elektron kitoblar, raqamli musiqa uskunalari, qo'l soatlari, netbuklar va smartbuklar uchun portativ (tarmoqli) operatsion tizimdir.

Keyinchalik Google Open Handset Alliance (OHA) alyansini tashkil qildi, u hozirda ham platformani qo'llab-quvvatlash va yanada rivojlantirish bilan shug'ullanadi. Android Google tomonidan ishlab chiqilgan kutubxona orqali qurilmani boshqaruvchi Java-ilovasini yaratishga imkon beradi. Android Native Development Kit Si va boshqa tillarda yozilgan ilovalarni yaratadi.

2012-yilning uchinchi choragida sotilgan smartfonlarning 75 foizida Android operatsion tizimi o'rnatilgan.

2.4.XML bilan ishlash

Android da XML lug'atdan foydalangan holda foydalanuvchi interfeysi hamda ekran elementlarini loyihalash mumkin. Bunda ham HTML Web-sahifa yaratish usullari qo'llaniladi.

Har bir XML fayli View yoki ViewGroup obyektini bo'lgan 1 ta asosiy element saqlaydi. Asosiy element aniqlangach, unga qo'shimcha elementlar yoki vidjetlar qo'shish mumkin. Bu elementlarni esa o'z navbatida doimiy ravishda tahrirlash mumkin. Misol:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/Button1"
        android:text="Ko'rsatish"
    </Button>
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="ResUzActivity"
    </TextView>
</LinearLayout>
```

Ushbu misolda 1 ta asosiy element, ya'ni <LinearLayout> va o'z atributlariga ega bo'lgan 1 ta ost element hisoblangan <TextView>, ya'ni matnli maydon mavjud.

xmlns:android Android muhitida umumiy atributlarni XML da e'lon qilish. Har bir fayldagi asosiy element o'zida quyidagi qiymatli atributni saqlashi lozim:

android: layout_width Ushbu atribut ekran uchun ruxsat etilgan kengligini ifodalaydi. Yuqoridagi misolda TextView yagona obyekt bo'lganligi uchun ham uni butun ekran bo'ylab cho'zish mumkin. Shuning uchun ham uning qiymati "fill_parent" ni tashkil etadi.

android: layout_height mos ravishda ekran uzunligini belgilaydi.

android: text TextView da ifodalanishi lozim bo'lgan matnni o'rnatadi.

View va ViewGroup ning har bir obykti xilma-xil XML-atributlarni qo'llab-quvvatlaydi.

Standart belgilar

Ilova yaratishda quyidagi standart belgilardan foydalaniladi:

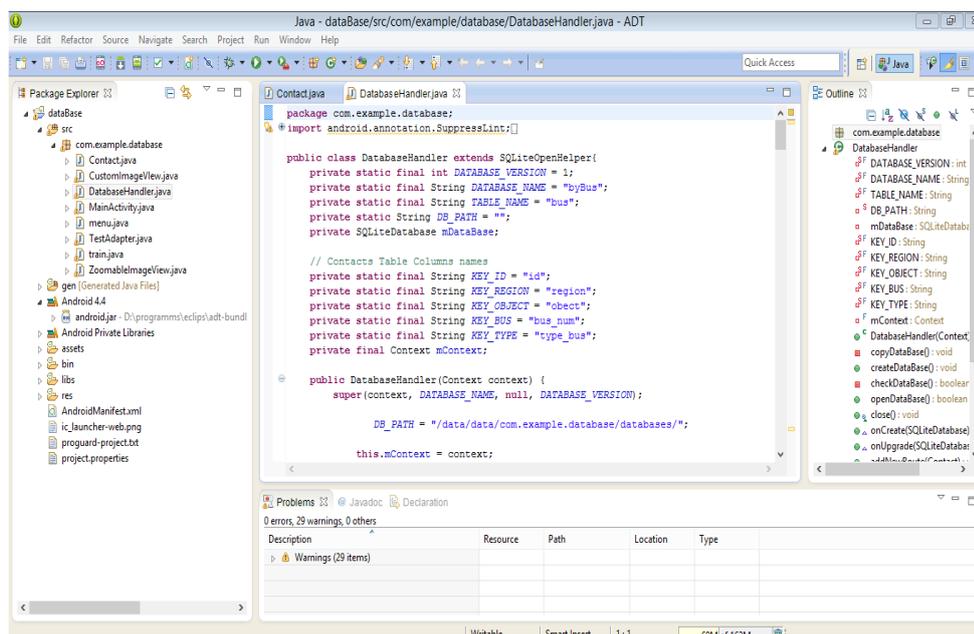
- `FrameLayout`;
- `LinearLayout`;
- `TableLayout`;
- `RelativeLayout`;

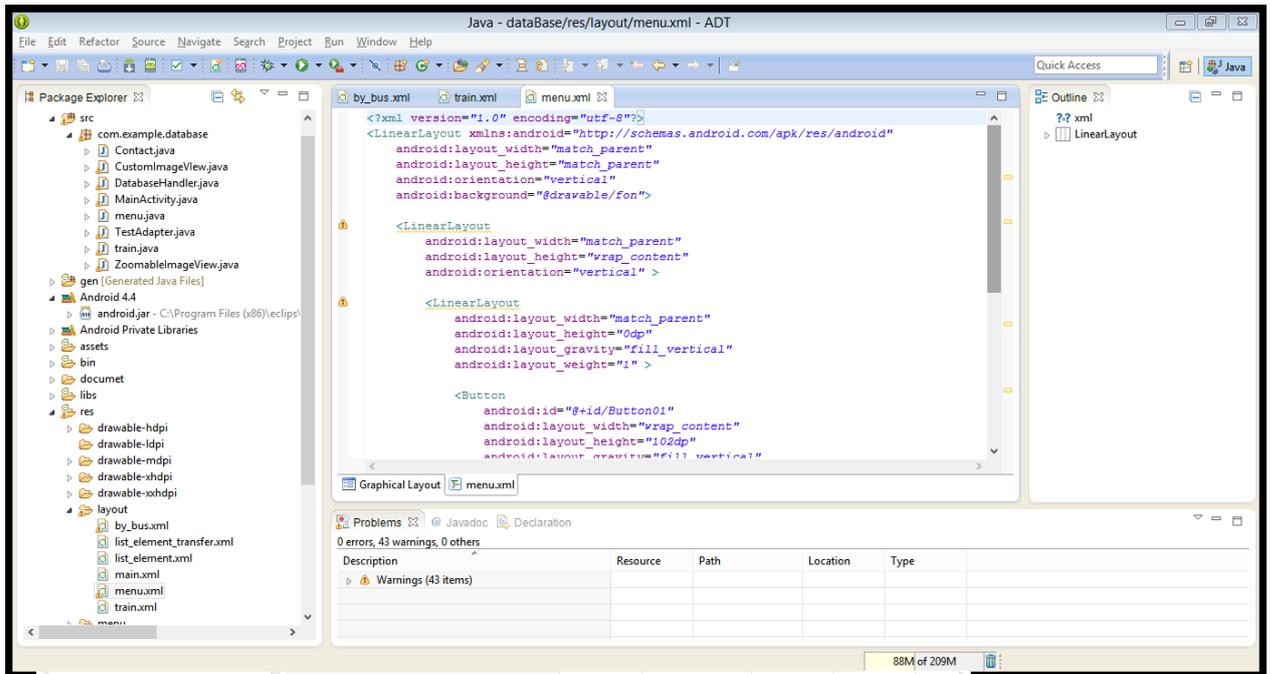
2.5. Eclipse platformasi haqida

Dastur “Eclipse” platformasida yaratildi

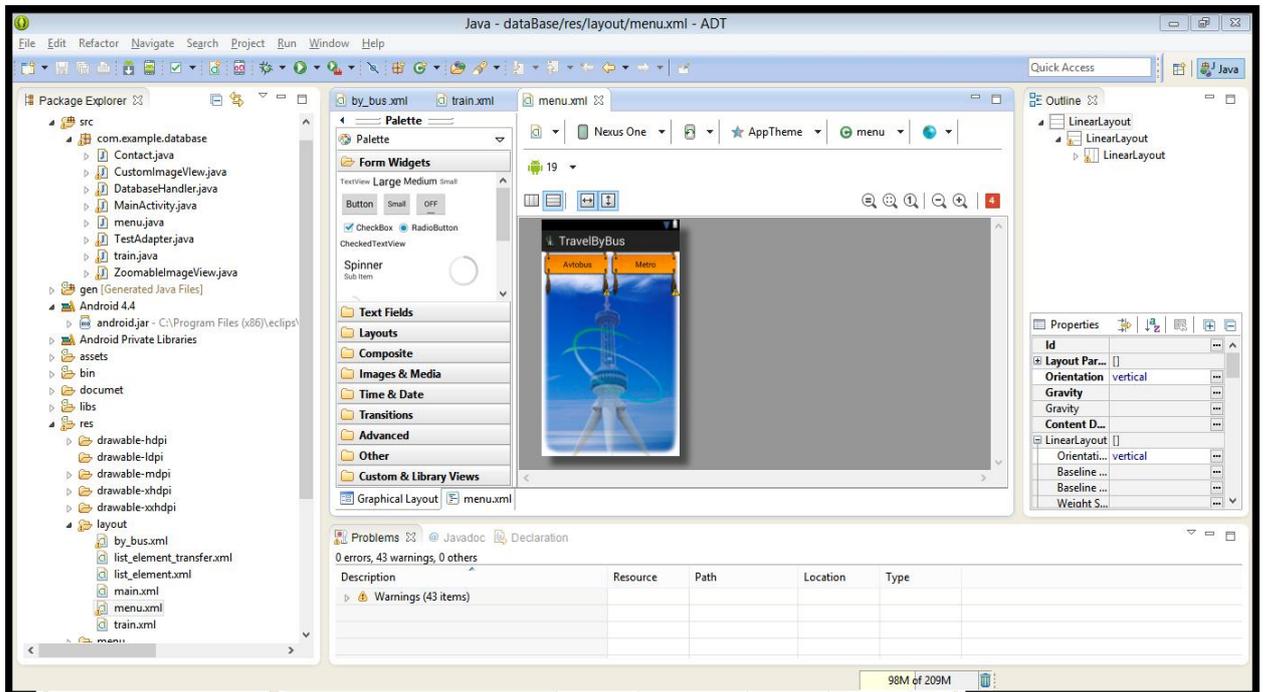


Eclipse muxitining ko'rinishi





Dasturning yaratilish jarayoni



III.AMALIY QISM

3.1. Dastur ma'lumotlar bazasi tuzilmasi

“SQLite” da yaratilgan

RecNo	id	region	oboct	bus_num	type_bus	obj
1	1	Yunusobod	Universam bozori	123	Avtobus	1
2	2	Yunusobod	Universam bozori	136	Avtobus	0
3	3	Yunusobod	Universam bozori	82	Avtobus	0
4	4	Yunusobod	Universam bozori	83	Avtobus	0
5	5	Yunusobod	Universam bozori	96	Avtobus	0
6	6	Yunusobod	Universam bozori	111	Avtobus	0
7	7	Yunusobod	Universam bozori	123	Avtobus	0
8	8	Yunusobod	Universam bozori	133	Avtobus	0
9	9	Yunusobod	Universam bozori	149	Avtobus	0
10	10	Yunusobod	Universam bozori	2 M	Avtobus	0
11	11	Yunusobod	Universam bozori	6 M	Avtobus	0
12	12	Yunusobod	Universam bozori	90 M	Avtobus	0
13	13	Yunusobod	Universam bozori	186 M	Avtobus	0
14	14	Yunusobod	Universam bozori	24	Avtobus	0
15	15	Yunusobod	Universam bozori	50	Avtobus	0
16	16	Yunusobod	Universam bozori	51	Avtobus	0
17	17	Yunusobod	Universam bozori	60	Avtobus	0
18	18	Yunusobod	Universam bozori	63	Avtobus	0
19	19	Yunusobod	Universam bozori	71	Avtobus	0

Metro bekatlari jadvali

id	Yonalishlar	Bekatlari	Yaqin obektlar
1	O'zbekiston yo'li	Beruniy	O'zmu, Mashxura Land,
		Tinchlik	Tinchlik supermarketi Registon restorani Guncha kinoteatri GANO glavniy offis

3.2. Dastur algoritmi

Bo'limlar:

I. Avtobuslar

1.1. Birinchi manzil

- a) Tuman
- b) Obekt

1.2. Ikkinchi manzil

- a) Tuman
- b) Obekt

1.3. Birinchi va ikkinchi obektlar o'rtasidagi to'g'ridan - to'g'ri avtobuslar va marshrutkalar ro'yxatini chiqarish tugmasi

1.4. Agar birinchi va ikkinchi obektlar o'rtasidagi to'g'ridan – to'g'ri avtobuslar yoki marshrutkalar mavjud bo'lmasa almashinishlar ro'yxatini chiqarish

1.5. Yangilash

I. Metro

2.1. Toshkent metropoliteni xaritasi

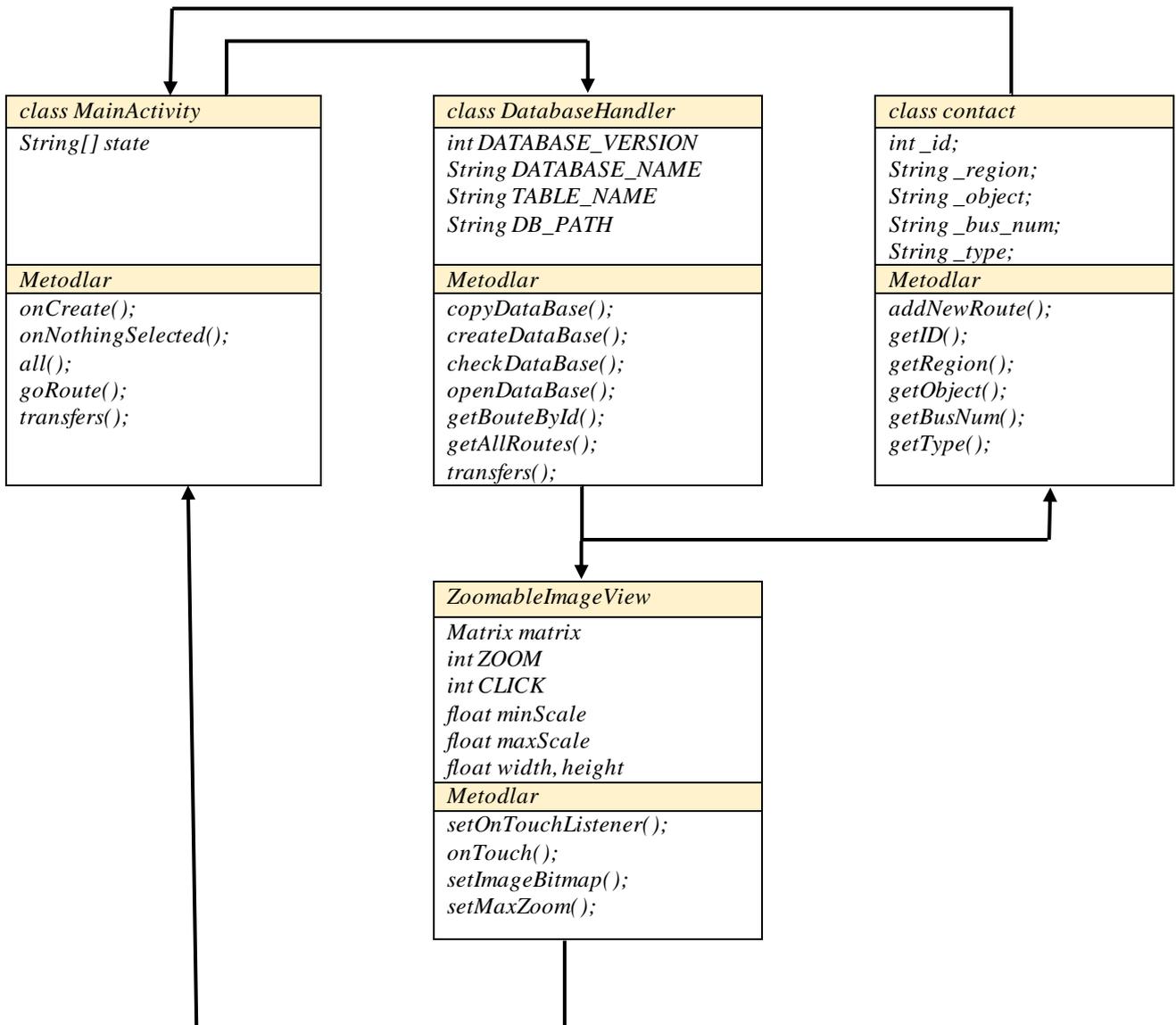
2.2. Har bir metro stansiyasi tanlanganda shu stansiya yaqinidagi asosiy obektlar ro'yxatini ko'rsatish

Misol: Bodomzor ->

- 1.NBU
- 2.Interkontenental mexmononasi
- 3.Aqua park
- 4.Toshkent Lend
- 5.Axborot Texnologiyalari
Universiteti

3.3. UML sxemasi

UML - bu yangi Unifikatsiyalangan modellashtirish tili (Unified Modeling Language) bo'lib, Gradi Buch, Ayvar Yakobson va Djejms Rambo (Grady Booch, Ivar Jacobson, James Rumbaugh) tomonidan ob'ektga-yo'naltirilgan loyihalash va taxlilni xujjatlashning yangi standarti sifatida taklif qilingan.



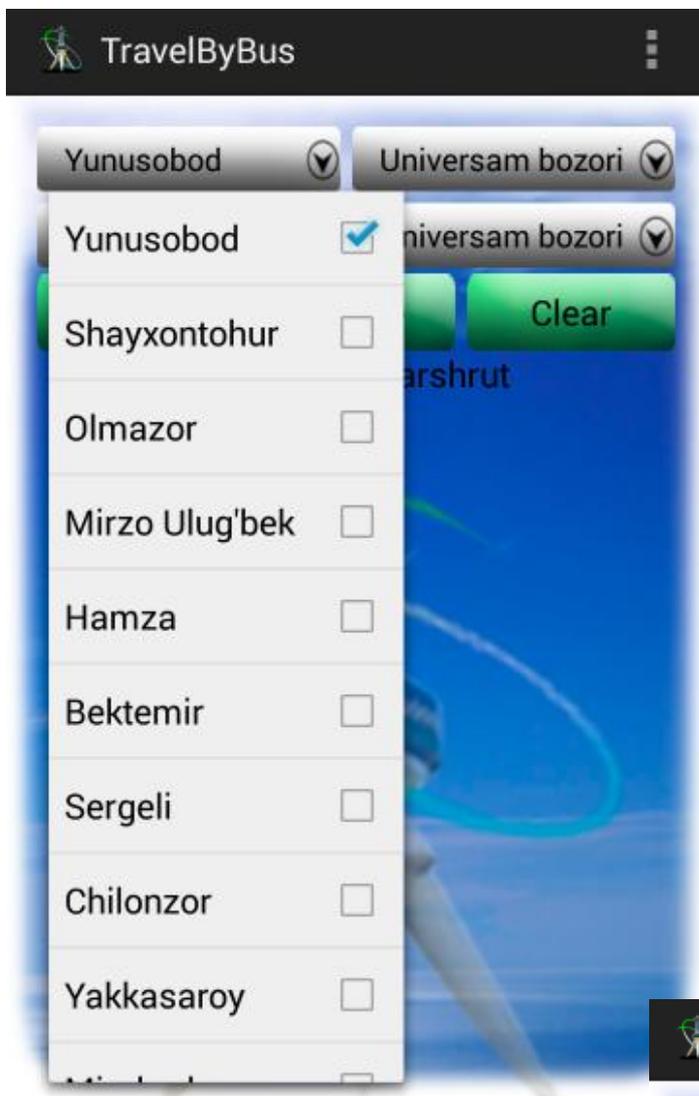
3.4. Dasturdan foydalanish



Bosh oynaning umumiy ko'rinishi

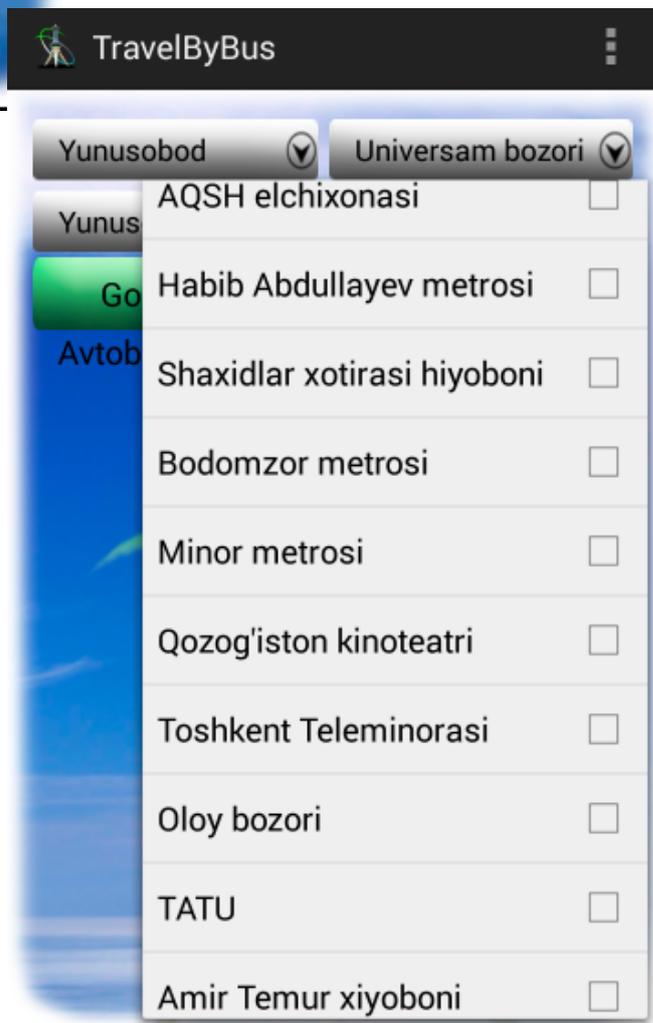
Bu yerdan

- Avtobus
 - Metro
- bo'limlariga o'tish mumkin



Avtobus bo'limi:

Birinchi tumanni tanlash



Avtobus bo'limi:

Birinchi obektni tanlash



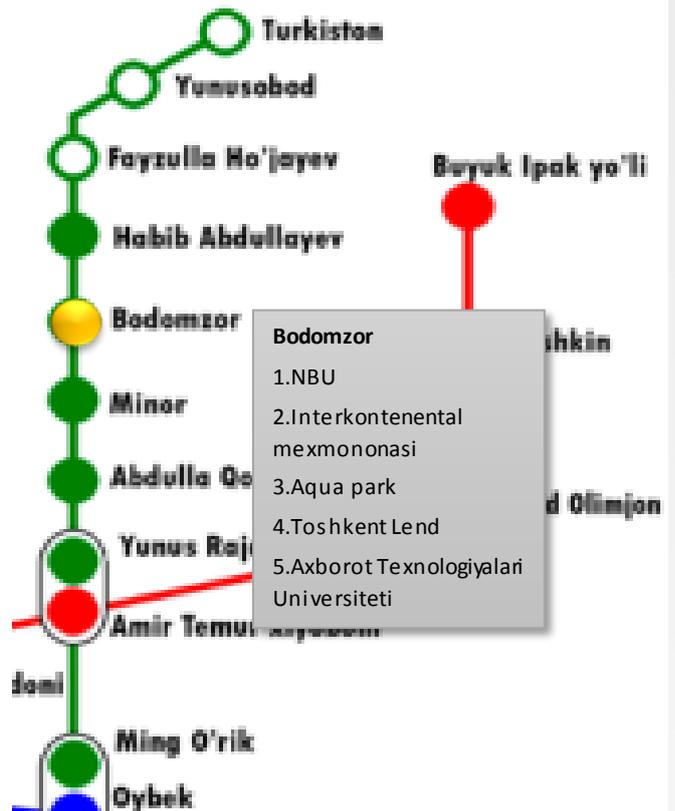
Avtobus bo'limi:

Tanlangan obektlar o'rtasida harakatlanuvchi
transportlar ro'yxati



Toshkent metrosi xaritasi

METROPOLITENI



Metro bekatlari yaqinidagi binolar

IV. XULOSA

Bugungi kun har qanday masalaning yechimi kompyuter yordamida hal qilinayotgan bir davr hisoblanadi. Hozirda hech bir soha yo'qki unda kompyuter, internet foydalanilmaydigan. Har bir sohaga chuqur kirib borayotga yangi texnologiyalar, insoniyat taraqqiyotiga o'zining bemiul ulushuni qo'shmoqda.

Shuni ham ta'kidlash lozimki, hozirda axborot sohasini rivojlantirish va faoliyatini yaxshilash bo'yicha ham O'zbekistonda bir qator Prezident Farmonlari va qonun hujjatlari ishlab chiqilmoqda. Bular esa o'z navbatida aholi ongini oshirish va malakali servis xizmatni tashkil qilishga qaratilgandir.

Xulosa qilib aytganda, hozirgi kunda axborotni istalغان vaqtda va ishonchli manba orqali qabul qilish va undan samarali foydalanish eng asosiy masaladir. Ayniqsa, ommalashib borayotgan mobile texnologiya sohasida.

Yurtimizdagi mobil texnologiya sohasidagi o'zgarishlarni olib qarasak, boshqa hech bir davlatdan qolishmagan holda endilikda O'zbekistonda ham Android operatsion tizimida ishlovchi telefonlar ishlab chiqarilishi biz yoshlarda faxr tuyg'usini uyg'otadi.

V. FOYDALANILGAN ADABIYOTLAR

1. J.F.Di Marzio Android A Programmer's Guide 2008.
2. Chris Haseman Android Essentials 2008.
3. Mark Murphy Beginning Android 2 ;
4. Jeff Friesen Learn Java for Android Development ;
5. <http://www.java.sun.com/javase/downloads/index.jsp>;
6. <http://www.eclipse.org/downloads/>;
7. <http://www.developer.android.com/sdk/index.html>;

VI. ILOVA

6.1. Dastur kodi

```
package com.example.database;

import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.nio.channels.FileChannel;
import java.util.List;

import android.app.Activity;
import android.content.Context;
import android.database.Cursor;
import android.opengl.Visibility;
import android.os.Bundle;
import android.os.Environment;
import android.util.Log;
import android.view.Menu;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends Activity {
    ArrayAdapter<String> adapter;
    ArrayAdapter<String> adapter2;

    private String[] state = { "Yunusobod",
                               "Shayxontohur",
                               "Olmazor",
                               "Mirzo Ulug'bek",
                               "Hamza",
                               "Bektemir",
                               "Sergeli",
                               "Chilonzor",
                               "Yakkasaroy",
                               "Mirobod",
                               "Uchtepa"
    };
};
```

```

TextView selVersion;
Spinner sp1, sp2, sp3, sp4;

DatabaseHandler db = new DatabaseHandler(this);

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

    TestAdapter mDbHelper = new TestAdapter(this);
    mDbHelper.createDatabase();
    mDbHelper.open();

    mDbHelper.close();

    Button rpBtn = (Button) findViewById(R.id.rp_btn);
    // rpBtn.setEnabled(false);
    //-- lLay = (LinearLayout)findViewById(R.id.lay);
    //-- lLay2 = (LinearLayout)findViewById(R.id.trlay);
    //-- lLay2.setVisibility(TRIM_MEMORY_UI_HIDDEN);

    lv = (ListView)findViewById(R.id.lv);
    adapter=new ArrayAdapter<String>(this, R.layout.list_element);
    lv.setAdapter(adapter);

    lv2 = (ListView)findViewById(R.id.lv2);
    adapter2=new ArrayAdapter<String>(this, R.layout.list_element);
    lv2.setAdapter(adapter2);

    //-- lvtrans = (ListView)findViewById(R.id.listTransf);
    //-- adapter3=new ArrayAdapter<String>(this, R.layout.list_element_transfer);
    //-- lvtrans.setAdapter(adapter3);

    selVersion = (TextView) findViewById(R.id.textView1);
    sp1 = (Spinner) findViewById(R.id.sp1);
    sp2 = (Spinner) findViewById(R.id.sp2);
    sp3 = (Spinner) findViewById(R.id.sp3);
    sp4 = (Spinner) findViewById(R.id.sp4);

    adapter_state = new
ArrayAdapter<String>(this,android.R.layout.simple_spinner_item, state);
adapter_state.setDropDownViewResource(android.R.layout.simple_list_item_multiple_choice);

```

```

        sp3.setAdapter(adapter_state);
        sp1.setOnItemSelectedListener(new OnItemSelectedListener(){
@Override
public void onItemSelected(AdapterView<?> parent, View view, int position, long id)
        {
                String selState1 = (String) sp1.getSelectedItem();
                setObj(sp2, selState1);
        }

@Override
public void onNothingSelected(AdapterView<?> arg0) {
Toast.makeText(getApplicationContext(),"Xato so'rov",Toast.LENGTH_LONG).show();
        }
        });
        sp1.setAdapter(adapter_state);
        sp3.setOnItemSelectedListener(new OnItemSelectedListener(){
@Override
public void onItemSelected(AdapterView<?> parent, View view, int position, long id)
        {
                String selState1 = (String) sp3.getSelectedItem();
                setObj(sp4, selState1);
        }

@Override
public void onNothingSelected(AdapterView<?> arg0) {
Toast.makeText(getApplicationContext(),"Xato so'rov",Toast.LENGTH_LONG).show();
        }
        });
}
public void clearList(View v)
{
        adapter.clear();
        adapter2.clear();
}
public void setObj(Spinner sp, String str)
{
        List<String> contacts = db.getObeject(str);

adapter_obj = new ArrayAdapter<String>(this,android.R.layout.simple_spinner_item, contacts);
adapter_obj.setDropDownViewResource(android.R.layout.simple_list_item_multiple_choice);
        sp.setAdapter(adapter_obj);

}

@Override

```

```

public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.main, menu);

    return true;
}

public void add(View v)
{
    EditText e_id = (EditText)findViewById(R.id._id);
    EditText e_region = (EditText)findViewById(R.id._region);
    EditText e_object = (EditText)findViewById(R.id._object);
    EditText e_bus_num = (EditText)findViewById(R.id._num_bus);
    EditText e_type = (EditText)findViewById(R.id._type);

    int ild = Integer.parseInt(e_id.getText().toString());
    String iRegion = e_region.getText().toString();
    String iObject = e_object.getText().toString();
    String iBusNum = e_bus_num.getText().toString();
    String iType = e_type.getText().toString();

    db.addNewRoute(new Contact(ild, iRegion, iObject, iBusNum, iType));

    e_id.setText("");
    e_region.setText("");
    e_object.setText("");
    e_bus_num.setText("");
    e_type.setText("");
}

public void all(View v)
{
    List<Contact> contacts = db.getAllRoutes();
    ListView lv = (ListView)findViewById(R.id.lv);
    String myBus;
    for (Contact cn : contacts)
    {
        myBus = cn.getBusNum();
        adapter.add(myBus);
    }
}

public void goRoute(View v)
{
    EditText e_region = (EditText)findViewById(R.id._region);
    EditText e_object = (EditText)findViewById(R.id._object);
    EditText e_bus_num = (EditText)findViewById(R.id._num_bus);
    EditText e_type = (EditText)findViewById(R.id._type);
}

```

```

//this.lv2.setVisibility(TRIM_MEMORY_COMPLETE);
lv.setVisibility(View.VISIBLE);
lv2.setVisibility(View.VISIBLE);

sp1 = (Spinner) findViewById(R.id.sp1);
sp2 = (Spinner) findViewById(R.id.sp2);
sp3 = (Spinner) findViewById(R.id.sp3);
sp4 = (Spinner) findViewById(R.id.sp4);

String adr1 = (String) sp1.getSelectedItem();
String obj1 = (String) sp2.getSelectedItem();
String adr2 = (String) sp3.getSelectedItem();
String obj2 = (String) sp4.getSelectedItem();

List <Contact> contacts = db.go(adr1, obj1, adr2, obj2);
ListView lv = (ListView)findViewById(R.id.lv);

adapter.clear(); adapter2.clear();
String myBus, myBusType;
String arg= "Avtobus";
for (Contact cn : contacts)
{
    myBus = cn.getBusNum();
    myBusType = cn.getType();
    // Log.d("Test",myBusType);

    if (myBusType.contentEquals(arg))
        adapter.add(" "+myBus);
    else
        adapter2.add(" "+myBus);
}
}
public void transfers(View v)
{
    EditText e_region = (EditText)findViewById(R.id._region);
    EditText e_object = (EditText)findViewById(R.id._object);
    EditText e_bus_num = (EditText)findViewById(R.id._num_bus);
    EditText e_type= (EditText)findViewById(R.id._type);

    //this.lv2.setVisibility(TRIM_MEMORY_UI_HIDDEN);

    lv2.setVisibility(View.INVISIBLE);
    lv.setVisibility(View.VISIBLE);
    sp2 = (Spinner) findViewById(R.id.sp2);
    sp4 = (Spinner) findViewById(R.id.sp4);
    String obj1 = (String) sp2.getSelectedItem();

```

```

String obj2 = (String) sp4.getSelectedItem();
List<Contact> contacts = db.transfers(obj1, obj2);
String myBus="", object="", myBus2="", object2="";
// String arg= "Avtobus";
boolean b=true;
for (Contact cn : contacts)
{
    if(b)
    {
        myBus = cn.getBusNum();
        object = cn.getObject();
        b=false;
    }
    else
    {
        myBus2 = cn.getBusNum();
        object2 = cn.getObject();
        b=true;
    }
    if(object == object2)
        adapter.add(myBus2+" "+object+" "+myBus);
}
}

public void update(View v)
{
    EditText e_id = (EditText)findViewById(R.id._id);
    EditText e_region = (EditText)findViewById(R.id._region);
    EditText e_object = (EditText)findViewById(R.id._object);
    EditText e_bus_num = (EditText)findViewById(R.id._num_bus);
    EditText e_type = (EditText)findViewById(R.id._type);

    Contact con= new Contact();
    con.setID(Integer.parseInt(e_id.getText().toString()));
    con.setRegion(e_region.getText().toString());
    con.setObject(e_object.getText().toString());
    con.setBusNum(e_bus_num.getText().toString());
    con.setType(e_type.getText().toString());
    db.updateContact(con);
}

public void delete(View v)
{
    EditText e_id = (EditText)findViewById(R.id._id);
    EditText e_region = (EditText)findViewById(R.id._region);
    EditText e_object = (EditText)findViewById(R.id._object);
    EditText e_bus_num = (EditText)findViewById(R.id._num_bus);

```

```

        EditText e_type= (EditText)findViewById(R.id._type);
Contact con= new Contact();
con.setID(Integer.parseInt(e_id.getText().toString()));
con.setRegion(e_region.getText().toString());
con.setObject(e_object.getText().toString());
con.setBusNum(e_bus_num.getText().toString());
con.setType(e_type.getText().toString());
db.deleteContact(con);
}
public void byId(View v)
{
    EditText e_id = (EditText)findViewById(R.id._id);
    EditText e_region = (EditText)findViewById(R.id._region);
    EditText e_object = (EditText)findViewById(R.id._object);
    EditText e_bus_num = (EditText)findViewById(R.id._num_bus);
    EditText e_type= (EditText)findViewById(R.id._type);

    int id = Integer.parseInt(e_id.getText().toString());

    Contact cn = db.getBouteById(id);

    String ild, iRegion, iObject, iBusNum, iType;

        ild = "Id: " + cn.getID();
        iRegion ="Region: "+ cn.getRegion();
        iObject = "Object: "+cn.getObject();
        iBusNum = "Bus: "+cn.getBusNum();
        iType = "Type: "+cn.getType();
        e_id.setText(ild);
        e_region.setText(iRegion);
        e_object.setText(iObject);
        e_bus_num.setText(iBusNum);
        e_type.setText(iType);
    }
}

```