Mobile Application development

Practical list

1. STUDY OF ANDROID.
2. IMPLEMENT HELLO WORD USING ANDROID.
3. CREATE AN ANDROID ACTIVITY TO NAVIGATE FROM ONE ACTIVITY TO ANOTHER ACTIVITY
4. CREATE AN ANDTOID APPLICATION TO SHOW THE DIFFERENCE BETWEEN WRAP CONTENT AND FILL PARENT TO CONTROL THE COMPONENT'S WIDTH AND HEIGHT.
5. CREATE A LOGIN ACTIVITY. IT ASK "USERNAME" AND "PASSWORD" FROM USER. IF USERNAME AND PASSWORD ARE VALID, IT DISPLAYS WELCOME MESSAGE USING NEW ACTIVITY.
6. CREATE AN ANDROID APP WHICH DISPLAY A FORM TO GET FOLLOWING INFORMATIONS FROM USER. USERNAME, PASSWORD, EMAIL, PHONENO, COUNTRY, STATE, GENDER, BIRTHDATE. FORM SHOULD BE FOLLOWD BY BUTTON WITH LABAL SUBMIT. WHEN USER CLICK THE BUTTON, DISPLAY THE INFORMATION ENTERED BY USER.
7. DEVELOP CALCULATOR ANDROID APPLICATION.
8. SEND SMS USING ANDROID APP.
9. CREATE CALLING APP IN ANDROID.
10. Prepare a wireless ad hoc network and show its working.

    An ad hoc network is a local area network you can build spontaneously in a pinch. It allows computers and devices to communicate directly with each other within a small radius. The ability to create an ad hoc network has been possible since XP and is a way to quickly share data and an Internet connection to other wireless devices. For this tutorial you'll need the host computer to be hardwired and have wireless capabilities such as a Netbook or Laptop.

PRACTICAL-1

AIM: STUDY OF ANDROID.

Android is an open source and Linux-based **Operating System** for mobile devices such as smartphones and tablet computers. Android was developed by the Open Handset Alliance, led by Google, and other companies.

Android offers a unified approach to application development for mobile devices which means developers need only develop for Android, and their applications should be able to run on different devices powered by Android.

The first beta version of the Android Software Development Kit (SDK) was released by Google in 2007 where as the first commercial version, Android 1.0, was released in September 2008.

On June 27, 2012, at the Google I/O conference, Google announced the next Android version, 4.1 **Jelly Bean**. Jelly Bean is an incremental update, with the primary aim of improving the user interface, both in terms of functionality and performance.

The source code for Android is available under free and open source software licenses. Google publishes most of the code under the Apache License version 2.0 and the rest, Linux kernel changes, under the GNU General Public License version 2.

Features of Android:

Android is a powerful operating system competing with Apple 4GS and supports great features. Few of them are listed below:

Feature	Description
Beautiful UI	Android OS basic screen provides a beautiful and intuitive user interface.
Connectivity	GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, NFC and WiMAX.
Storage	SQLite, a lightweight relational database, is used for data storage purposes.
Media support	H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC, AAC 5.1, MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF, and BMP
Messaging	SMS and MMS
Web browser	Based on the open-source WebKit layout engine, coupled with Chrome's V8 JavaScript engine supporting HTML5 and CSS3.
Multi-touch	Android has native support for multi-touch which was initially made available in handsets such as the HTC Hero.
Multi-tasking	User can jump from one task to another and same time various application can run simultaneously.
Resizable widgets	Widgets are resizable, so users can expand them to show more content or shrink them to save space
Multi-Language	Supports single direction and bi-directional text.
GCM	Google Cloud Messaging (GCM) is a service that lets developers send short message data to their users on Android devices, without needing a proprietary sync solution.
Wi-Fi Direct	A technology that lets apps discover and pair directly, over a high-bandwidth peer-to-peer connection.
Android Beam	A popular NFC-based technology that lets users instantly share, just by touching two NFC-enabled phones together.

Android Applications:

Android applications are usually developed in the Java language using the Android Software Development Kit.

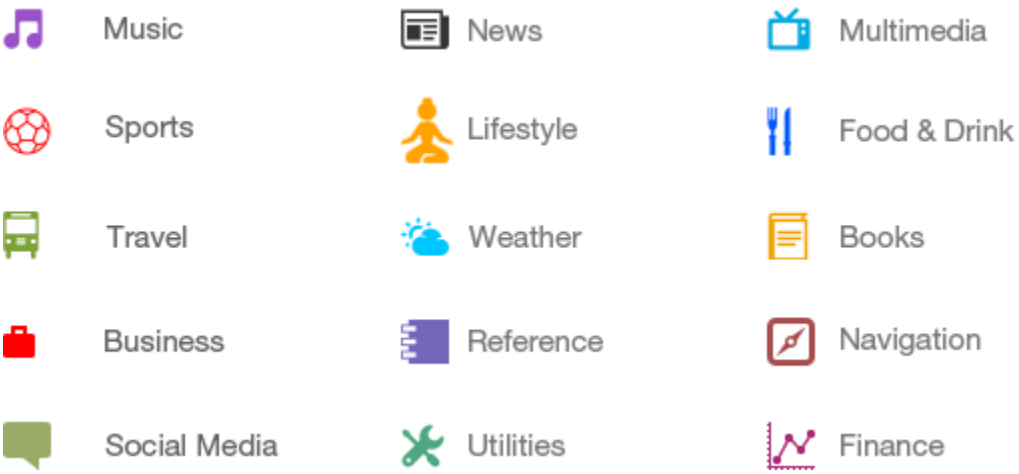
Once developed, Android applications can be packaged easily and sold out either through a store such as **Google Play**,**SlideME**,**Opera Mobile Store**,**Mobango**,**F-droid** and the **Amazon Appstore**.

Android powers hundreds of millions of mobile devices in more than 190 countries around the world. It's the largest installed base of any mobile platform and growing fast. Every day more than 1 million new Android devices are activated worldwide.

This tutorial has been written with an aim to teach you how to develop and package Android application. We will start from environment setup for Android application programming and then drill down to look into various aspects of Android applications.

Categories of Android applications:

There are many android applications in the market. The top categories are:



History of Android:

The code names of android ranges from A to L currently, such as Aestro, Blender, Cupcake, Donut, Eclair, Froyo, Gingerbread, Honeycomb, Ice Cream Sandwitch, Jelly Bean, KitKat and Lollipop. Let's understand the android history in a sequence.

What is API level?

API Level is an integer value that uniquely identifies the framework API revision offered by a version of the Android platform.

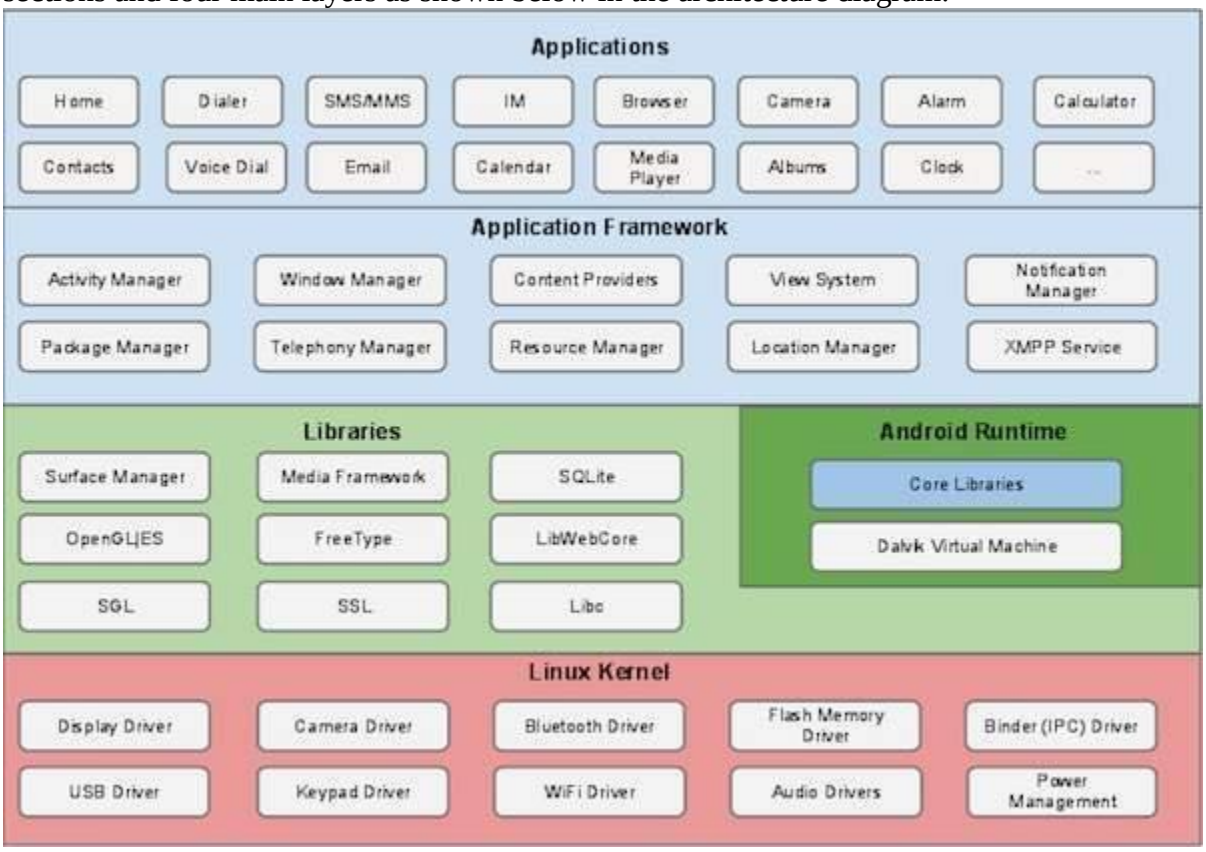
Platform Version	API Level	VERSION_CODE	
Android 5.1	22	LOLLIPOP_MR1	
Android 5.0	21	LOLLIPOP	
Android 4.4W	20	KITKAT_WATCH	KitKat for Wearables Only
Android 4.4	19	KITKAT	
Android 4.3	18	JELLY_BEAN_MR2	
Android 4.2, 4.2.2	17	JELLY_BEAN_MR1	
Android 4.1, 4.1.1	16	JELLY_BEAN	
Android 4.0.3, 4.0.4	15	ICE_CREAM_SANDWICH_MR1	
Android 4.0, 4.0.1, 4.0.2	14	ICE_CREAM_SANDWICH	
Android 3.2	13	HONEYCOMB_MR2	
Android 3.1.x	12	HONEYCOMB_MR1	
Android 3.0.x	11	HONEYCOMB	
Android 2.3.4 Android 2.3.3	10	GINGERBREAD_MR1	
Android 2.3.2 Android 2.3.1 Android 2.3	9	GINGERBREAD	
Android 2.2.x	8	FROYO	
Android 2.1.x	7	ECLAIR_MR1	
Android 2.0.1	6	ECLAIR_0_1	
Android 2.0	5	ÉCLAIR	
Android 1.6	4	DONUT	
Android 1.5	3	CUPCAKE	
Android 1.1	2	BASE_1_1	
Android 1.0	1	BASE	

Android IDEs:

There are so many sophisticated Technologies are available to develop android applications, the familiar technologies, which are predominantly using tools as follows

- Android Studio
- Eclipse IDE

Android operating system is a stack of software components which is roughly divided into five sections and four main layers as shown below in the architecture diagram.



Linux kernel:

At the bottom of the layers is Linux - Linux 3.6 with approximately 115 patches. This provides a level of abstraction between the device hardware and it contains all the essential hardware drivers like camera, keypad, display etc. Also, the kernel handles all the things that Linux is really good at such as networking and a vast array of device drivers, which take the pain out of interfacing to peripheral hardware.

Libraries:

On top of Linux kernel there is a set of libraries including open-source Web browser engine WebKit, well known library libc, SQLite database which is a useful repository for storage and

sharing of application data, libraries to play and record audio and video, SSL libraries responsible for Internet security etc.

Android Libraries:

This category encompasses those Java-based libraries that are specific to Android development. Examples of libraries in this category include the application framework libraries in addition to those that facilitate user interface building, graphics drawing and database access. A summary of some key core Android libraries available to the Android developer is as follows −

- **android.app** − Provides access to the application model and is the cornerstone of all Android applications.
- **android.content** − Facilitates content access, publishing and messaging between applications and application components.
- **android.database** − Used to access data published by content providers and includes SQLite database management classes.
- **android.opengl** − A Java interface to the OpenGL ES 3D graphics rendering API.
- **android.os** − Provides applications with access to standard operating system services including messages, system services and inter-process communication.
- **android.text** − Used to render and manipulate text on a device display.
- **android.view** − The fundamental building blocks of application user interfaces.
- **android.widget** − A rich collection of pre-built user interface components such as buttons, labels, list views, layout managers, radio buttons etc.
- **android.webkit** − A set of classes intended to allow web-browsing capabilities to be built into applications.

Having covered the Java-based core libraries in the Android runtime, it is now time to turn our attention to the C/C++ based libraries contained in this layer of the Android software stack.

Android Runtime:

This is the third section of the architecture and available on the second layer from the bottom. This section provides a key component called **Dalvik Virtual Machine** which is a kind of Java Virtual Machine specially designed and optimized for Android.

The Dalvik VM makes use of Linux core features like memory management and multi-threading, which is intrinsic in the Java language. The Dalvik VM enables every Android application to run in its own process, with its own instance of the Dalvik virtual machine.

The Android runtime also provides a set of core libraries which enable Android application developers to write Android applications using standard Java programming language.

Application Framework:

The Application Framework layer provides many higher-level services to applications in the form of Java classes. Application developers are allowed to make use of these services in their applications.

The Android framework includes the following key services −

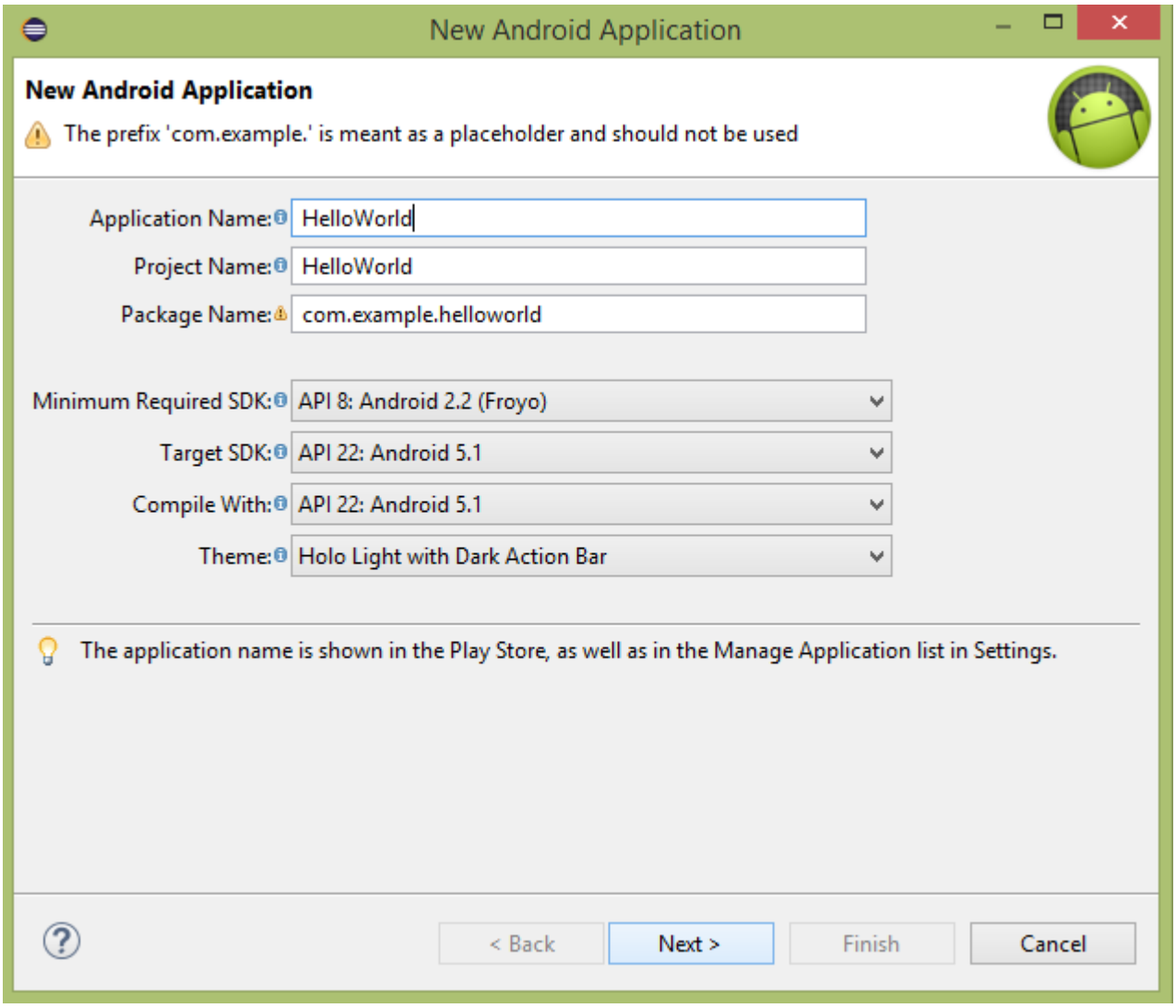
- **Activity Manager** − Controls all aspects of the application lifecycle and activity stack.
- **Content Providers** − Allows applications to publish and share data with other applications.
- **Resource Manager** − Provides access to non-code embedded resources such as strings, color settings and user interface layouts.
- **Notifications Manager** − Allows applications to display alerts and notifications to the user.
- **View System** − An extensible set of views used to create application user interfaces.

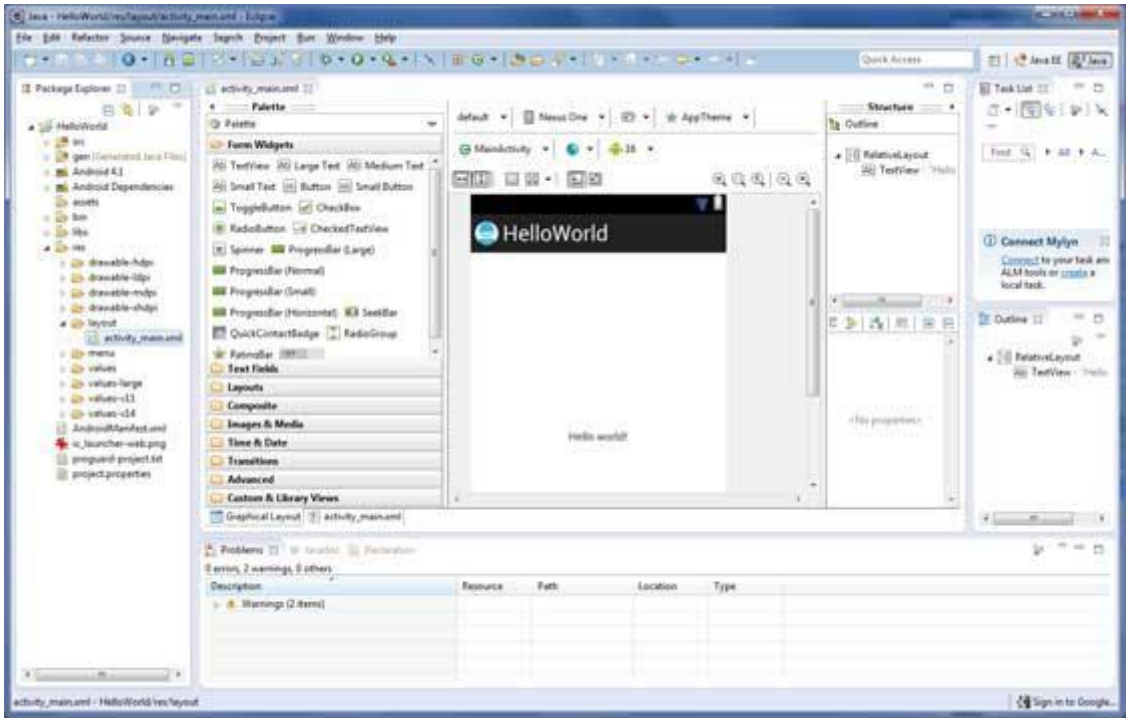
Let us start actual programming with Android Framework. Before you start writing your first example using Android SDK, you have to make sure that you have set-up your Android development environment properly as explained in Android - Environment Set-up tutorial. I also assume that you have a little bit working knowledge with Eclipse IDE.

So let us proceed to write a simple Android Application which will print "Hello World!".

Create Android Application:

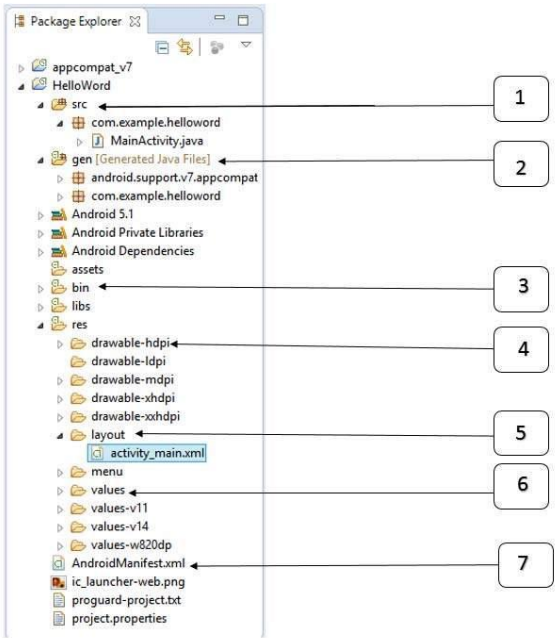
The first step is to create a simple Android Application using Eclipse IDE. Follow the option **File -> New -> Project** and finally select **Android New Application** wizard from the wizard list. Now name your application as **HelloWorld** using the wizard window as follows:


New Android Application
New Android Application
The prefix 'com.example.' is meant as a placeholder and should not be used
Application Name: HelloWorld
Project Name: HelloWorld
Package Name: com.example.helloworld
Minimum Required SDK: API 8: Android 2.2 (Froyo)
Target SDK: API 22: Android 5.1
Compile With: API 22: Android 5.1
Theme: Holo Light with Dark Action Bar
The application name is shown in the Play Store, as well as in the Manage Application list in Settings.
< Back
Next >
Finish
Cancel



Anatomy of Android Application:

Before you run your app, you should be aware of a few directories and files in the Android project −



S. N.	Folder, File & Description
1	**src** This contains the **.java** source files for your project. By default, it includes an MainActivity.java source file having an activity class that runs when your app is launched using the app icon.
2	**gen** This contains the **.R** file, a compiler-generated file that references all the resources found in your project. You should not modify this file.
3	**bin** This folder contains the Android package files **.apk** built by the ADT during the build process and everything else needed to run an Android application.
4	**res/drawable-hdpi** This is a directory for drawable objects that are designed for high-density screens.
5	**res/layout** This is a directory for files that define your app's user interface.
6	**res/values** This is a directory for other various XML files that contain a collection of resources, such as strings and colours definitions.
7	**AndroidManifest.xml** This is the manifest file which describes the fundamental characteristics of the app and defines each of its components.

PRACTICAL- 2

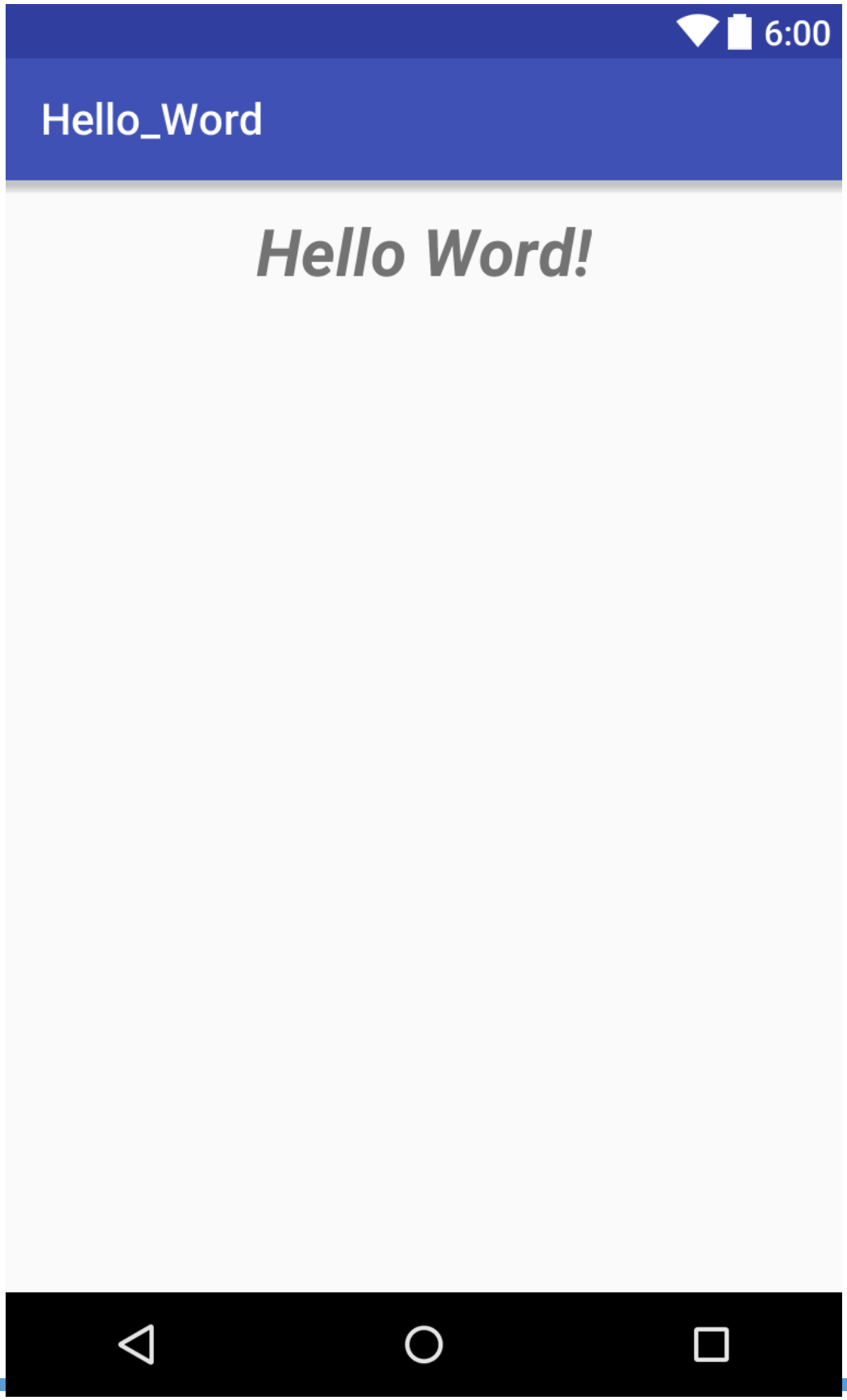
AIM: IMPLEMENT HELLO WORD USING ANDROID.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.vaibhav.hello_word.MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello Word!"
        android:layout_gravity="center"
        android:textSize="30dp"
        android:textStyle="bold|italic"
        />
</LinearLayout>
```

MainActivity.java

```
package com.example.vaibhav.hello_word;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

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


6:00
Hello_Word
Hello Word!

PRACTICAL 3

AIM: CREATE AN ANDROID ACTIVITY TO NAVIGATE FROM ONE ACTIVITY TO ANOTHER ACTIVITY.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:paddingBottom="@dimen/activity_vertical_margin"
   android:paddingLeft="@dimen/activity_horizontal_margin"
   android:paddingRight="@dimen/activity_horizontal_margin"
   android:paddingTop="@dimen/activity_vertical_margin"
   tools:context="com.example.vaibhav.intent.MainActivity">

   <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="This is 1st activity"
      android:textSize="40dp"
      android:textStyle="bold|italic"/>
</RelativeLayout>
```

ActivityMain.java

```
package com.example.vaibhav.intent;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity implements View.OnClickListener {

   @Override
   protected void onCreate(Bundle savedInstanceState) {

      TextView  firstactivity;
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
```

```
        firstactivity=(TextView)findViewById(R.id.first);
        firstactivity.setOnClickListener(this);

    }

    @Override
    public void onClick(View v) {

        Intent intent = new Intent(MainActivity.this,SecondActivity.class);
        MainActivity.this.startActivity(intent);
        MainActivity.this.finish();

    }
}
```

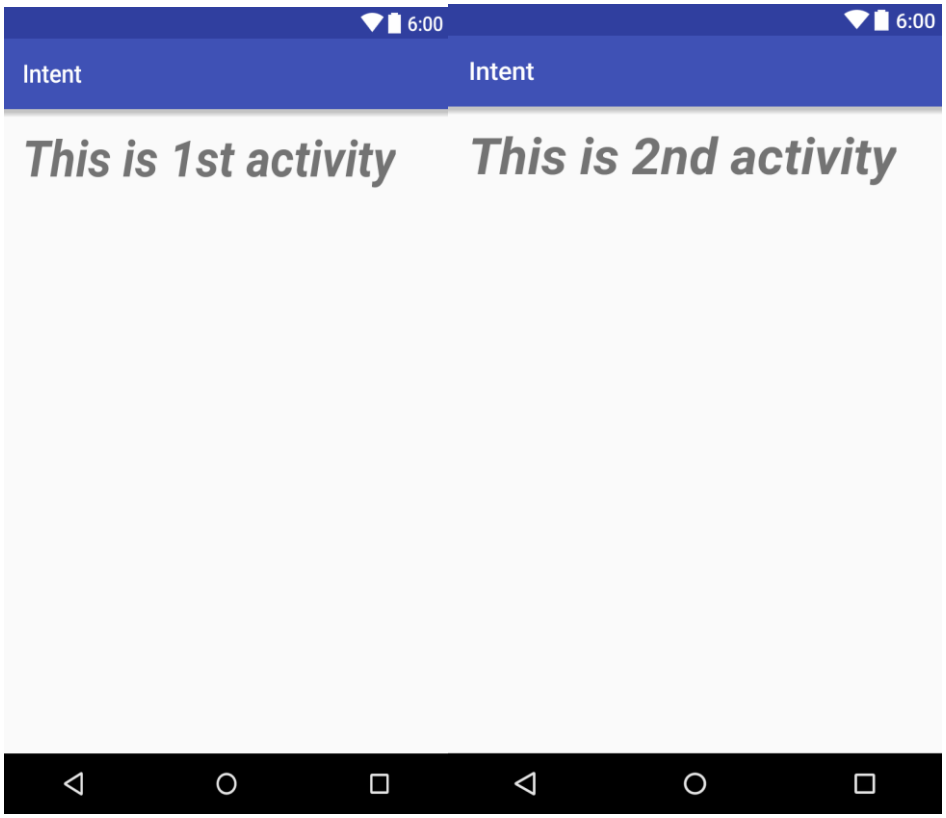
activity_second.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.vaibhav.intent.SecondActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="This is 2nd activity"
        android:textSize="40dp"
        android:textStyle="bold|italic"/>


</RelativeLayout>
```

SecondActivity.java

```
package com.example.vaibhav.intent;

import android.support.v7.app.AppCompatActivity;
```

```
import android.os.Bundle;

public class SecondActivity extends AppCompatActivity {

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



]

PRACTICAL 4

AIM: CREATE AN ANDTOID APPLICATION TO SHOW THE DIFFERENCE BETWEEN WRAP CONTENT AND FILL PARENT TO CONTROL THE COMPONENT'S WIDTH AND HEIGHT.

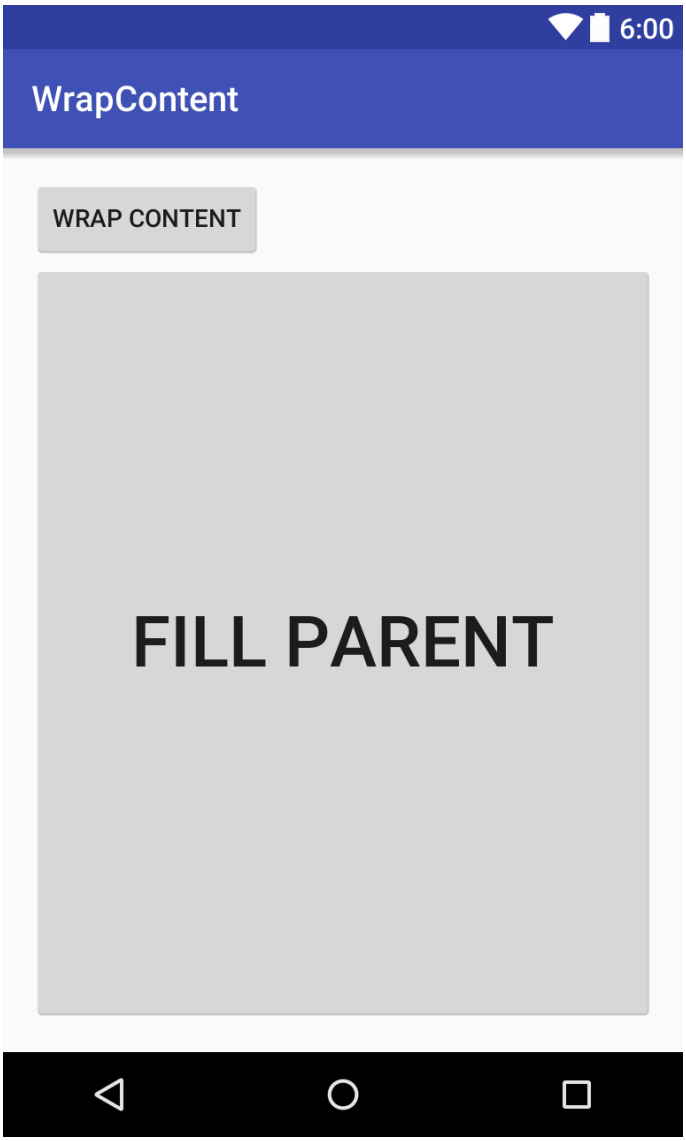
activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:orientation="vertical"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.vaibhav.wrapcontent.MainActivity">

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Wrap Content"
        />


    <Button
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:text="Fill parent"
        android:textSize="40dp"/>
</LinearLayout>
```

MainActivity.java

```
package com.example.vaibhav.wrapcontent;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {
```

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



PRACTICAL 5

AIM: CREATE A LOGIN ACTIVITY. IT ASK “USERNAME” AND “PASSWORD” FROM USER. IF USERNAME AND PASSWORD ARE VALID, IT DISPLAYS WELCOME MESSAGE USING NEW ACTIVITY.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.vaibhav.login.MainActivity">

    <EditText
        android:id="@+id/nameEt"
        android:layout_marginTop="10dp"
        android:gravity="center"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Your UserName"/>

    <EditText
        android:id="@+id/passwordEt"
        android:layout_marginTop="2dp"
        android:gravity="center"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
        android:hint="Enter password"/>

   <Button
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:text="LogIn"
```

```
    android:textSize="20dp"
    android:id="@+id/login"
    />
</LinearLayout>
```

MainActivity.java

```
package com.example.vaibhav.login;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity implements View.OnClickListener{

   private EditText nameET, numberEt, passwordEt, confirmPass;
   private Button Login;

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

      nameET = (EditText) findViewById(R.id.nameEt);
      passwordEt = (EditText) findViewById(R.id.passwordEt);
      Login = (Button) findViewById(R.id.login);


      Login.setOnClickListener(this);
   }

   public void onClick(View v) {

      switch (v.getId()) {

         case R.id.login:
            loginUser();
            break;
      }

   }
```

```
    private void loginUser() {

        String name = nameET.getText().toString();
        String password = passwordEt.getText().toString();


        if (name.equals("vaibhav") && password.equals("1234")) {
            Intent intent = new Intent(MainActivity.this, SecondActivity.class);
            Bundle bundle = new Bundle();

            bundle.putString("name_key", name);
            intent.putExtras(bundle);

            startActivity(intent);
            finish();
        }

        else {
            Toast.makeText(MainActivity.this, "invalid user", Toast.LENGTH_LONG).show();
        }

    }
}
```

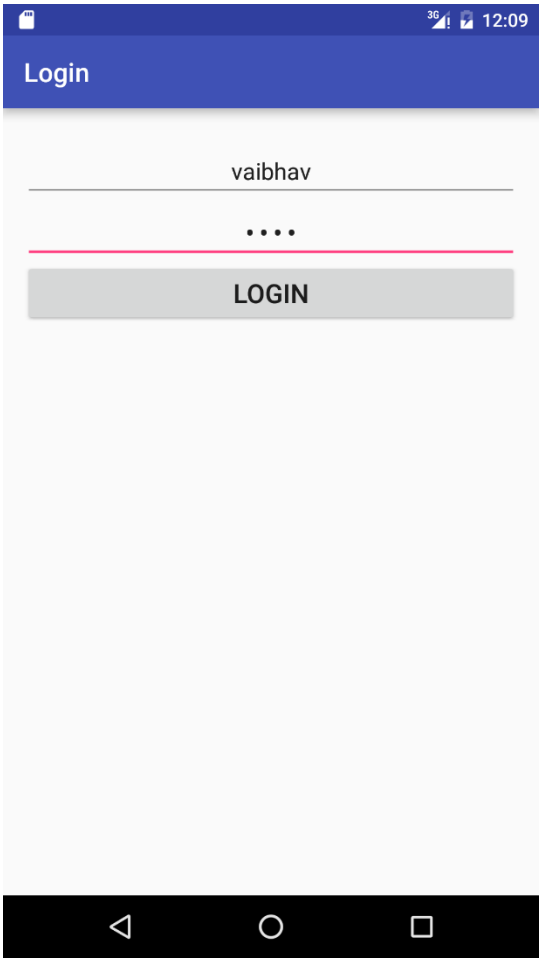
activity_second.xml

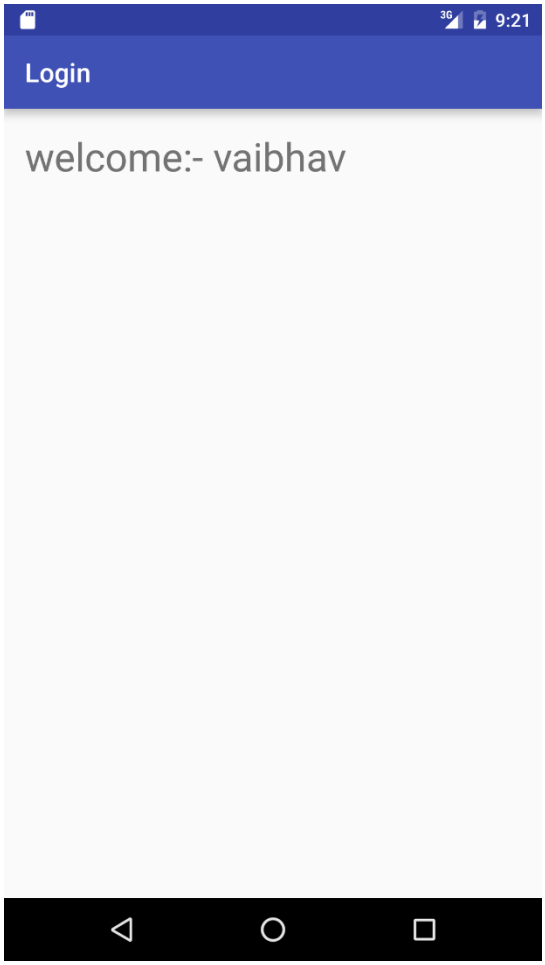
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.vaibhav.login.SecondActivity">


    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="welcome user"
        android:textSize="30dp"
        android:gravity="center"
```

```
        android:id="@+id/second_name"
        />

</RelativeLayout>
```

SecondActivity.java

```
package com.example.vaibhav.login;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.TextView;

public class SecondActivity extends AppCompatActivity {

    TextView nametv;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_second);

        nametv =(TextView) findViewById(R.id.second_name);

        Bundle bundle = getIntent().getExtras();
        String name = bundle.getString("name_key");

        nametv.setText("welcome:- "+name);



    }

}
```


12:09
Login
vaibhav
••••
LOGIN


9:21
Login
welcome:- vaibhav

PRACTICAL 6

AIM: CREATE AN ANDROID APP WHICH DISPLAY A FORM TO GET FOLLOWING INFORMATIONS FROM USER. USERNAME, PASSWORD, EMAIL, PHONENO, COUNTRY, STATE, GENDER, BIRTHDATE. FORM SHOULD BE FOLLOWD BY BUTTON WITH LABAL SUBMIT. WHEN USER CLICK THE BUTTON, DISPLAY THE INFORMATION ENTERED BY USER.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.vaibhav.form.MainActivity">

    <EditText
        android:id="@+id/nameEt"
        android:layout_marginTop="10dp"
        android:gravity="center"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Your UserName"/>

    <EditText
        android:id="@+id/numberEt"
        android:layout_marginTop="0dp"
        android:gravity="center"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="phone"
        android:hint="Enter Your Number"/>
    <EditText
        android:id="@+id/emailET"
        android:layout_marginTop="0dp"
        android:gravity="center"
        android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
        android:inputType="textEmailAddress"
        android:hint="Enter Your Email"/>
    <EditText
        android:id="@+id/addressET"
        android:layout_marginTop="0dp"
        android:gravity="center"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textMultiLine"
        android:hint="Enter Your Address"/>

    <EditText
        android:id="@+id/countryET"
        android:layout_marginTop="3dp"
        android:gravity="center"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Your Country"/>

    <EditText
        android:id="@+id/stateET"
        android:layout_marginTop="0dp"
        android:gravity="center"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Your State"/>

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:inputType="date"
        android:hint="Birth Date"
        android:id="@+id/bdate"/>
<RadioGroup

    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
    <RadioButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:hint="MALE"
        android:id="@+id/malebtn"
        android:onClick="onRadioButtonClicked"
```

```
      />
   <RadioButton
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:hint="FEMALE"
      android:id="@+id/femalebtn"
      android:onClick="onRadioButtonClicked"
      />
</RadioGroup>

   <Button
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:text="SUBMIT"
      android:id="@+id/submit"
      android:layout_gravity="center"/>

</LinearLayout>
```

MainActivity.java

```
package com.example.vaibhav.form;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity implements View.OnClickListener {

    EditText nameET,numberEt,emailET,addressET,countryET,stateET,birthdateET;
   Button submitBtn;
   RadioButton male,female;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);

      nameET =(EditText) findViewById(R.id.nameEt);
      numberEt =(EditText) findViewById(R.id.numberEt);

      emailET =(EditText) findViewById(R.id.emailET);
```

```
    addressET =(EditText) findViewById(R.id.addressET);
    countryET =(EditText) findViewById(R.id.countryET);
    stateET =(EditText) findViewById(R.id.stateET);
    birthdateET =(EditText) findViewById(R.id.bdate);
    submitBtn =(Button) findViewById(R.id.submit);
    male =(RadioButton) findViewById(R.id.malebtn);
    female =(RadioButton) findViewById(R.id.femalebtn);

    submitBtn.setOnClickListener(this);


}

    public void onRadioButtonClicked(View view) {
        // Is the button now checked?
        boolean checked = ((RadioButton) view).isChecked();

        // Check which radio button was clicked
        switch(view.getId()) {
            case R.id.malebtn:
                if (checked)

                    break;
            case R.id.femalebtn:
                if (checked)

                    break;
        }
    }

@Override
public void onClick(View v) {

    String name=nameET.getText().toString();
    String number=numberEt.getText().toString();
    String address=addressET.getText().toString();
    String email=emailET.getText().toString();
    String country=countryET.getText().toString();
    String state=stateET.getText().toString();
    String bdate=birthdateET.getText().toString();

    Intent intent = new Intent(MainActivity.this, SecondActivity.class);
    Bundle bundle = new Bundle();
    bundle.putString("name_key",name);
    bundle.putString("no_key",number);
    bundle.putString("address_key",address);
```

```
        bundle.putString("email_key",email);
        bundle.putString("country_key",country);
        bundle.putString("state_key",state);
        bundle.putString("bdate_key",bdate);

        intent.putExtras(bundle);
        startActivity(intent);
        finish();

    }

}
```

activity_second.xml

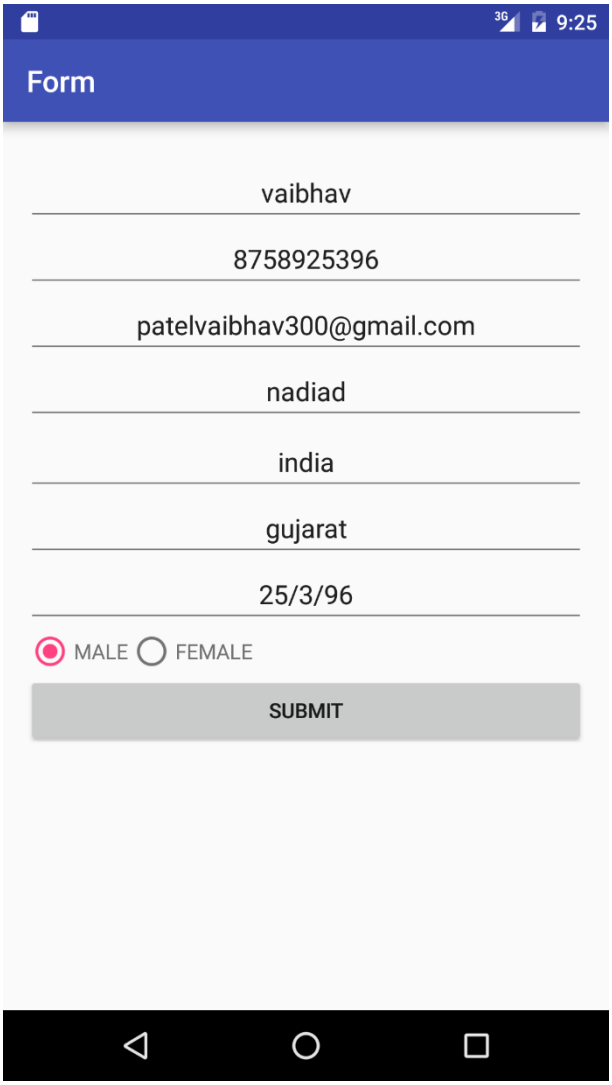
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.vaibhav.form.SecondActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="NAME"
        android:textSize="30dp"
        android:gravity="center"
        android:id="@+id/name"
        />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="NUMBER"
        android:textSize="30dp"
        android:gravity="center"
        android:id="@+id/number"
        />

    <TextView
```

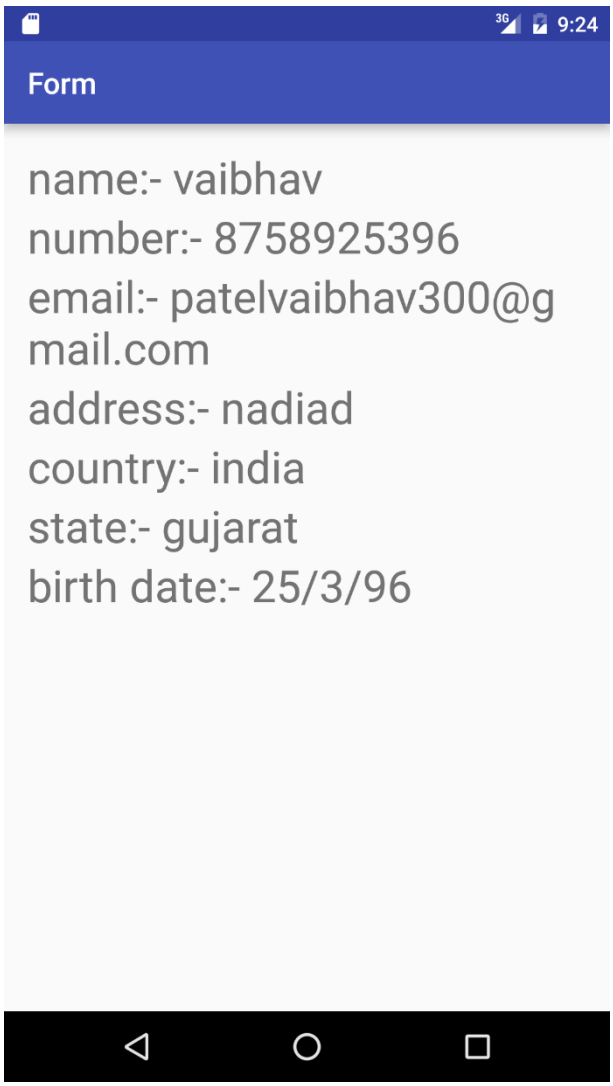
```
        android:id="@+id/email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="EMAIL"
        android:textSize="30dp"
        />
    <TextView
        android:id="@+id/address"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="address"
        android:textSize="30dp"/>

    <TextView
        android:id="@+id/country"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="country"
        android:textSize="30dp"/>

    <TextView
        android:id="@+id/state"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="STATE"
        android:textSize="30dp"/>
    <TextView
        android:id="@+id/bdate"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="BDATE"
        android:textSize="30dp"/>

</LinearLayout>
```

SeconActivtiy.java

```
package com.example.vaibhav.form;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.EditText;
import android.widget.TextView;

public class SecondActivity extends AppCompatActivity {
```

```
    TextView nametv,numbertv,addresstv,emailtv,countrytv,statetv,birthdatetv;

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


        nametv =(TextView) findViewById(R.id.name);
        numbertv =(TextView) findViewById(R.id.number);
        addresstv =(TextView) findViewById(R.id.address);
        emailtv =(TextView) findViewById(R.id.email);
        countrytv =(TextView) findViewById(R.id.country);
        statetv =(TextView) findViewById(R.id.state);
        birthdatetv =(TextView) findViewById(R.id.bdate);

        Bundle bundle = getIntent().getExtras();

        String name = bundle.getString("name_key");
        String number = bundle.getString("no_key");
        String email = bundle.getString("email_key");
        String address = bundle.getString("address_key");
        String country = bundle.getString("country_key");
        String state = bundle.getString("state_key");
        String birthdate = bundle.getString("bdate_key");


        nametv.setText("name:- "+name);
        numbertv.setText("number:- "+number);
        emailtv.setText("email:- "+email);
        addresstv.setText("address:- "+address);
        countrytv.setText("country:- "+country);
        statetv.setText("state:- "+state);
        birthdatetv.setText("birth date:- "+birthdate);


    }
}
```


9:25
Form
vaibhav
8758925396
patelvaibhav300@gmail.com
nadiad
india
gujarat
25/3/96
MALE
FEMALE
SUBMIT


9:24
Form
name:- vaibhav
number:- 8758925396
email:- patelvaibhav300@gmail.com
address:- nadiad
country:- india
state:- gujarat
birth date:- 25/3/96

PRACTICAL 7

AIM: DEVELOP CALCULATOR ANDROID APPLICATION.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:tools="http://schemas.android.com/tools"
   android:orientation="vertical"
   android:layout_width="fill_parent"
   android:layout_height="fill_parent"
   tools:context="com.example.vaibhav.smiplecalc.MainActivity">
   <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:id="@+id/linearLayout1"
      android:layout_marginLeft="10pt"
      android:layout_marginRight="10pt"
      android:layout_marginTop="3pt">
      <EditText
         android:layout_weight="1"
         android:layout_height="wrap_content"
         android:layout_marginRight="5pt"
         android:id="@+id/etNum1"
         android:layout_width="match_parent"
         android:inputType="numberDecimal">
      </EditText>
      <EditText
         android:layout_height="wrap_content"
         android:layout_weight="1"
         android:layout_marginLeft="5pt"
         android:id="@+id/etNum2"
         android:layout_width="match_parent"
         android:inputType="numberDecimal">
      </EditText>
   </LinearLayout>
   <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:id="@+id/linearLayout2"
      android:layout_marginTop="3pt"
      android:layout_marginLeft="5pt"
      android:layout_marginRight="5pt">
      <Button
```

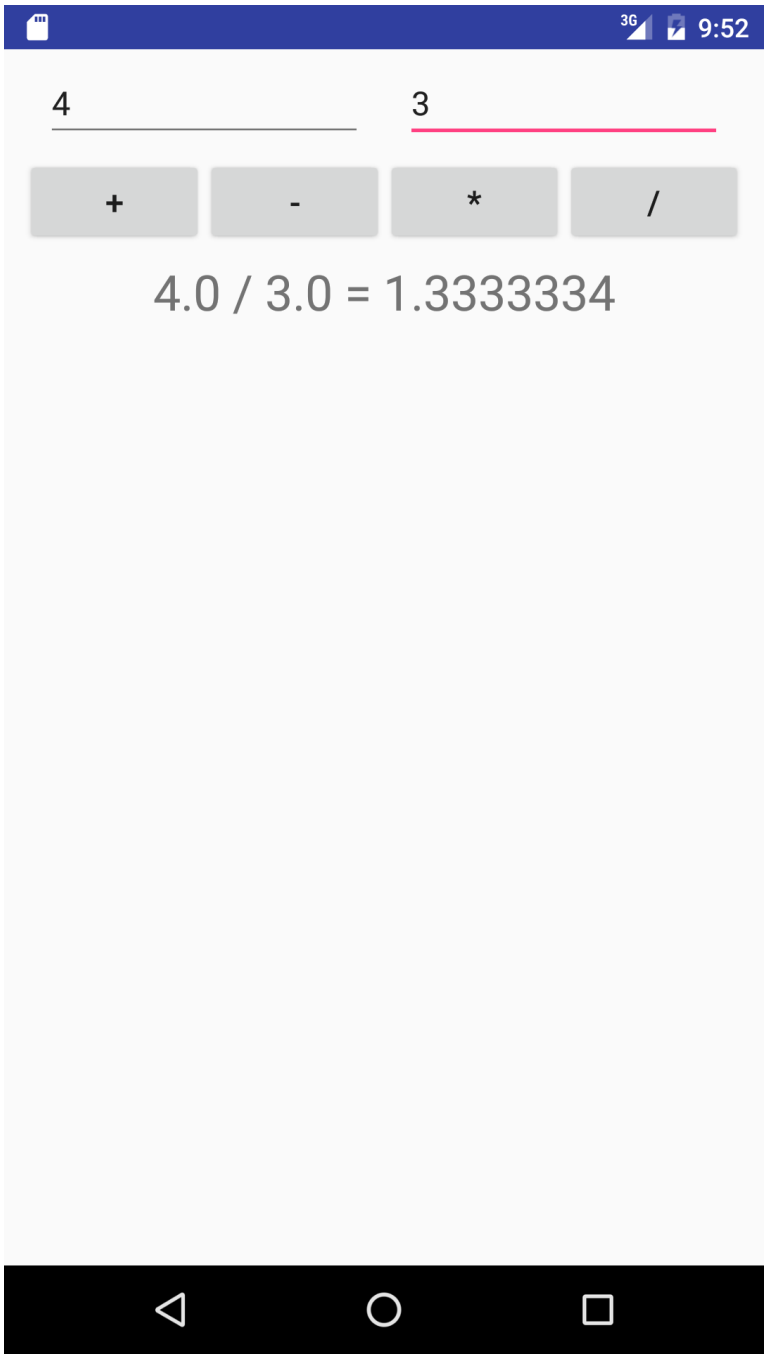
```
            android:layout_height="wrap_content"
            android:layout_width="match_parent"
            android:layout_weight="1"
            android:text="+"
            android:textSize="8pt"
            android:id="@+id/btnAdd">
        </Button>
        <Button
            android:layout_height="wrap_content"
            android:layout_width="match_parent"
            android:layout_weight="1"
            android:text="-"
            android:textSize="8pt"
            android:id="@+id/btnSub">
        </Button>
        <Button
            android:layout_height="wrap_content"
            android:layout_width="match_parent"
            android:layout_weight="1"
            android:text="*"
            android:textSize="8pt"
            android:id="@+id/btnMult">
        </Button>
        <Button
            android:layout_height="wrap_content"
            android:layout_width="match_parent"
            android:layout_weight="1"
            android:text="/"
            android:textSize="8pt"
            android:id="@+id/btnDiv">
        </Button>
    </LinearLayout>
    <TextView
        android:layout_height="wrap_content"
        android:layout_width="match_parent"
        android:layout_marginLeft="5pt"
        android:layout_marginRight="5pt"
        android:textSize="12pt"
        android:layout_marginTop="3pt"
        android:id="@+id/tvResult"
        android:gravity="center_horizontal">
    </TextView>

</LinearLayout>
```

MainActivity.java

```
package com.example.vaibhav.smiplecalc;

import android.app.Activity;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class MainActivity extends Activity implements View.OnClickListener {

    EditText etNum1;
    EditText etNum2;

    Button btnAdd;
    Button btnSub;
    Button btnMult;
    Button btnDiv;

    TextView tvResult;

    String oper = "";

    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        // find the elements
        etNum1 = (EditText) findViewById(R.id.etNum1);
        etNum2 = (EditText) findViewById(R.id.etNum2);

        btnAdd = (Button) findViewById(R.id.btnAdd);
        btnSub = (Button) findViewById(R.id.btnSub);
        btnMult = (Button) findViewById(R.id.btnMult);
        btnDiv = (Button) findViewById(R.id.btnDiv);

        tvResult = (TextView) findViewById(R.id.tvResult);
```

```
    // set a listener
    btnAdd.setOnClickListener(this);
    btnSub.setOnClickListener(this);
    btnMult.setOnClickListener(this);
    btnDiv.setOnClickListener(this);

}

@Override
public void onClick(View v) {
    // TODO Auto-generated method stub
    float num1 = 0;
    float num2 = 0;
    float result = 0;

    // check if the fields are empty
    if (TextUtils.isEmpty(etNum1.getText().toString())
            || TextUtils.isEmpty(etNum2.getText().toString())) {
        return;
    }

    // read EditText and fill variables with numbers
    num1 = Float.parseFloat(etNum1.getText().toString());
    num2 = Float.parseFloat(etNum2.getText().toString());

    // defines the button that has been clicked and performs the corresponding operation
    // write operation into oper, we will use it later for output
    switch (v.getId()) {
        case R.id.btnAdd:
            oper = "+";
            result = num1 + num2;
            break;
        case R.id.btnSub:
            oper = "-";
            result = num1 - num2;
            break;
        case R.id.btnMult:
            oper = "*";
            result = num1 * num2;
            break;
        case R.id.btnDiv:
            oper = "/";
            result = num1 / num2;
            break;
        default:
```

```
            break;
        }

        // form the output line
        tvResult.setText(num1 + " " + oper + " " + num2 + " = " + result);
    }
}
```


9:52
4
3
+
-
*
/
4.0 / 3.0 = 1.3333334

PRACTICAL 8

AIM: SEND SMS USING ANDROID APP.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.vaibhav.call.MainActivity">


    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:id="@+id/number"
        android:hint="Enter NUMBER:"
        android:gravity="center"
        android:textSize="30dp"
        />

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="40dp"
        android:id="@+id/call"
        android:text="MAKE CALL"
        android:gravity="center"
        android:textSize="30dp"
        />



</LinearLayout>
```

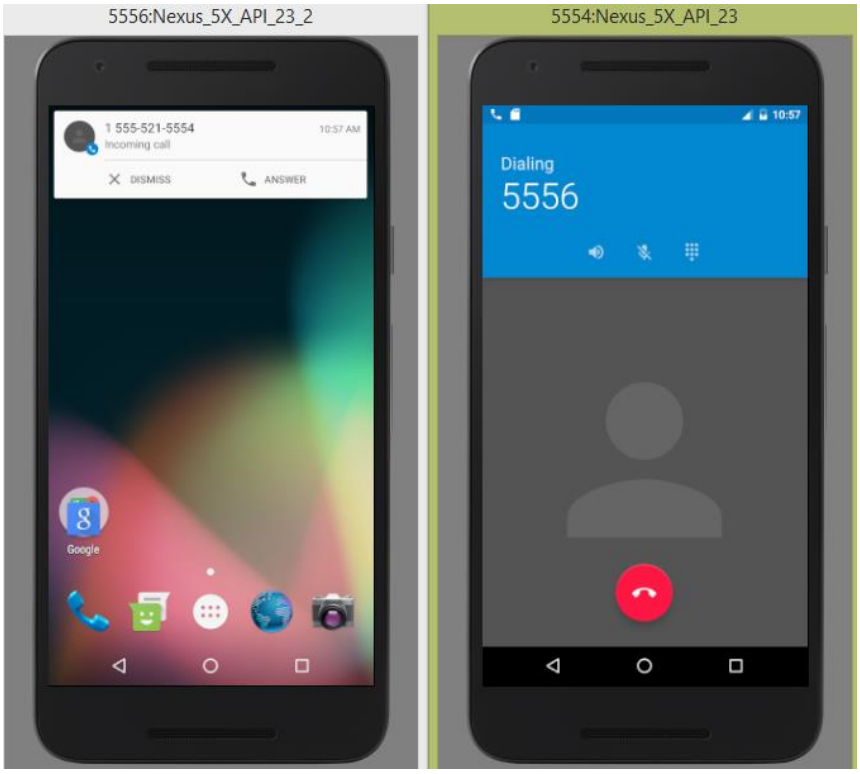
MainActivity.java

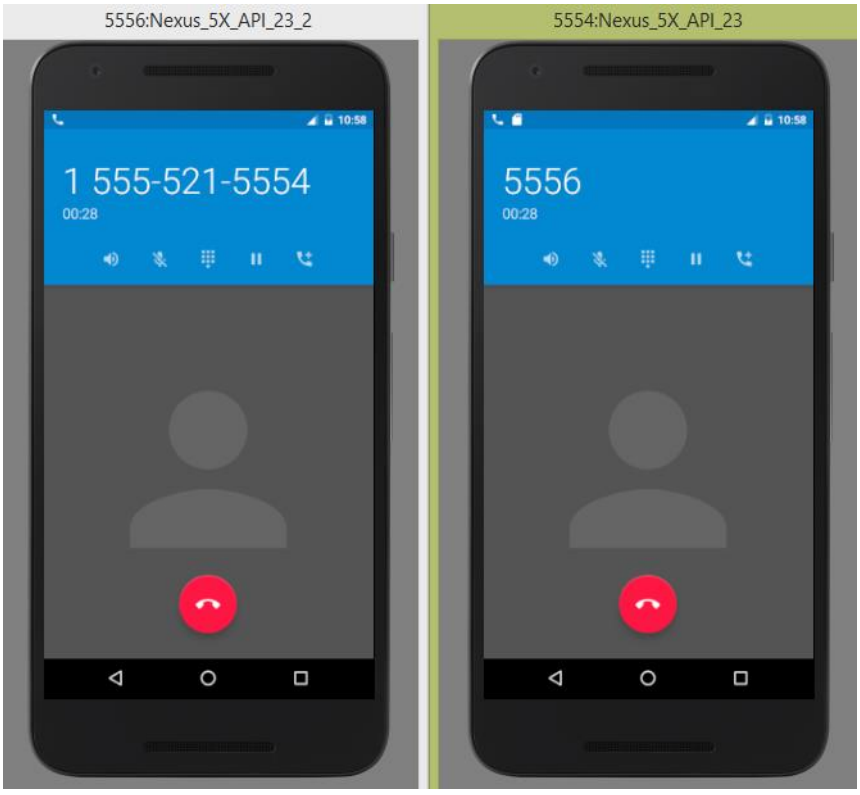
```
package com.example.vaibhav.call;

import android.content.Intent;
import android.net.Uri;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.telecom.Call;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {


    EditText num;
    Button call;

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


        num = (EditText)findViewById(R.id.number);
        call = (Button) findViewById(R.id.call);


        call.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                makeCall();
            }

        });

    }

    private void makeCall() {


        Intent intent = new Intent(Intent.ACTION_CALL);
        intent.setData(Uri.parse("tel:5556"));
```

```
        try {
            startActivity(intent);
        } catch (SecurityException e){

            Toast.makeText(MainActivity.this,"security permission
problem",Toast.LENGTH_LONG).show();


        }
    }
}
```

AndroidManifest.xml

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

    package="com.example.vaibhav.firstdemo">
<uses-permission android:name="android.permission.CALL_PHONE"/>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".SecondActivity"></activity>
    </application>
</manifest>
```


5556:Nexus_5X_API_23_2
1 555-521-5554
Incoming call
10:57 AM
DISMISS
ANSWER
Google
5554:Nexus_5X_API_23
10:57
Dialing
5556


5556:Nexus_5X_API_23_2
10:58
1 555-521-5554
00:28
5554:Nexus_5X_API_23
10:58
5556
00:28

PRACTICAL 9

AIM: CREATE CALLING APP IN ANDROID.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.vaibhav.sms.MainActivity">

 <EditText
     android:layout_width="match_parent"
     android:layout_height="wrap_content"
     android:id="@+id/number"
     android:hint="Enter Number:"
     android:gravity="center"
     android:textSize="30dp"
     />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:id="@+id/msg"
        android:hint="Enter Message:"
        android:gravity="center"
        android:textSize="30dp"
        />

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="40dp"
        android:id="@+id/sms"
        android:text="SEND SMS"
        android:gravity="center"
        android:textSize="30dp"
```

```
    />
</LinearLayout>
```

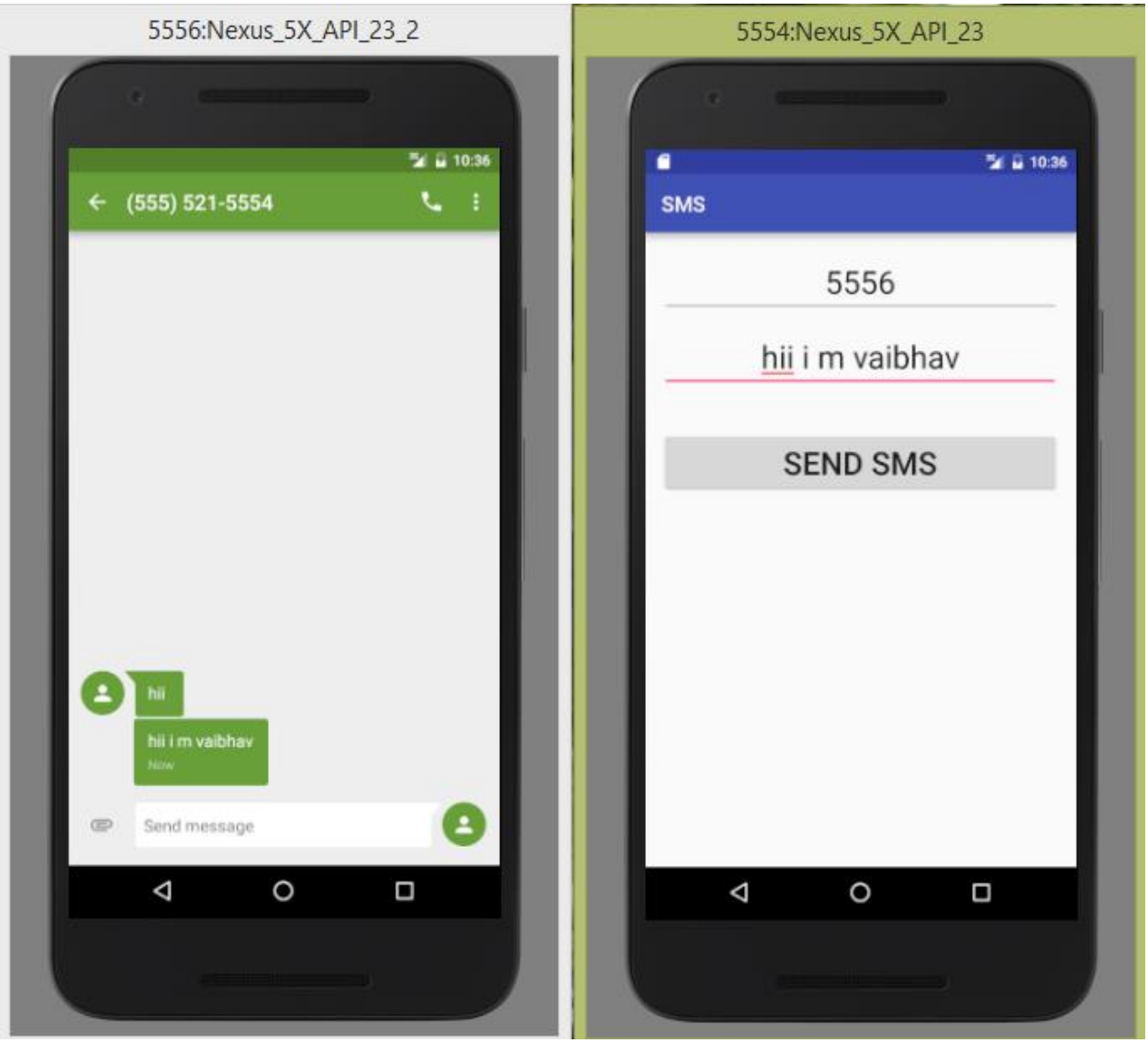
MainActivity.java

```
package com.example.vaibhav.sms;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.telephony.SmsManager;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

   EditText num,msg;
   Button send;

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

      num = (EditText)findViewById(R.id.number);
      msg = (EditText)findViewById(R.id.msg);
      send =(Button)findViewById(R.id.sms);

      send.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View v) {
         sendSms();
         }

      });
   }
   protected void sendSms() {

      String number=num.getText().toString();
      String message=msg.getText().toString();

      SmsManager manager = SmsManager.getDefault();
      manager.sendTextMessage(number,null,message,null,null);
```

```
        Toast.makeText(MainActivity.this,"send successfully",Toast.LENGTH_LONG).show();

    }

}
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.vaibhav.sms">
    <uses-permission android:name="android.permission.SEND_SMS"></uses-permission>


    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```


5556:Nexus_5X_API_23_2
10:36
(555) 521-5554
hii
hii i m vaibhav
Now
Send message
5554:Nexus_5X_API_23
10:36
SMS
5556
hii i m vaibhav
SEND SMS

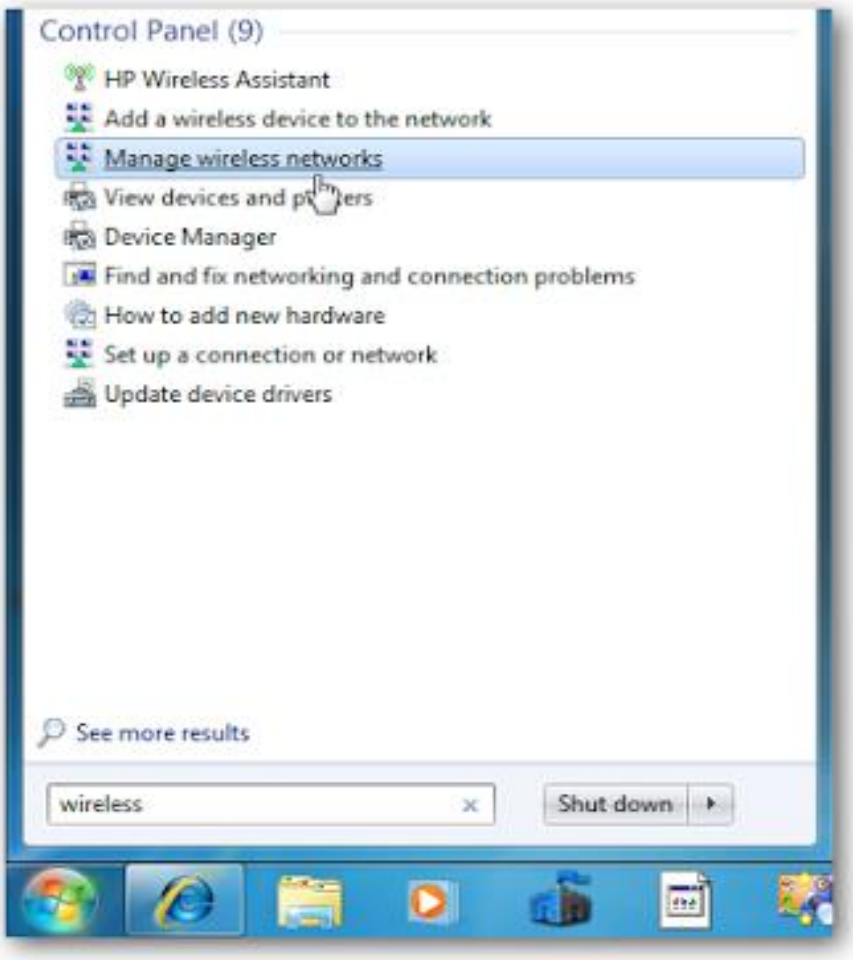
PRACTICAL 10

AIM: Prepare a wireless ad hoc network and show its working.

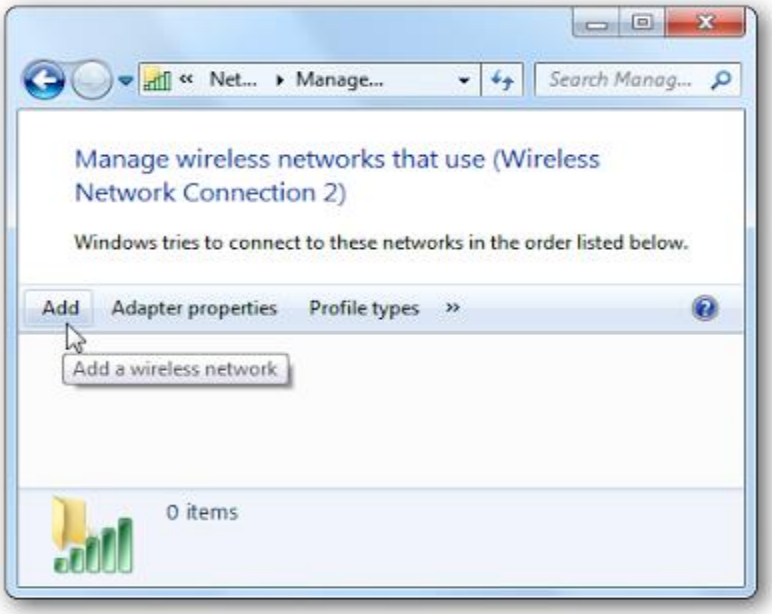
An ad hoc network is a local area network you can build spontaneously in a pinch. It allows computers and devices to communicate directly with each other within a small radius. The ability to create an ad hoc network has been possible since XP and is a way to quickly share data and an Internet connection to other wireless devices. For this tutorial you'll need the host computer to be hardwired and have wireless capabilities such as a Netbook or Laptop.

Create an ad hoc Network

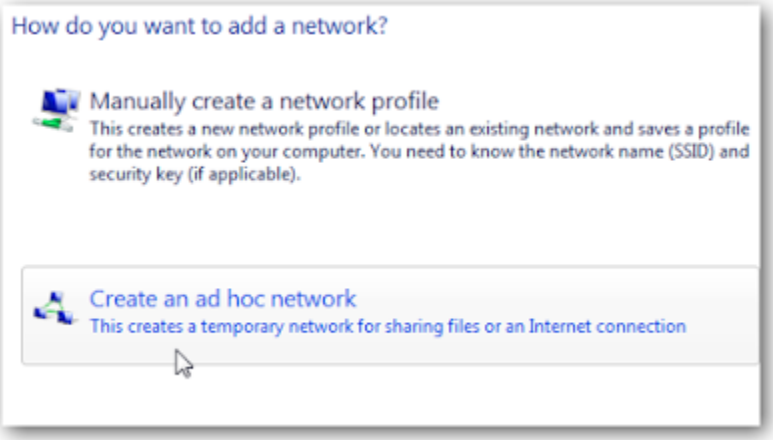
Open the Start Menu and type *wireless* into the Search box and select *Manage wireless networks*.



Click the Add button to add a wireless network.



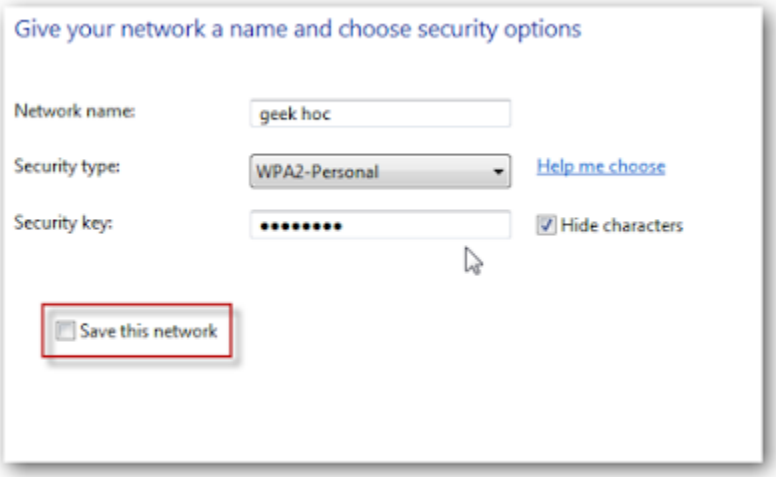
Next click on *Create an ad hoc network*…



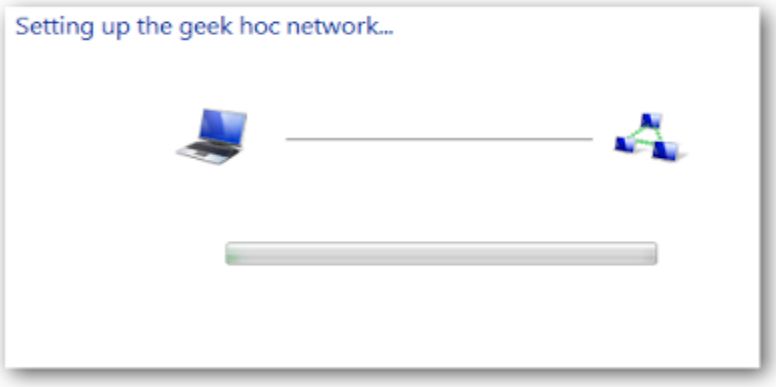
You'll get a message instructing you of what an ad hoc network is…click Next.

Now give your network a name and choose security options…check *Save this network* if you plan to use it repeatedly so you don't have to set one up each time. The security type will depend on

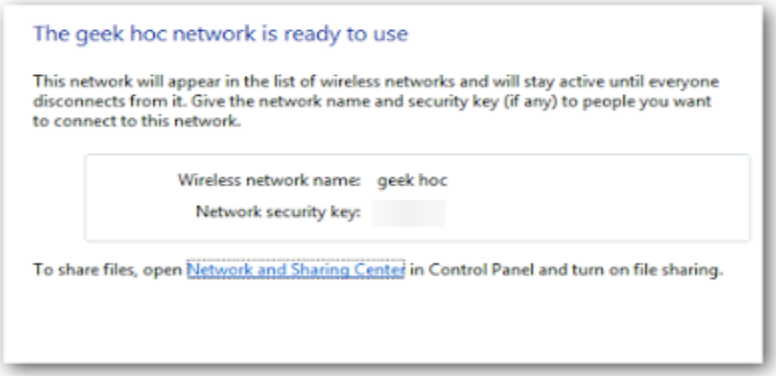
what the wireless adapter is capable of. We found if you're only using it quickly with different types of devices it's easier to have no security. Of course if you are keeping it on for full-time use, you'll definitely want to use security.



Wait while the network is created…

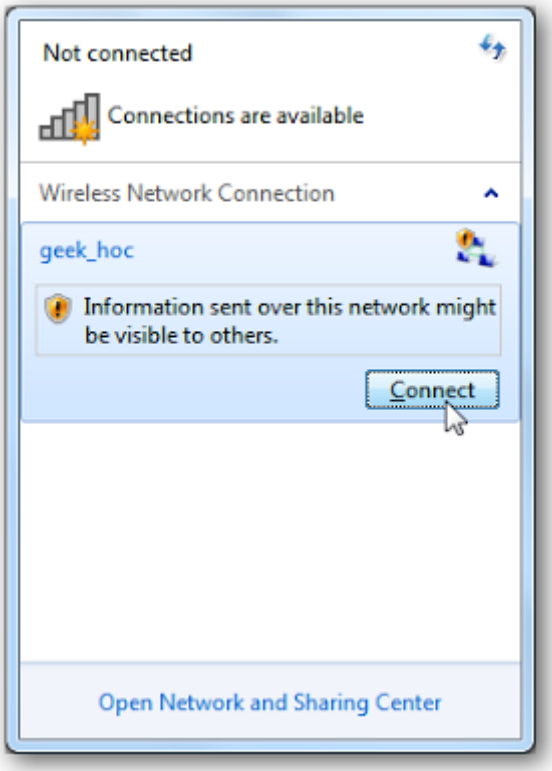


The network was successfully created and is ready to use.

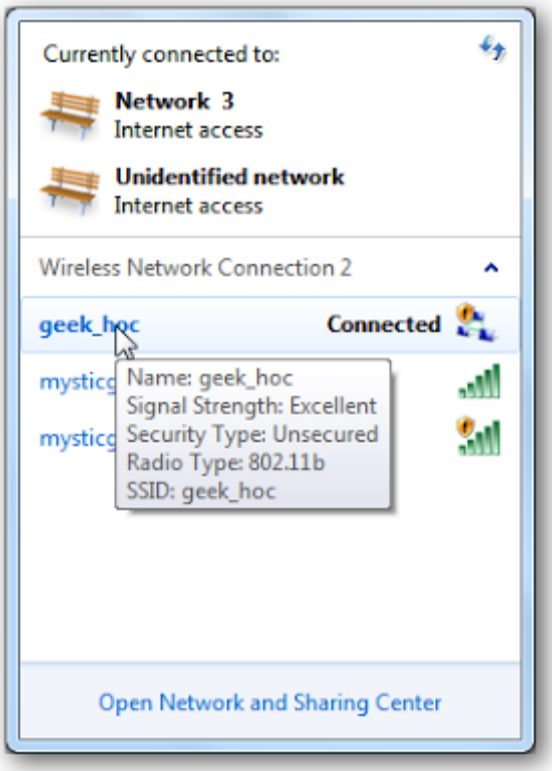


Verify the network by clicking the wireless icon on the Taskbar…here you can see ours is waiting for users to connect.

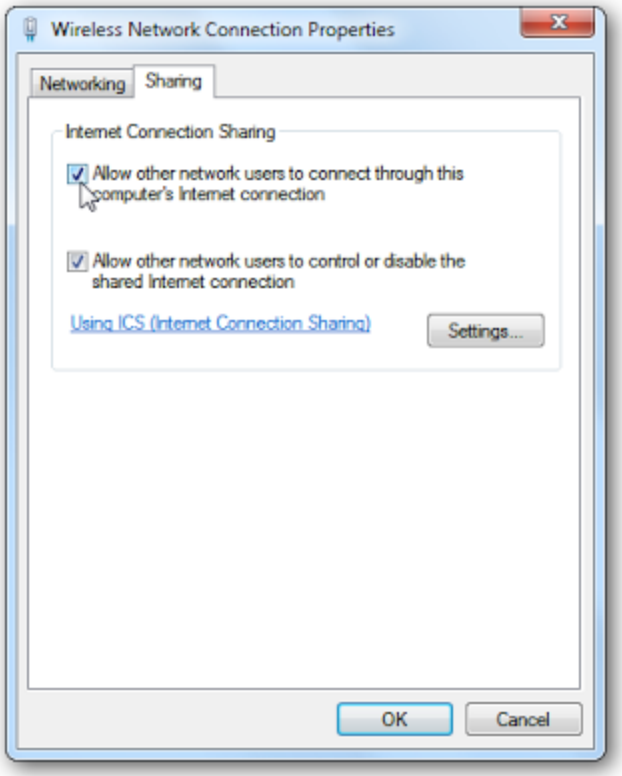
On the Client computer, click on the wireless icon to see available networks and connect to the ad hoc network.



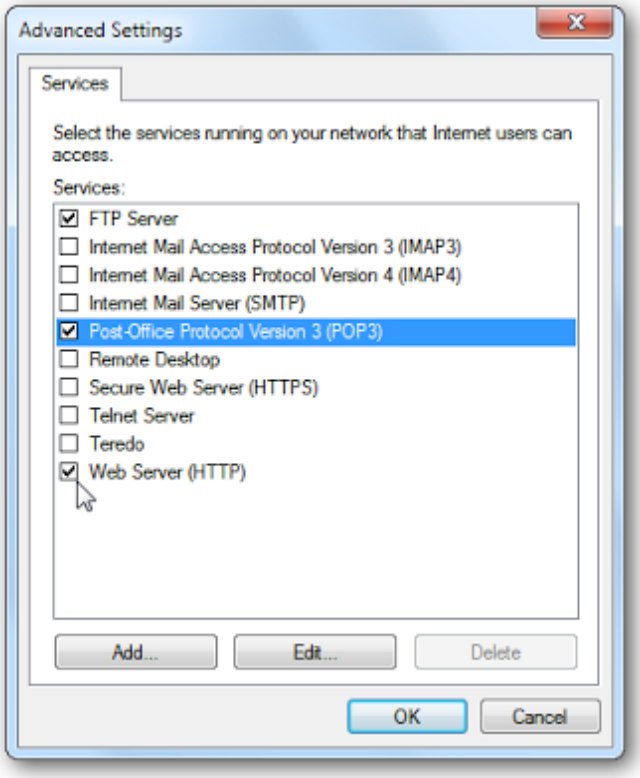
Then on the Host machine you will see that the other computer is connected to your ad hoc network.



At this point you'll be able to share files between machines. To share an Internet connection with other devices, right-click on the Host's wireless adapter icon. Go to Properties, click the Sharing tab, and select *Allow other network users to connect through this computer's Internet connection*.



In Settings you can select the network services the Client machine can access.



Now you can share documents and the Internet connection between machines and devices. In this example we're sharing between a Windows 7 desktop with a wireless card and a Netbook with Windows 7 Home Premium.