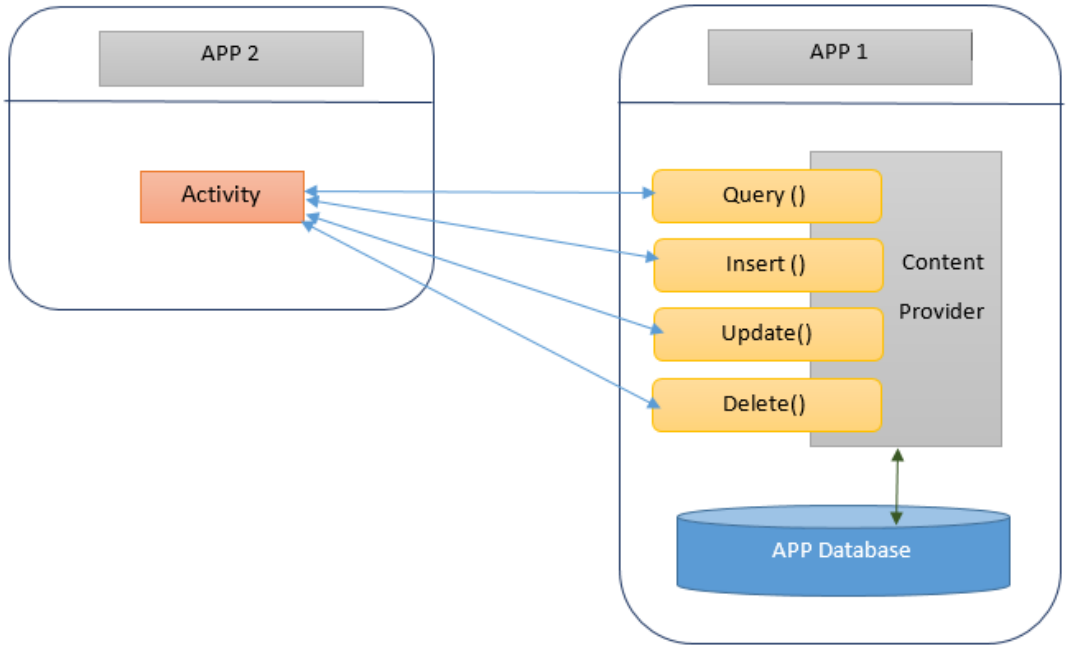
Content Provider

Background

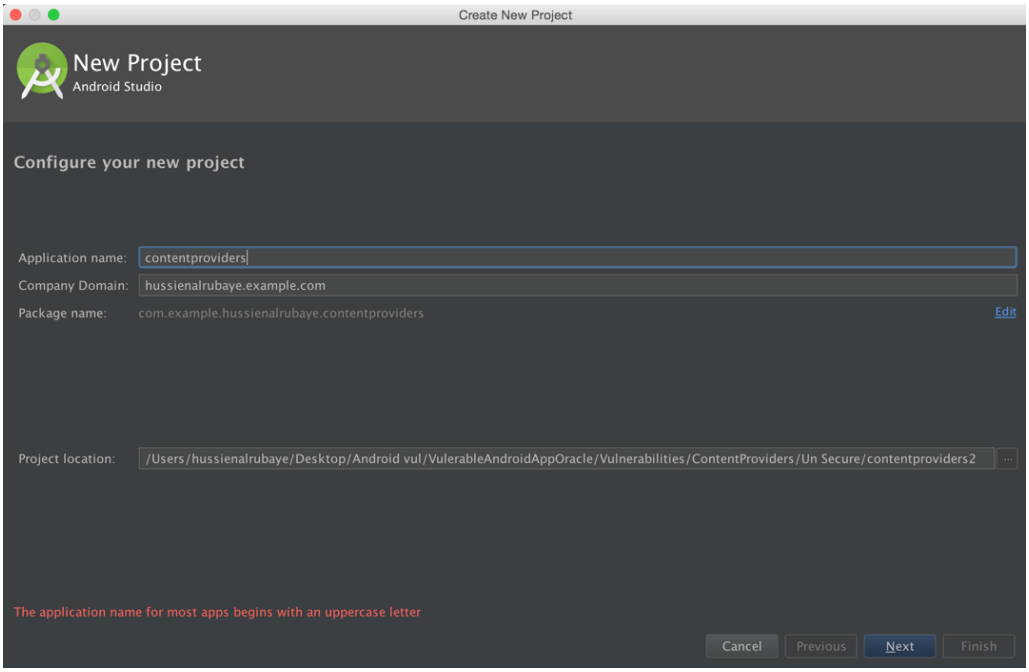
Content Providers allow one app to share its data with another app. They are the standard interface for code in one process to connect with data in another. Many Android apps have data specific to the app, while in some cases, we may need to share app data with another app. For instance, consider the default contact app that has all our contact information. Another app, such as an instant messaging app that needs the user to pick one or more contacts, as part of their actions, may need to access contact information. This requires the contact app to have contact information on the device managed by a content provider for other apps to read from. The Messenger (default messaging app) app is another example of an app that requires access to contact information.

The security weakness here is that if we develop an app that shares data, we must make sure that we are aware of what the data constitutes and what other apps have access to the data, especially, if we need to share sensitive information. For example, if we save the user's credit card information, in plain text, encapsulated by a content provider, another app on the device may use it without having the user to enter it again. We must configure content providers to allow secure access by other applications.

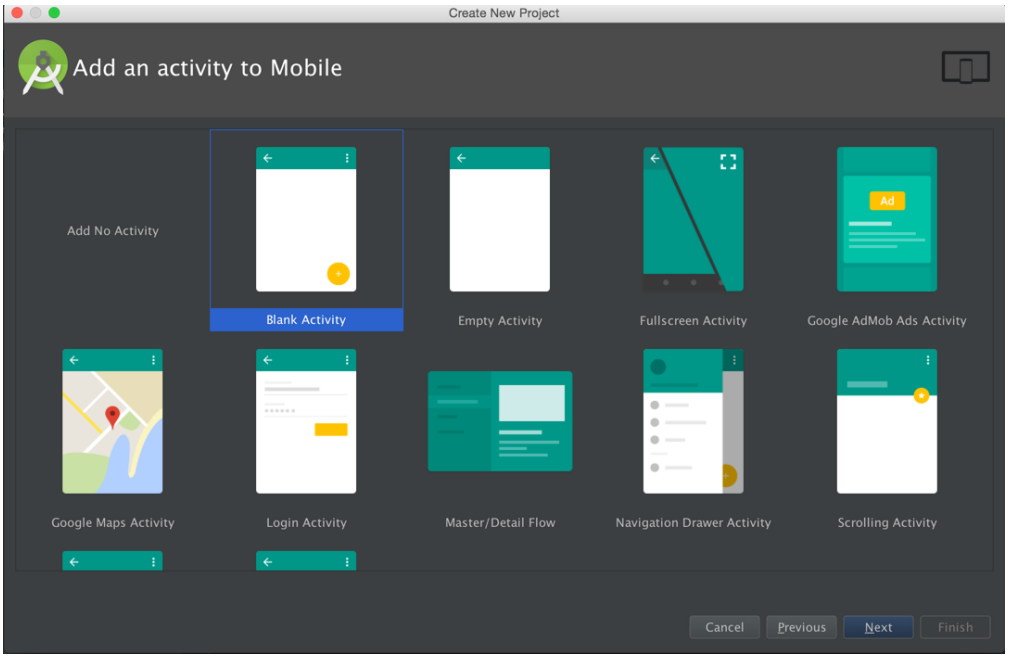


We will demonstrate an example, showing how one app stores a student's name and age behind a content provider. Then we shall see how we will share this data only with the apps we trust. We shall also see how easy it for a hacker to read this data if it is not encrypted.

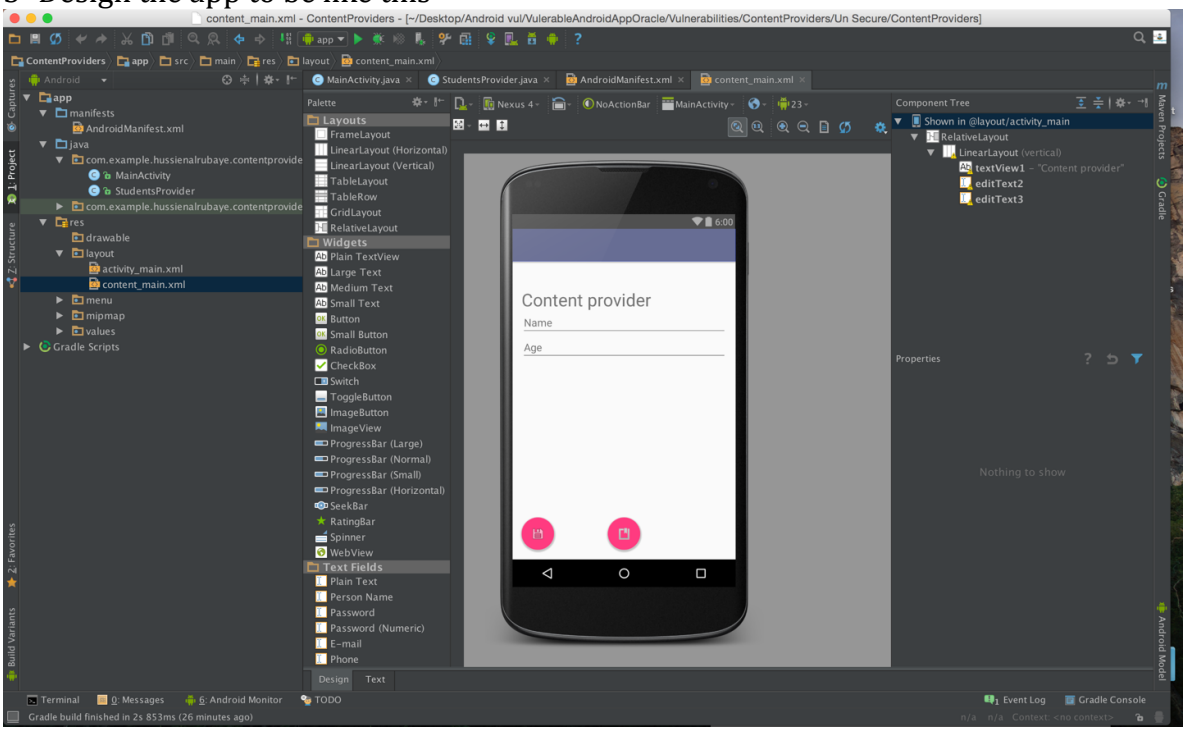
Activity Instructions
Our sender app : this app will write content provider
1-Create new project named “contentproviders”, and the make sure to remember package name.



2- select project from type Blank Activity



3- Design the app to be like this



4- update **Content_main.xml** to be like this

Java

```
 <?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:paddingBottom="@dimen/activity_vertical_margin"
   android:paddingLeft="@dimen/activity_horizontal_margin"
   android:paddingRight="@dimen/activity_horizontal_margin"
   android:paddingTop="@dimen/activity_vertical_margin"
   app:layout_behavior="@string/appbar_scrolling_view_behavior"
   tools:context="com.example.hussienalrubaye.contentproviders.MainActivity"
   tools:showIn="@layout/activity_main">


   <LinearLayout
      android:textAlignment="center"
      android:orientation="vertical"
      android:layout_width="match_parent"
      android:layout_height="match_parent"
```

```
    android:paddingTop="33dp">

    <TextView
      android:id="@+id/textView1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Content provider"
      android:layout_alignParentTop="true"
      android:layout_centerHorizontal="true"
      android:textSize="30dp" />


    <EditText
      android:layout_width="fill_parent"
      android:layout_height="wrap_content"
      android:id="@+id/editText2"
      android:layout_alignTop="@+id/editText"
      android:layout_alignLeft="@+id/textView1"
      android:layout_alignStart="@+id/textView1"
      android:layout_alignRight="@+id/textView1"
      android:layout_alignEnd="@+id/textView1"
      android:hint="Name"
      />

    <EditText
      android:layout_width="fill_parent"
      android:layout_height="wrap_content"
      android:id="@+id/editText3"
      android:layout_below="@+id/editText"
      android:layout_alignLeft="@+id/editText2"
      android:layout_alignStart="@+id/editText2"
      android:layout_alignRight="@+id/editText2"
      android:layout_alignEnd="@+id/editText2"
      android:hint="Age"
       />

  </LinearLayout>
</RelativeLayout>
```

5- update **Activity_main.xml** to be like this

Java

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="http://schemas.android.com/apk/res-auto"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:fitsSystemWindows="true"
```

```
    tools:context="com.example.hussienalrubaye.contentproviders.MainActivity">

    <android.support.design.widget.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/AppTheme.AppBarOverlay">

        <android.support.v7.widget.Toolbar
            android:id="@+id/toolbar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:background="?attr/colorPrimary"
            app:popupTheme="@style/AppTheme.PopupOverlay" />

    </android.support.design.widget.AppBarLayout>

    <include layout="@layout/content_main" />

    <android.support.design.widget.FloatingActionButton
        android:id="@+id/fbSave"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom|left"
        android:layout_margin="@dimen/fab_margin"
        android:src="@android:drawable/ic_menu_save" />
    <android.support.design.widget.FloatingActionButton
        android:id="@+id/fbQuery"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom"
        android:layout_margin="@dimen/fab_margin"
        android:src="@android:drawable/ic_input_get"
        />


</android.support.design.widget.CoordinatorLayout>
```

6- add class named "**StudentProvider.java**"

Java

```
package com.example.hussienalrubaye.contentproviders;

/**
 * Created by hussienalrubaye on 3/6/16.
 */

import java.util.HashMap;

import android.content.ContentProvider;
import android.content.ContentUris;
import android.content.ContentValues;
import android.content.Context;
```

```
import android.content.UriMatcher;

import android.database.Cursor;
import android.database.SQLException;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.database.sqlite.SQLiteQueryBuilder;

import android.net.Uri;
import android.text.TextUtils;

public class StudentsProvider extends ContentProvider {

   static final String PROVIDER_NAME = "com.example.provider.College";
   static final String URL = "content://" + PROVIDER_NAME + "/students";
   static final Uri CONTENT_URI = Uri.parse(URL);

   static final String _ID = "_id";
   static final String NAME = "name";
   static final String Age = "age";

   private static HashMap<String, String> STUDENTS_PROJECTION_MAP;

   static final int STUDENTS = 1;
   static final int STUDENT_ID = 2;

   static final UriMatcher uriMatcher;
   static{
      uriMatcher = new UriMatcher(UriMatcher.NO_MATCH);
      uriMatcher.addURI(PROVIDER_NAME, "students", STUDENTS);
      uriMatcher.addURI(PROVIDER_NAME, "students/#", STUDENT_ID);
   }

   /**
    * Database specific constant declarations
    */
   private SQLiteDatabase db;
   static final String DATABASE_NAME = "College";
   static final String STUDENTS_TABLE_NAME = "students";
   static final int DATABASE_VERSION = 1;
   static final String CREATE_DB_TABLE =
         " CREATE TABLE IF NOT EXISTS " + STUDENTS_TABLE_NAME +
               " (_id INTEGER PRIMARY KEY AUTOINCREMENT, " +
               " name TEXT NOT NULL, " +
               " age TEXT NOT NULL);";

   /**
    * Helper class that actually creates and manages
    * the provider's underlying data repository.
    */
   private static class DatabaseHelper extends SQLiteOpenHelper {
      DatabaseHelper(Context context){
         super(context, DATABASE_NAME, null, DATABASE_VERSION);
      }

      @Override
      public void onCreate(SQLiteDatabase db)
      {
```

```
        db.execSQL(CREATE_DB_TABLE);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " +  STUDENTS_TABLE_NAME);
        onCreate(db);
    }
}

@Override
public boolean onCreate() {
    Context context = getContext();
    DatabaseHelper dbHelper = new DatabaseHelper(context);

    /**
     * Create a write able database which will trigger its
     * creation if it doesn't already exist.
     */
    db = dbHelper.getWritableDatabase();
    return (db == null)? false:true;
}

@Override
public Uri insert(Uri uri, ContentValues values)
{
    /**
     * Add a new student record
     */
    long rowID = db.insert(    STUDENTS_TABLE_NAME, "", values);

    /**
     * If record is added successfully
     */

    if (rowID > 0)
    {
        Uri _uri = ContentUris.withAppendedId(CONTENT_URI, rowID);
        getContext().getContentResolver().notifyChange(_uri, null);
        return _uri;
    }
    throw new SQLException("Failed to add a record into " + uri);
}
/*
//selection select "name=12 and age =122"
// String[] projection = new String[] { "_id", "name", "age" };
String selection = " age in (?, ?, ?)";
String selectionArgs[] = new String[]{"1","2","3"};
*/
@Override
public Cursor query(Uri uri, String[] projection, String selection,String[] selectionArgs, String sortOrder)
{
    SQLiteQueryBuilder qb = new SQLiteQueryBuilder();
    qb.setTables(STUDENTS_TABLE_NAME);

    switch (uriMatcher.match(uri)) {
        case STUDENTS:
            qb.setProjectionMap(STUDENTS_PROJECTION_MAP);
```

```
            break;

        case STUDENT_ID:
            qb.appendWhere( _ID + "=" + uri.getPathSegments().get(1));
            break;

        default:
            throw new IllegalArgumentException("Unknown URI " + uri);
    }

    if (sortOrder == null || sortOrder == ""){
        /**
         * By default sort on student names
         */
        sortOrder = NAME;
    }
    Cursor c = qb.query(db,    projection,    selection, selectionArgs,null, null, sortOrder);

    /**
     * register to watch a content URI for changes
     */
    c.setNotificationUri(getContext().getContentResolver(), uri);
    return c;
}

@Override
public int delete(Uri uri, String selection, String[] selectionArgs)
{
    int count = 0;

    switch (uriMatcher.match(uri)){
        case STUDENTS:
            count = db.delete(STUDENTS_TABLE_NAME, selection, selectionArgs);
            break;

        case STUDENT_ID:
            String id = uri.getPathSegments().get(1);
            count = db.delete( STUDENTS_TABLE_NAME, _ID +  " = " + id +
                    (!TextUtils.isEmpty(selection) ? " AND (" + selection + ')' : ""), selectionArgs);
            break;

        default:
            throw new IllegalArgumentException("Unknown URI " + uri);
    }

    getContext().getContentResolver().notifyChange(uri, null);
    return count;
}

@Override
public int update(Uri uri, ContentValues values, String selection, String[] selectionArgs)
{
    int count = 0;

    switch (uriMatcher.match(uri)){
        case STUDENTS:
            count = db.update(STUDENTS_TABLE_NAME, values, selection, selectionArgs);
            break;
```

```
        case STUDENT_ID:
            count = db.update(STUDENTS_TABLE_NAME, values, _ID + " = " + uri.getPathSegments().get(1) +
                    (!TextUtils.isEmpty(selection) ? " AND (" +selection + ')' : ""), selectionArgs);
            break;

        default:
            throw new IllegalArgumentException("Unknown URI " + uri );
    }
    getContext().getContentResolver().notifyChange(uri, null);
    return count;
  }

  @Override
  public String getType(Uri uri) {
    switch (uriMatcher.match(uri)){
      /**
       * Get all student records (dir)
       * http://developer.android.com/reference/android/content/UriMatcher.html
       */
      case STUDENTS:
        return "vnd.android.cursor.dir/vnd.example.students";

      /**
       * Get a particular student(item)
       */
      case STUDENT_ID:
        return "vnd.android.cursor.item/vnd.example.students";

      default:
        throw new IllegalArgumentException("Unsupported URI: " + uri);
    } }}
```

7- Update **Manifest.xml** to be like this

Java

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

   <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"
         android:label="@string/app_name"
         android:theme="@style/AppTheme.NoActionBar">
         <intent-filter>
```

```
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
      </intent-filter>
    </activity>
  <provider
    android:authorities="com.example.provider.College"
    android:name="StudentsProvider" android:exported="true"
    />
  </application>

</manifest>
```

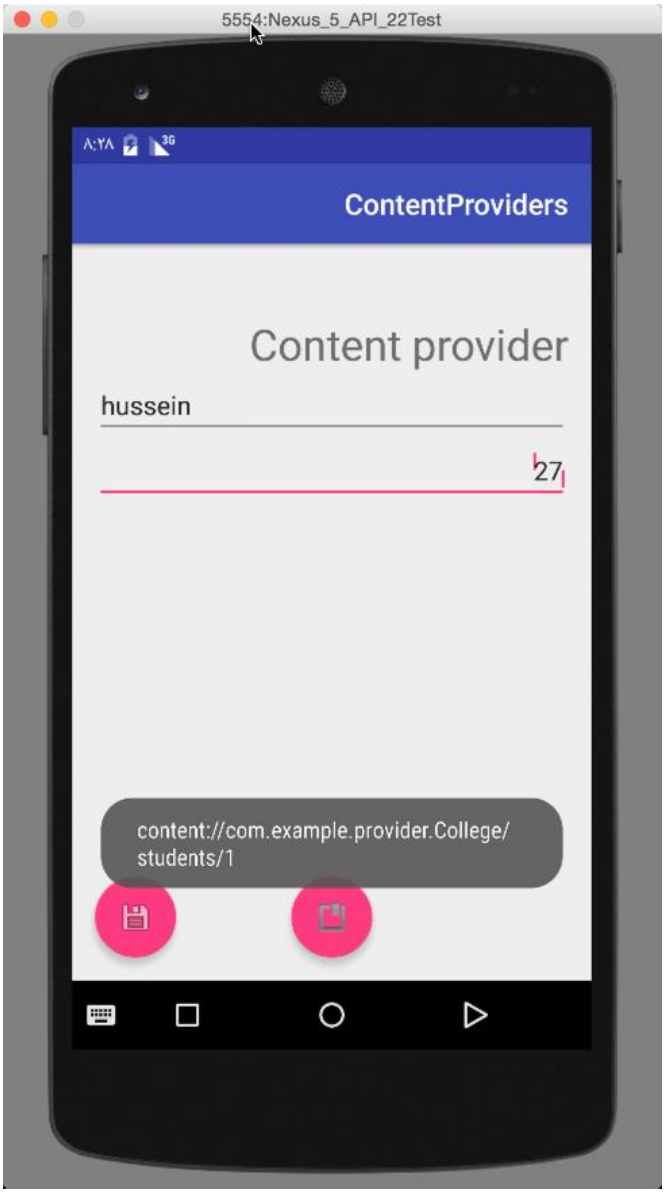
8- update **MainActivity.java** to be like this

Java

```
public class MainActivity extends AppCompatActivity {

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);

    FloatingActionButton fabAdd = (FloatingActionButton) findViewById(R.id.fbSave);
    fabAdd.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View view) {
        // add new record
       Add();
      }
    });
    FloatingActionButton fbQuery = (FloatingActionButton) findViewById(R.id.fbQuery);
    fbQuery.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View view) {
        //load all records
        QueryGet();
      }
    });
  }

  // Retrieve student records
  String URL = "content://com.example.provider.College/students";

  public void Add() {
    // Add a new student record
    ContentValues values = new ContentValues();
// insert value
    values.put(StudentsProvider.NAME,
((EditText)findViewById(R.id.editText2)).getText().toString());

    values.put(StudentsProvider.Age,
        ((EditText)findViewById(R.id.editText3)).getText().toString());
// define the play to insert the values in
```

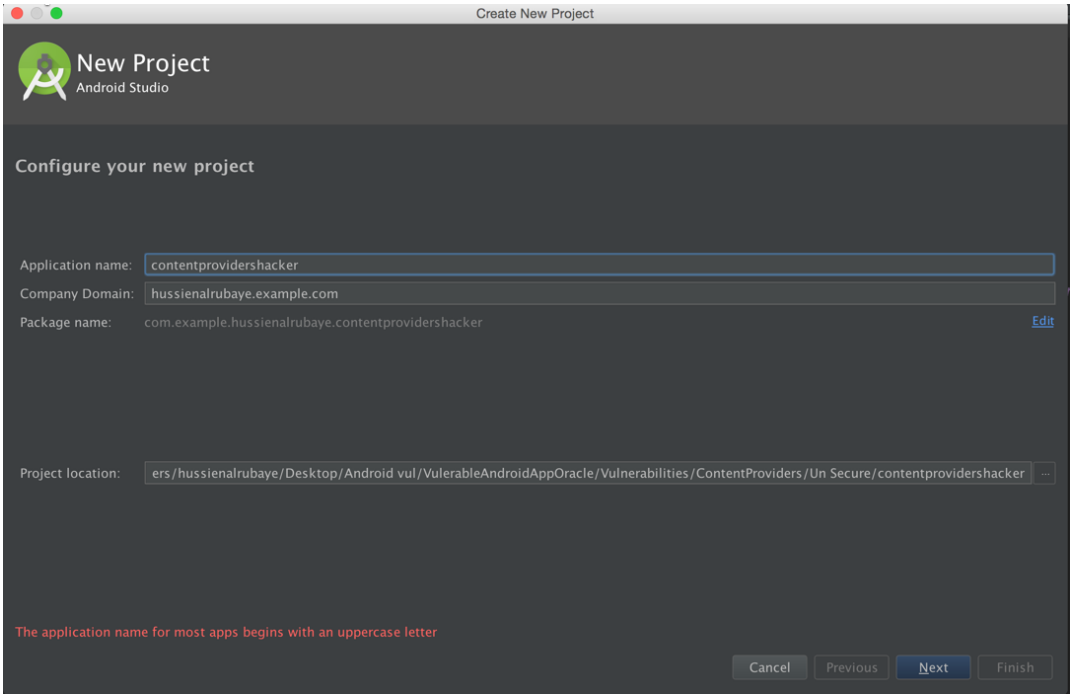
```
    Uri uri = getContentResolver().insert(
          StudentsProvider.CONTENT_URI, values);
// display messages
    Toast.makeText(getBaseContext(),
          uri.toString(), Toast.LENGTH_LONG).show();
  }

  public void QueryGet() {

 // define content provider url to read from
    Uri students = Uri.parse(URL);
// get data ordered by name
    Cursor c = getContentResolver().query(students, null, null, null, "name");
// move through all items
    if (c.moveToFirst()) {
      do{
        // load the record name and age and id
        Toast.makeText(this,
            c.getString(c.getColumnIndex(StudentsProvider._ID)) +
              ", " +  c.getString(c.getColumnIndex( StudentsProvider.NAME)) +
              ", " + c.getString(c.getColumnIndex( StudentsProvider.Age)),
            Toast.LENGTH_SHORT).show();
      } while (c.moveToNext());
    }
  }
}
```

5- Run the app you will see this output, add one student name “Hussein, age “27”

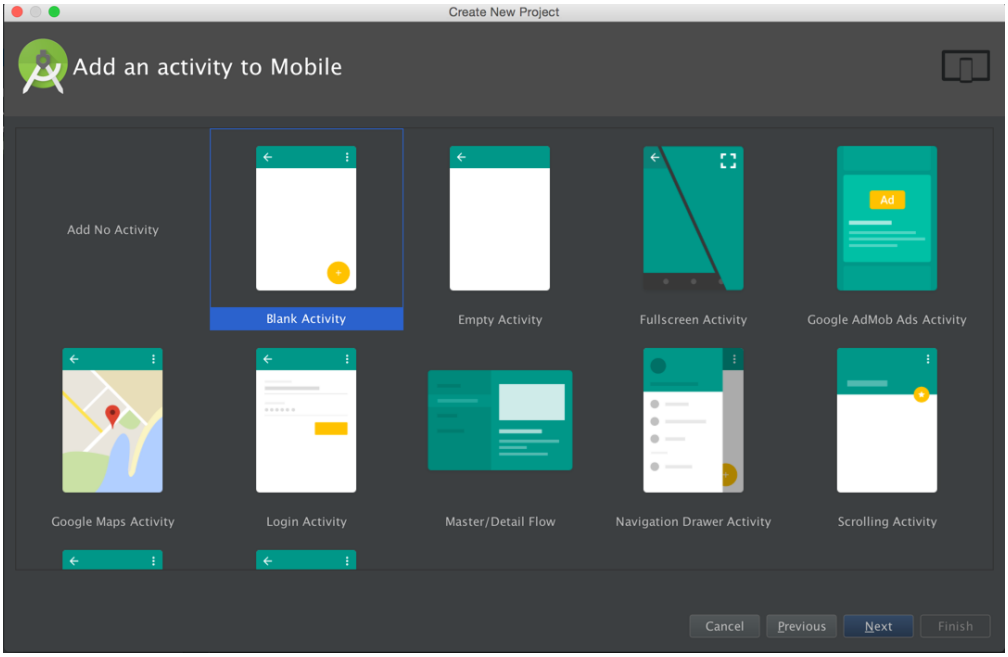


Our receiver app: this app will read from content provider

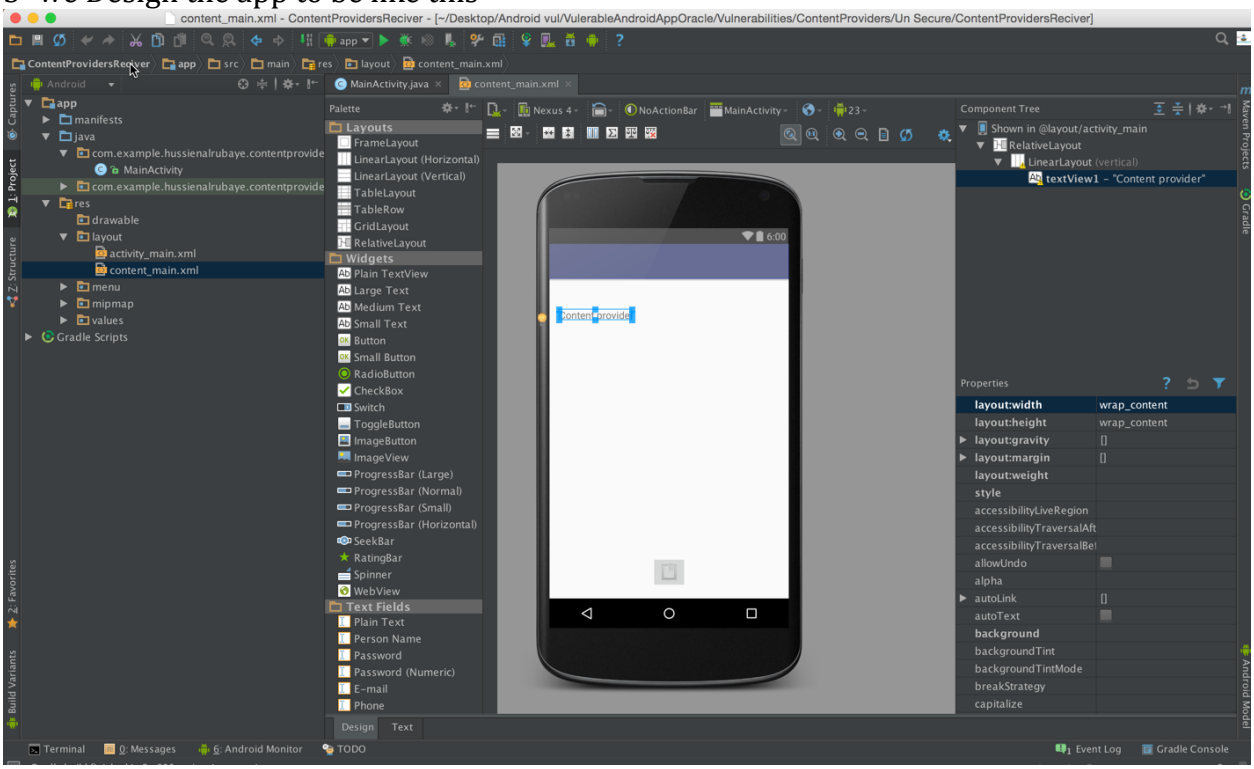
1-Create new project named “contentprovidershacker”, and the make sure to remember package name.



2- select project from type Blank Activity



3- we Design the app to be like this



4- update **Content_main.xml** to be like this

Java

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:paddingBottom="@dimen/activity_vertical_margin"
   android:paddingLeft="@dimen/activity_horizontal_margin"
   android:paddingRight="@dimen/activity_horizontal_margin"
   android:paddingTop="@dimen/activity_vertical_margin"
   app:layout_behavior="@string/appbar_scrolling_view_behavior"
   tools:context="com.example.hussienalrubaye.contentprovidershacker.MainActivity"
   tools:showIn="@layout/activity_main">

   <LinearLayout
      android:textAlignment="center"
      android:orientation="vertical"
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:paddingTop="33dp">

      <TextView
         android:id="@+id/textView1"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:text="Content provider"
         android:layout_alignParentTop="true"
         android:layout_centerHorizontal="true"
         android:textSize="16dp" />




   </LinearLayout>
</RelativeLayout>
```

5- update **Activity_main.xml** to be like this

Java

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:fitsSystemWindows="true"
   tools:context="com.example.hussienalrubaye.contentprovidershacker.MainActivity">

   <android.support.design.widget.AppBarLayout
```

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:theme="@style/AppTheme.AppBarOverlay">

    <android.support.v7.widget.Toolbar
      android:id="@+id/toolbar"
      android:layout_width="match_parent"
      android:layout_height="?attr/actionBarSize"
      android:background="?attr/colorPrimary"
      app:popupTheme="@style/AppTheme.PopupOverlay" />

  </android.support.design.widget.AppBarLayout>

  <include layout="@layout/content_main" />


  <android.support.design.widget.FloatingActionButton
    android:id="@+id/fbQuery"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="bottom"
    android:layout_margin="@dimen/fab_margin"
    android:src="@android:drawable/ic_input_get"
    />

</android.support.design.widget.CoordinatorLayout>
```

6- add class named "**MainActivity.java**"

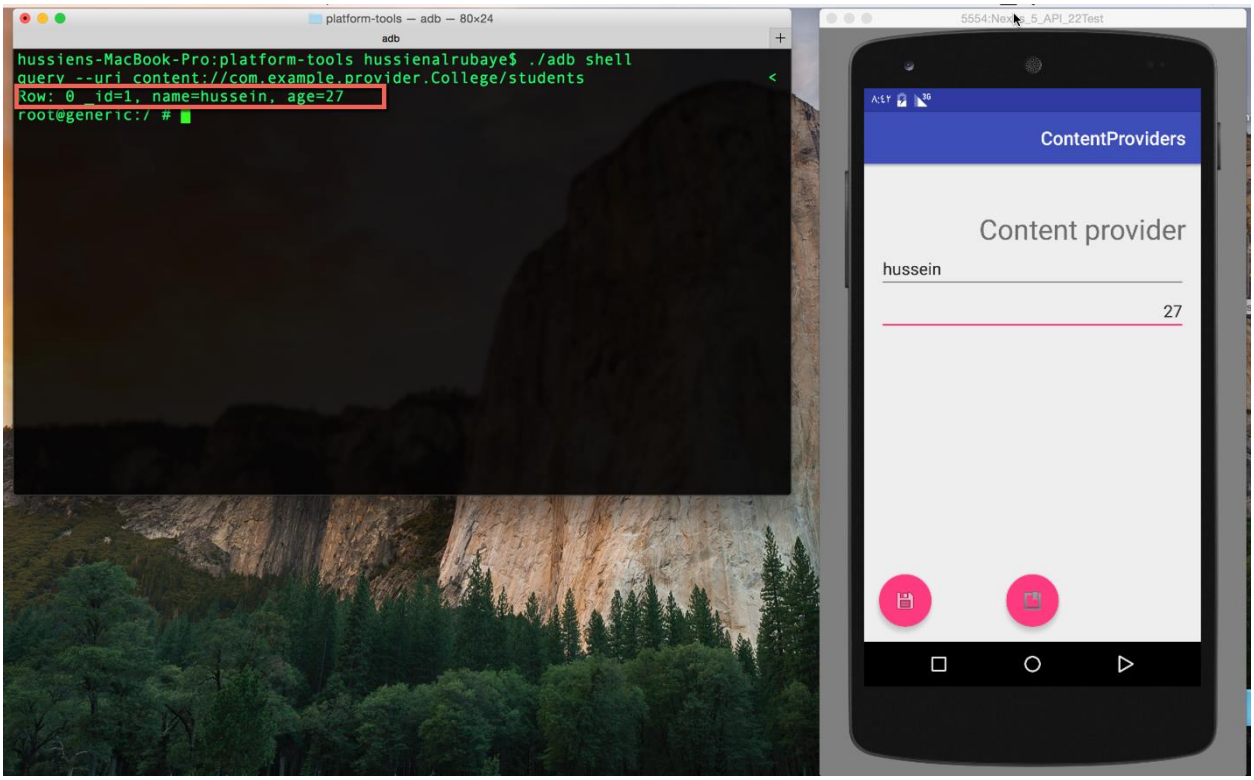
Java

```
public class MainActivity extends AppCompatActivity {
TextView textView1;
  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);
    textView1=(TextView)findViewById(R.id.textView1);

    FloatingActionButton fbQuery = (FloatingActionButton) findViewById(R.id.fbQuery);
    fbQuery.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View view) {
        QueryGet();
      }
    });
  }

  // Retrieve student records
  String URL = "content://com.example.provider.College/students";
  static final String _ID = "_id";
  static final String NAME = "name";
```

```
   static final String GRADE = "age";


   public void QueryGet() {


      Uri students = Uri.parse(URL);
      Cursor c = getContentResolver().query(students, null, null, null, "name");
String data="";
      if (c.moveToFirst()) {
         do{
            data+=(c.getString(c.getColumnIndex( _ID)) +
                        ", " +  c.getString(c.getColumnIndex( NAME)) +
                        ", " + c.getString(c.getColumnIndex( GRADE)));
         } while (c.moveToNext());


      }
      textView1.setText(data);
   }
}
```

Run the app and click the button you will see the content data stored

What the hacker can do:
It easy to the hacker to read this data by opening the app manifest.xml file and read the content provider URL, then going to the terminal "adb" and run this command.
./adb shell
Content query --uri "content provider url"



How to protect our Data:

We need to encrypt and decrypt the data, so when we save into content provider we have to encrypt and when we read we have to decrypt.

1- Update MaiActivity.java in “contentProvidershacker” to be like this

Java

```
public class MainActivity extends AppCompatActivity {
TextView textView1;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
      setSupportActionBar(toolbar);
      textView1=(TextView)findViewById(R.id.textView1);

      FloatingActionButton fbQuery = (FloatingActionButton) findViewById(R.id.fbQuery);
      fbQuery.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View view) {
            QueryGet();
         }
      });
   }

   // Retrieve student records
   String URL = "content://com.example.provider.College/students";
   static final String _ID = "_id";
   static final String NAME = "name";
   static final String GRADE = "age";


   public void QueryGet() {


      Uri students = Uri.parse(URL);
      Cursor c = getContentResolver().query(students, null, null, null, "name");
String data="";
      if (c.moveToFirst()) {
         do{
            data+=(c.getString(c.getColumnIndex( _ID)) +
                        ", " + cipher(c.getString(c.getColumnIndex( NAME)) ,-10)+
                        ", " + cipher(c.getString(c.getColumnIndex( GRADE)),-10));
         } while (c.moveToNext());


      }
      textView1.setText(data);
   }

   // cipher encryption add shift for key
   public   String cipher(String msg, int shift) {
      String s = "";
```

```
    int len = msg.length(); // get string length
    for (int x = 0; x < len; x++) {
      char c = (char) (msg.charAt(x) + shift);  // shift every character
      s += c; // append the characters
    }
    return s;
  }
}
```

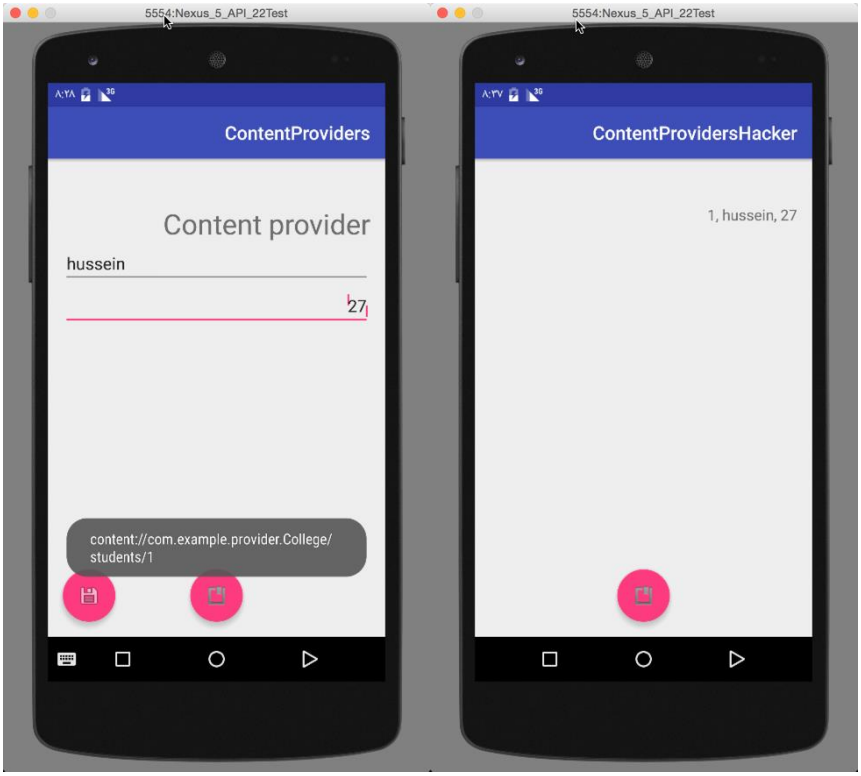
2- Update MaiActivity.java in “contentProviders” to be like this

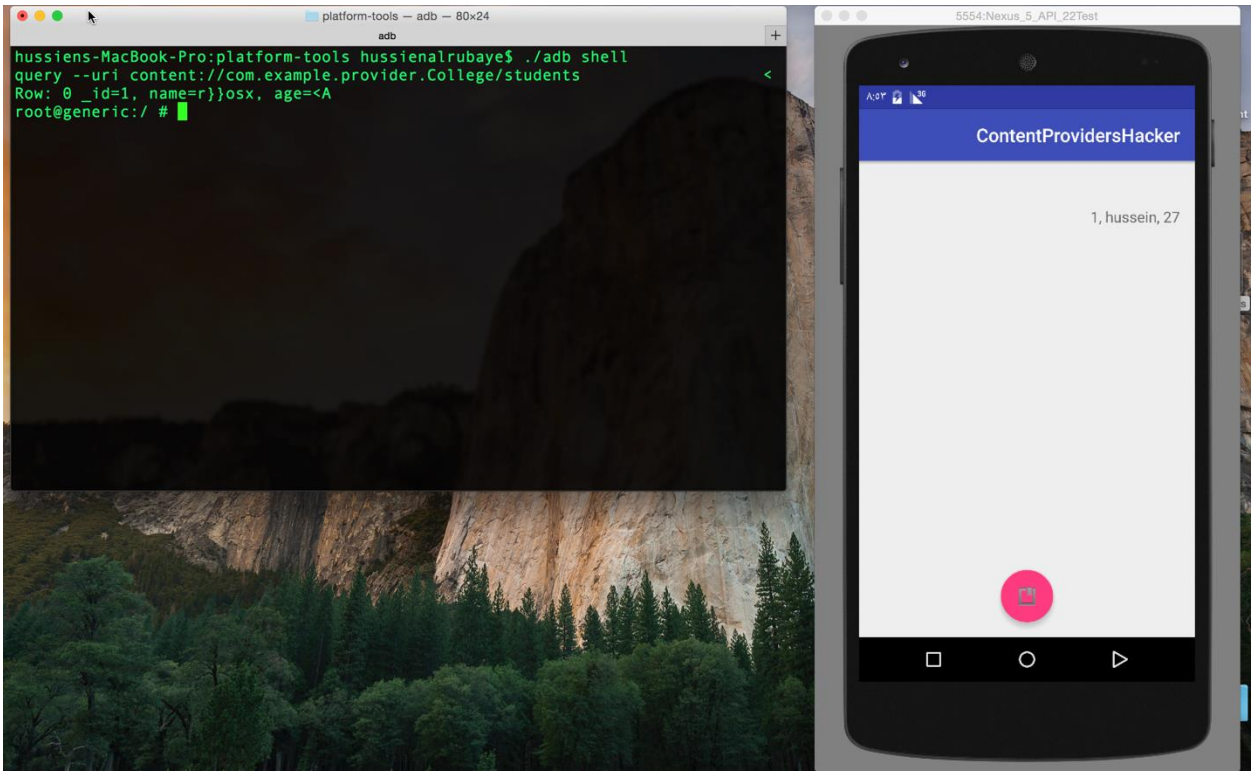
Java

```
public class MainActivity extends AppCompatActivity {

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);

    FloatingActionButton fabAdd = (FloatingActionButton) findViewById(R.id.fbSave);
    fabAdd.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View view) {
        Add();
      }
    });
    FloatingActionButton fbQuery = (FloatingActionButton) findViewById(R.id.fbQuery);
    fbQuery.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View view) {
        QueryGet();
      }
    });
  }

  // Retrieve student records
  String URL = "content://com.example.provider.College/students";

  public void Add() {
    // Add a new student record
    ContentValues values = new ContentValues();
// insert value
    values.put(StudentsProvider.NAME,
        cipher(((EditText) findViewById(R.id.editText2)).getText().toString(), 10));

    values.put(StudentsProvider.Age,
        cipher(((EditText) findViewById(R.id.editText3)).getText().toString(), 10));
// define the play to insert the values in
    Uri uri = getContentResolver().insert(
        StudentsProvider.CONTENT_URI, values);
```

```
// display messages
    Toast.makeText(getBaseContext(),
          uri.toString(), Toast.LENGTH_LONG).show();
  }

  public void QueryGet() {


    Uri students = Uri.parse(URL);

    Cursor c = getContentResolver().query(students, null, null, null, "name");

    if (c.moveToFirst()) {
      do {
        Toast.makeText(this,
              c.getString(c.getColumnIndex(StudentsProvider._ID))+
                  ", " + cipher(c.getString(c.getColumnIndex(StudentsProvider.NAME)),-10)
                   +
                  ", " +cipher( c.getString(c.getColumnIndex(StudentsProvider.Age)),-10) ,
            Toast.LENGTH_SHORT).show();
      } while (c.moveToNext());
    }
  }

  // cipher encryption add shift for key
    public   String cipher(String msg, int shift) {
      String s = "";
      int len = msg.length(); // get string length
      for (int x = 0; x < len; x++) {
        char c = (char) (msg.charAt(x) + shift);  // shift every character
        s += c; // append the characters
      }
      return s;
    }


}
```

Results:
Run the apps and see, as we see the hacker cannot read the data.




platform-tools — adb — 80×24
adb
hussiens-MacBook-Pro:platform-tools hussienalrubaye$ ./adb shell
query --uri content://com.example.provider.College/students
Row: 0 _id=1, name=r}}osx, age=<A
root@generic:/ #
5554:Nexus_5_API_22Test
ContentProvidersHacker
1, hussein, 27