# Android Internship Report

## Personal Details

| | | |
|---|---|---|
| Name | : | Shah Yash Harihar |
| College Name | : | Dharmsinh Desai University |
| Degree | : | B.Tech. |
| Semester | : | 7 |
| GitHub URL | : | https://github.com/syash2581 |
| Repo Name | : | 15DaysTrainingAtAP |

## Company Details

| | | |
|---|---|---|
| Company Name | : | Akash Technolabs |
| External Guide | : | Akash Padhiyar |
| Training Duration | : | 26-05-2021 to 21-06-2021 |

# Index

| Index | Task Detail | Page Name |
|---|---|---|
| 1 | Introduction to Android | 1 |
| 2 | Hello world application | 2 |
| 3 | Making a small application using UI controls and Use of Toast | 3 |
| 4 | Basic Calc (Addition and Subtraction) | 4 |
| 5 | Use of Explicit and implicit Intent | 7 |
| 6 | How to make splash screen | 10 |
| 7 | List View, Array Adapter (List.setAdapter), Grid View, Vertical and Horizontal ScrollBar | 13 |
| 8 | Custom List View and Grid View | 17 |
| 9 | SharedPreferences - getSharedPreferences and Editor | 21 |
| 10 | Making a call, Sending a mail, Sending a sms, Playing an audio using MediaPlayer, Playing a video using VideoView and MediaController | 25 |
| 11 | Android Application Project | 32 |

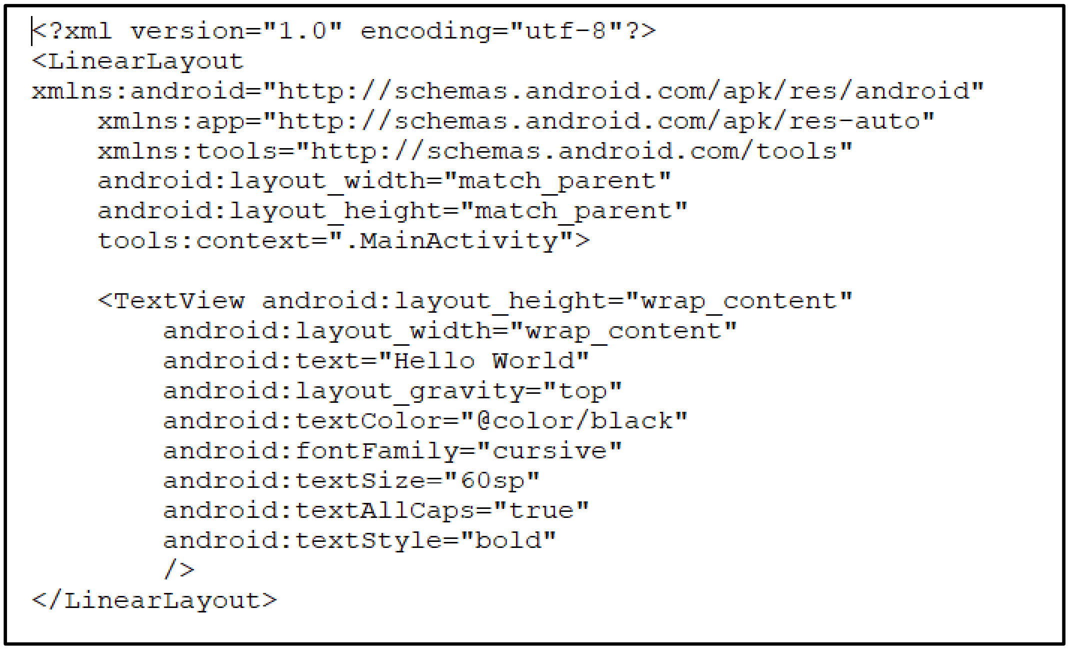
## 1 – Introduction to the android.

- ⇨ Evolution of android technology.
- ⇨ Current value of android technology in market
- ⇨ Ways to create applications.
- ⇨ Ways to host those applications.
- ⇨ What we are going to cover in these 15 days training.

## 2 – Hello world Application

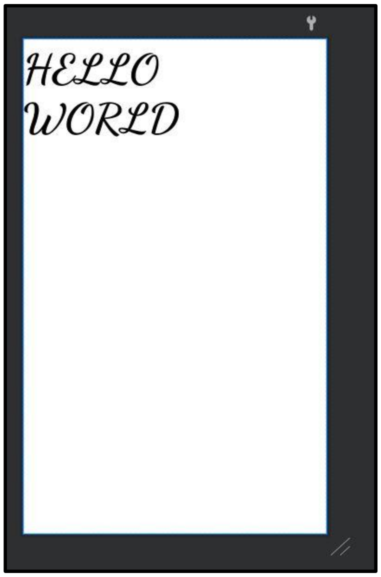
- Starting with android studio and SDK installation
- Introduction to AVD Manager in order to use virtual device for applications
- Creating your first android app
- Running it on to the simulator

### Code



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

    <TextView android:layout_height="wrap_content"
        android:layout_width="wrap_content"
        android:text="Hello World"
        android:layout_gravity="top"
        android:textColor="@color/black"
        android:fontFamily="cursive"
        android:textSize="60sp"
        android:textAllCaps="true"
        android:textStyle="bold"
        />
</LinearLayout>
```

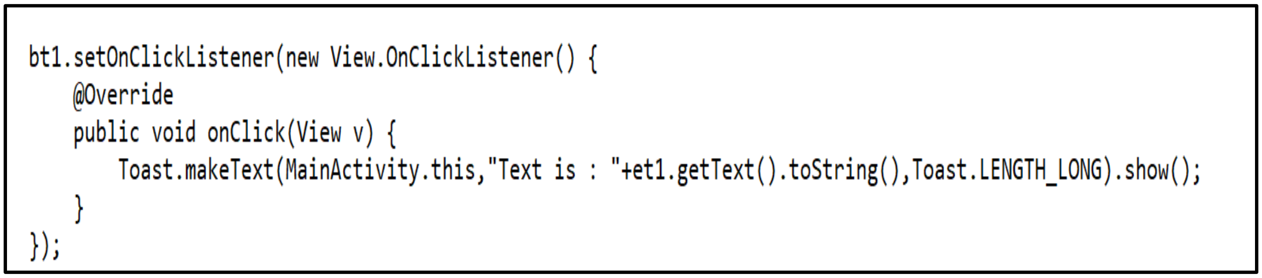
### Output



```
HELLO
WORLD
```

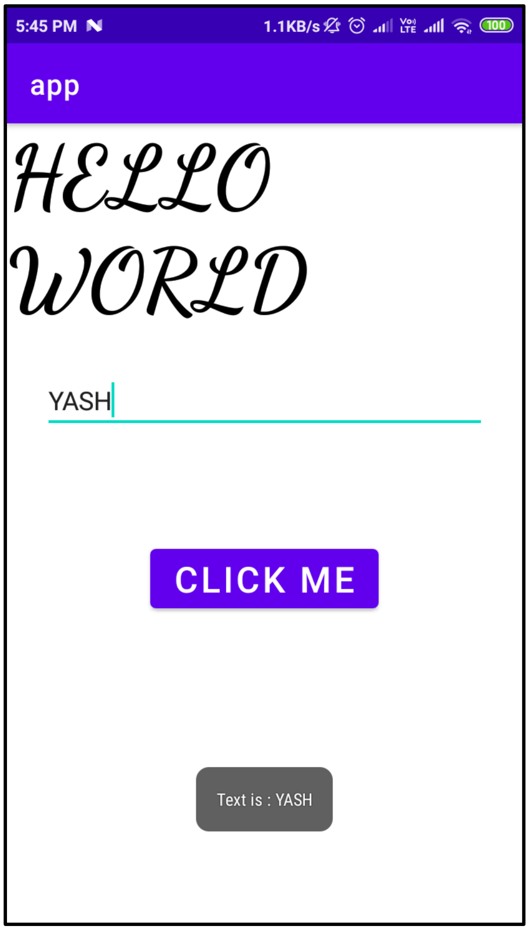
## 3 - Making a small application using UI controls and Use of Toast

Code



```
bt1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Toast.makeText(MainActivity.this,"Text is : "+et1.getText().toString(),Toast.LENGTH_LONG).show();
    }
});
```

Output



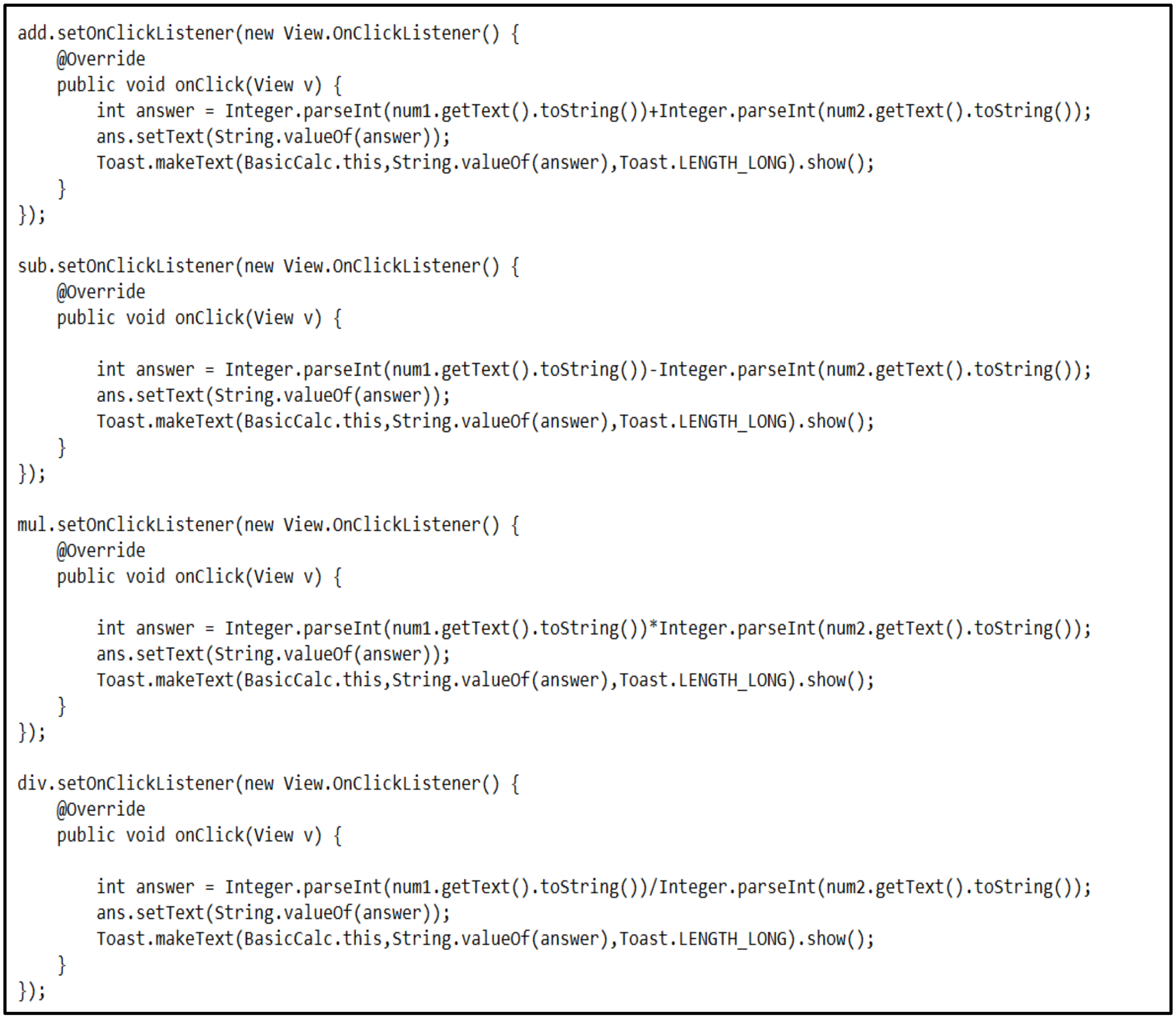
HELLO WORLD

YASH

Text is : YASH

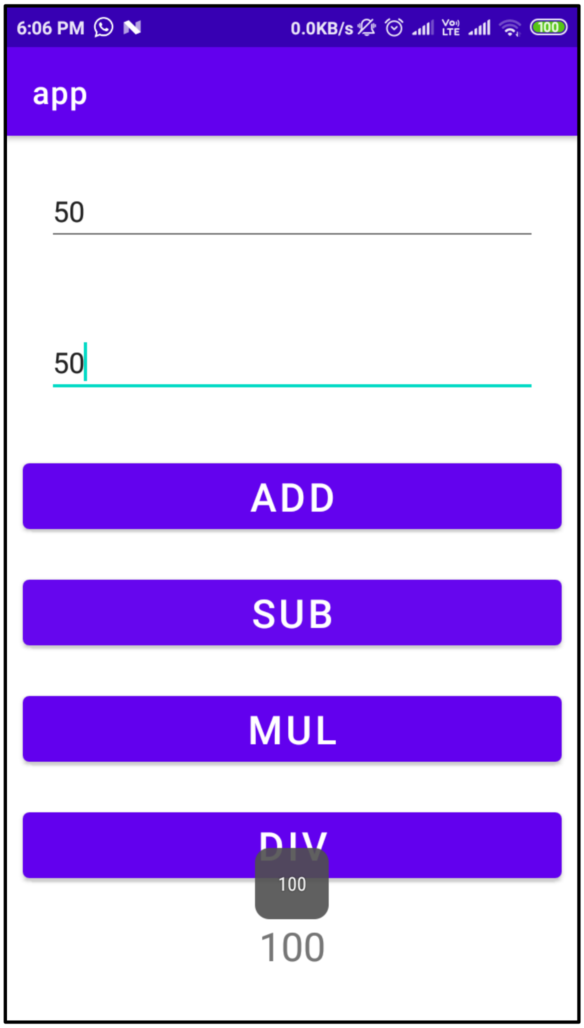
## 4 - Basic Calc (Addition and Subtraction)

### Code



```
add.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        int answer = Integer.parseInt(num1.getText().toString())+Integer.parseInt(num2.getText().toString());
        ans.setText(String.valueOf(answer));
        Toast.makeText(BasicCalc.this,String.valueOf(answer),Toast.LENGTH_LONG).show();
    }
});

sub.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        int answer = Integer.parseInt(num1.getText().toString())-Integer.parseInt(num2.getText().toString());
        ans.setText(String.valueOf(answer));
        Toast.makeText(BasicCalc.this,String.valueOf(answer),Toast.LENGTH_LONG).show();
    }
});

mul.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        int answer = Integer.parseInt(num1.getText().toString())*Integer.parseInt(num2.getText().toString());
        ans.setText(String.valueOf(answer));
        Toast.makeText(BasicCalc.this,String.valueOf(answer),Toast.LENGTH_LONG).show();
    }
});

div.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        int answer = Integer.parseInt(num1.getText().toString())/Integer.parseInt(num2.getText().toString());
        ans.setText(String.valueOf(answer));
        Toast.makeText(BasicCalc.this,String.valueOf(answer),Toast.LENGTH_LONG).show();
    }
});
```

Output

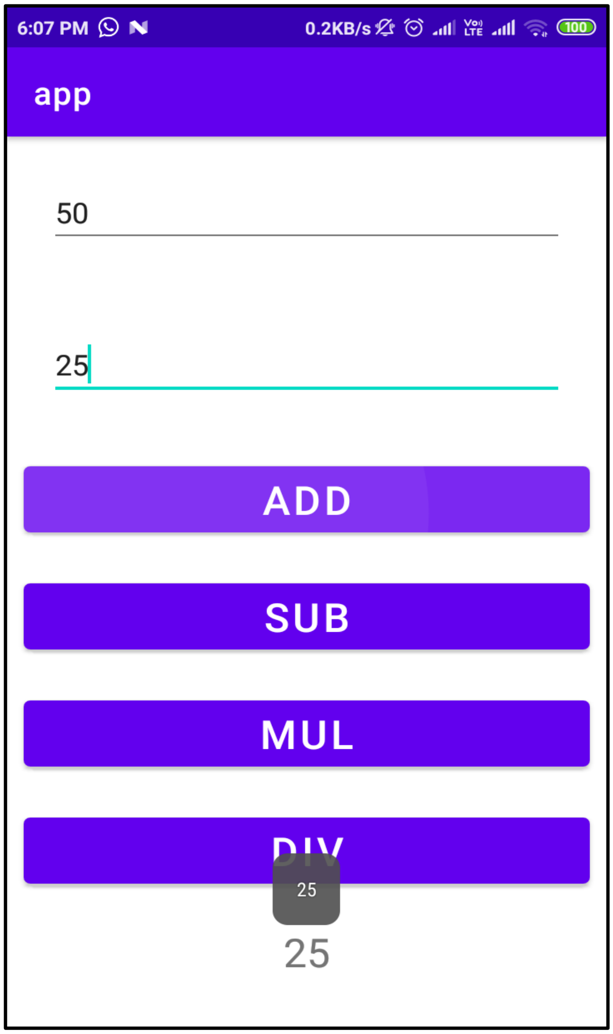


50

50

100

100

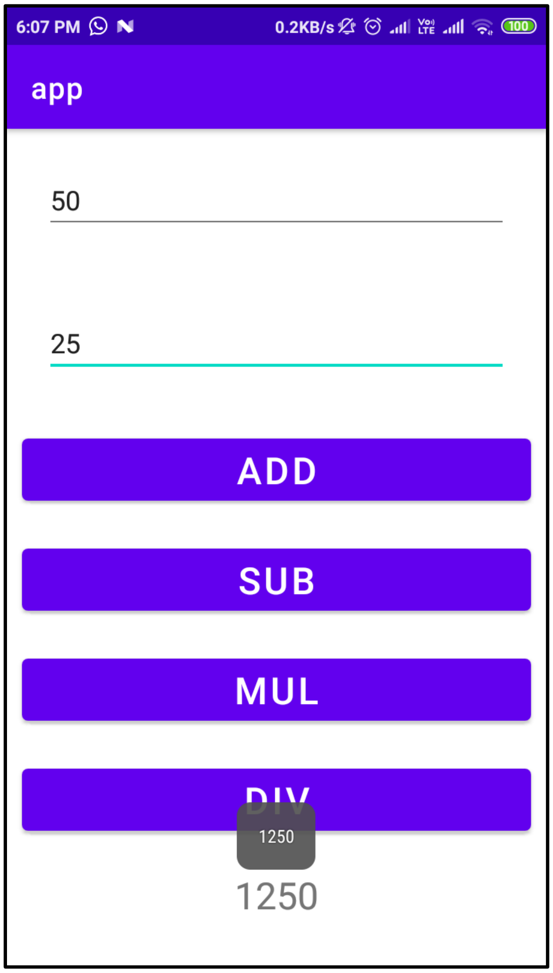


50

25

25

25

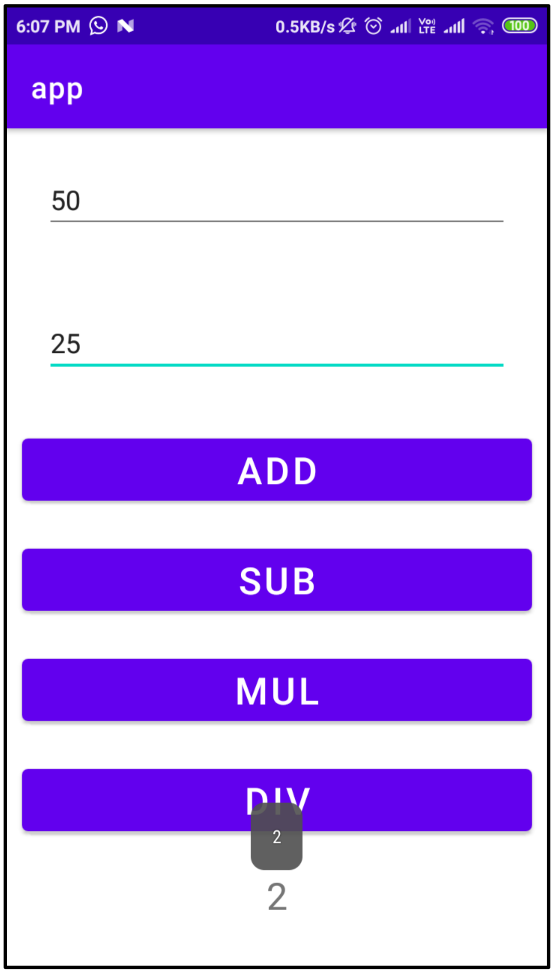


50

25

1250

1250



50

25

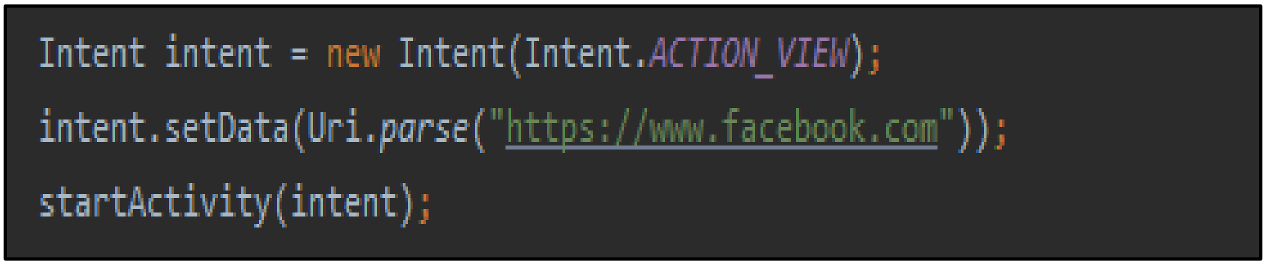
2

2

## 5 - Use of Explicit and implicit Intent

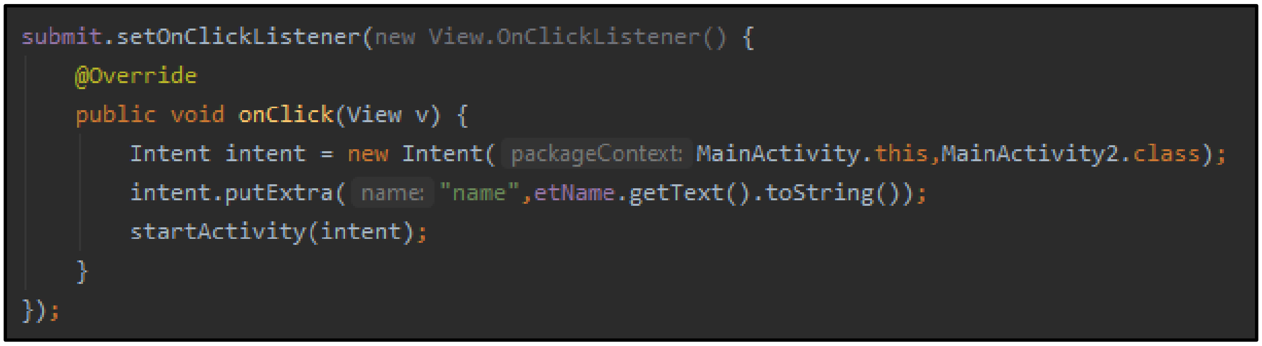
Code

ImplicitIntent.java



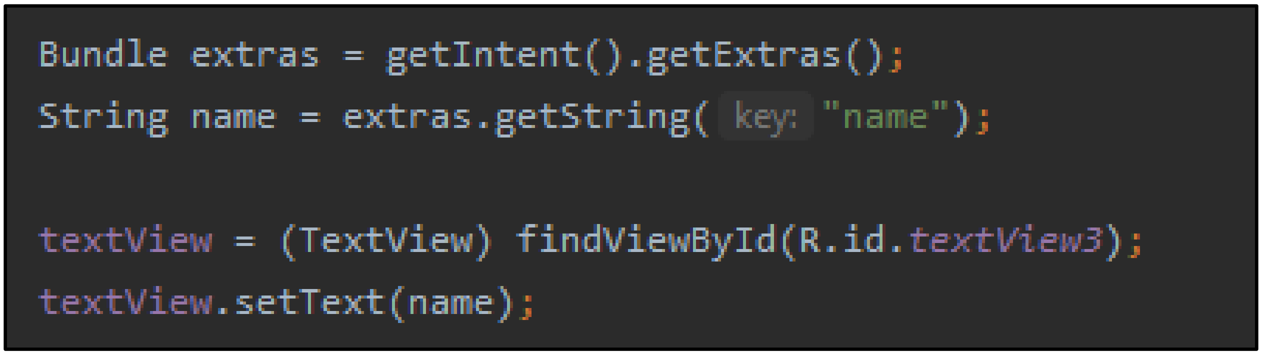
```
Intent intent = new Intent(Intent.ACTION_VIEW);
intent.setData(Uri.parse("https://www.facebook.com"));
startActivity(intent);
```

MainActivity1.java



```
submit.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent( packageContext: MainActivity.this,MainActivity2.class);
        intent.putExtra( name: "name",etName.getText().toString());
        startActivity(intent);
    }
});
```

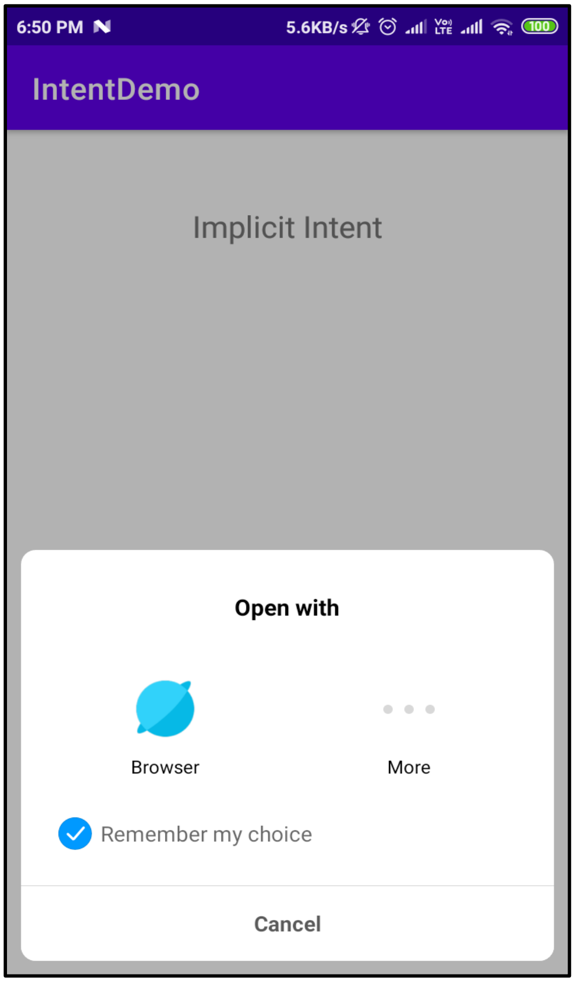
MainActivity2.java

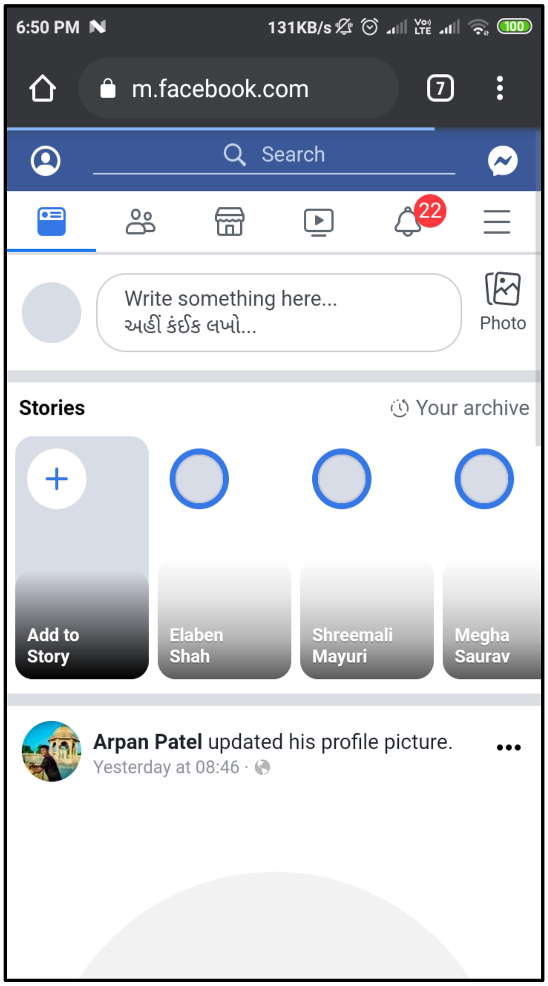


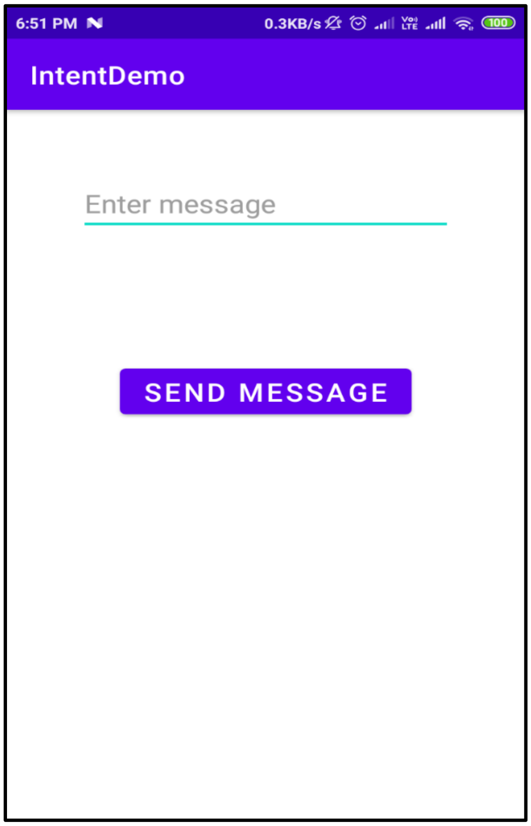
```
Bundle extras = getIntent().getExtras();
String name = extras.getString( key: "name");

textView = (TextView) findViewById(R.id.textView3);
textView.setText(name);
```

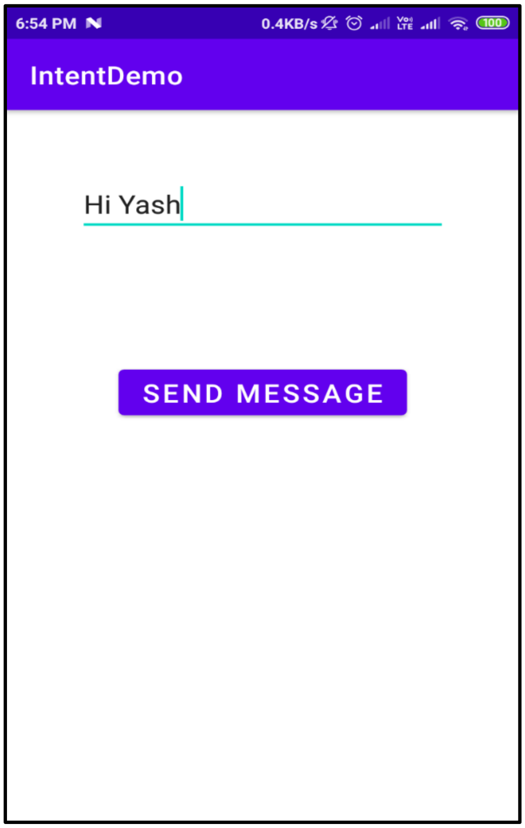
- ⇨ putExtra() method is used to put additional information in intent.
- ⇨ On the other hand, in second screen, getExtras() is used to get the extra information which are stored in MainActivity.
- ⇨ Nevertheless, after getting intent we can use getStringExtra() directly to the intent object.

Output

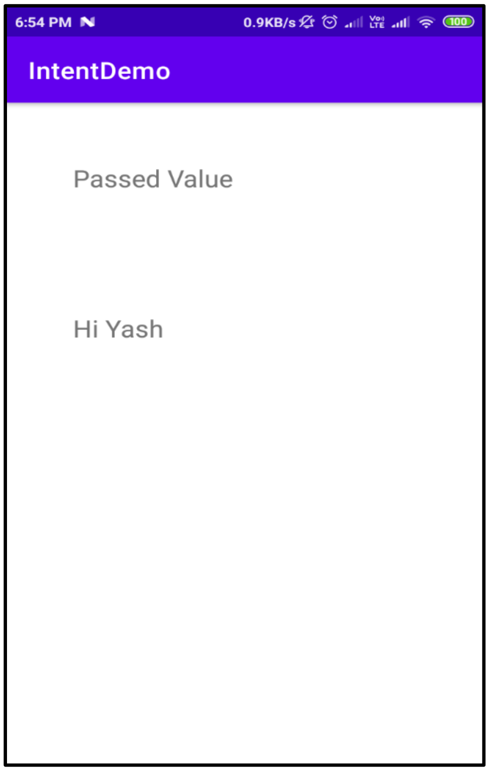

6:50 PM
5.6KB/s
100
IntentDemo
Implicit Intent
Open with
Browser
More
Remember my choice
Cancel


6:50 PM
131KB/s
100
m.facebook.com
7
Search
22
Write something here...
અહીં કંઈક લખો...
Photo
Stories
Your archive
Add to Story
Elaben Shah
Shreemali Mayuri
Megha Saurav
Arpan Patel updated his profile picture.
Yesterday at 08:46 ·





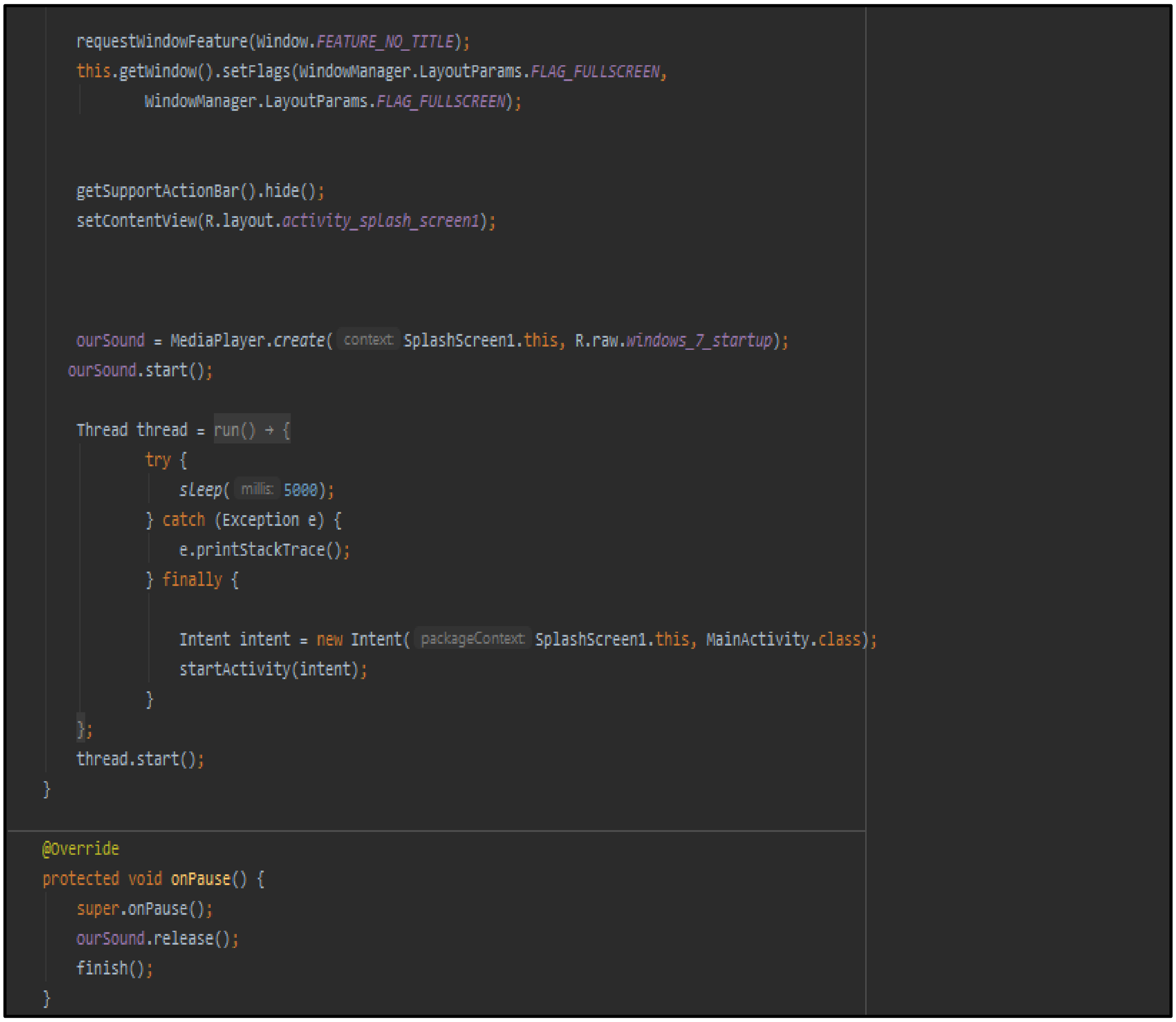
Hi Yash



```
Passed Value

Hi Yash
```

# 6 - How to make splash screen

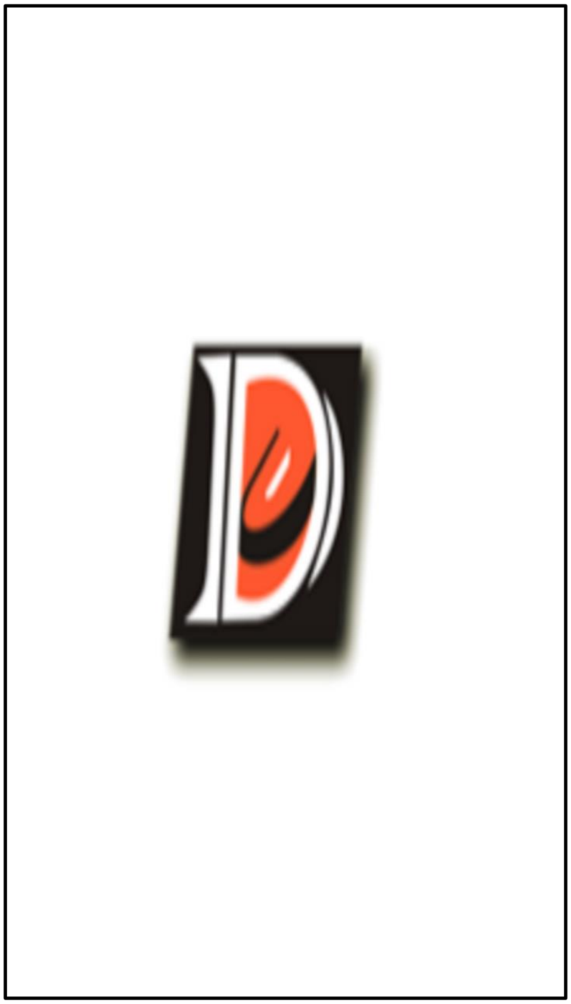
Code

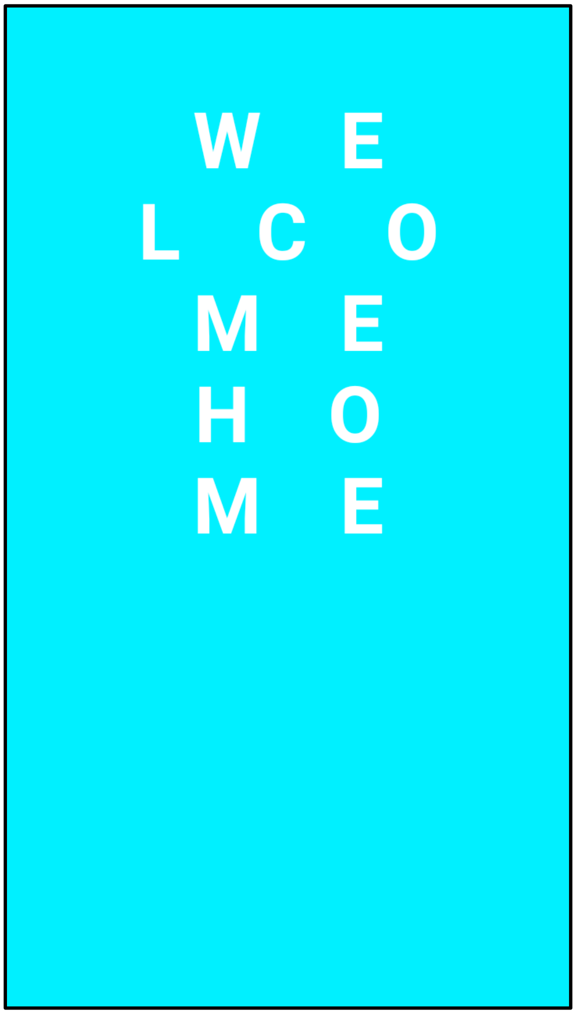


```
    requestWindowFeature(Window.FEATURE_NO_TITLE);
    this.getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
            WindowManager.LayoutParams.FLAG_FULLSCREEN);


    getSupportActionBar().hide();
    setContentView(R.layout.activity_splash_screen1);



    ourSound = MediaPlayer.create( context: SplashScreen1.this, R.raw.windows_7_startup);
   ourSound.start();

    Thread thread = run() → {
            try {
                sleep( millis: 5000);
            } catch (Exception e) {
                e.printStackTrace();
            } finally {

                Intent intent = new Intent( packageContext: SplashScreen1.this, MainActivity.class);
                startActivity(intent);
            }
    };
    thread.start();
}

@Override
protected void onPause() {
    super.onPause();
    ourSound.release();
    finish();
}
```

- ⇨ Using mediaplayer.create() to create an object
- ⇨ Object.start() = to start music
- ⇨ And in onPause method object.release() = to release the resources.
- ⇨ Also finish() is called because we don't want our splash screen to appear again and again in a same session.

Output





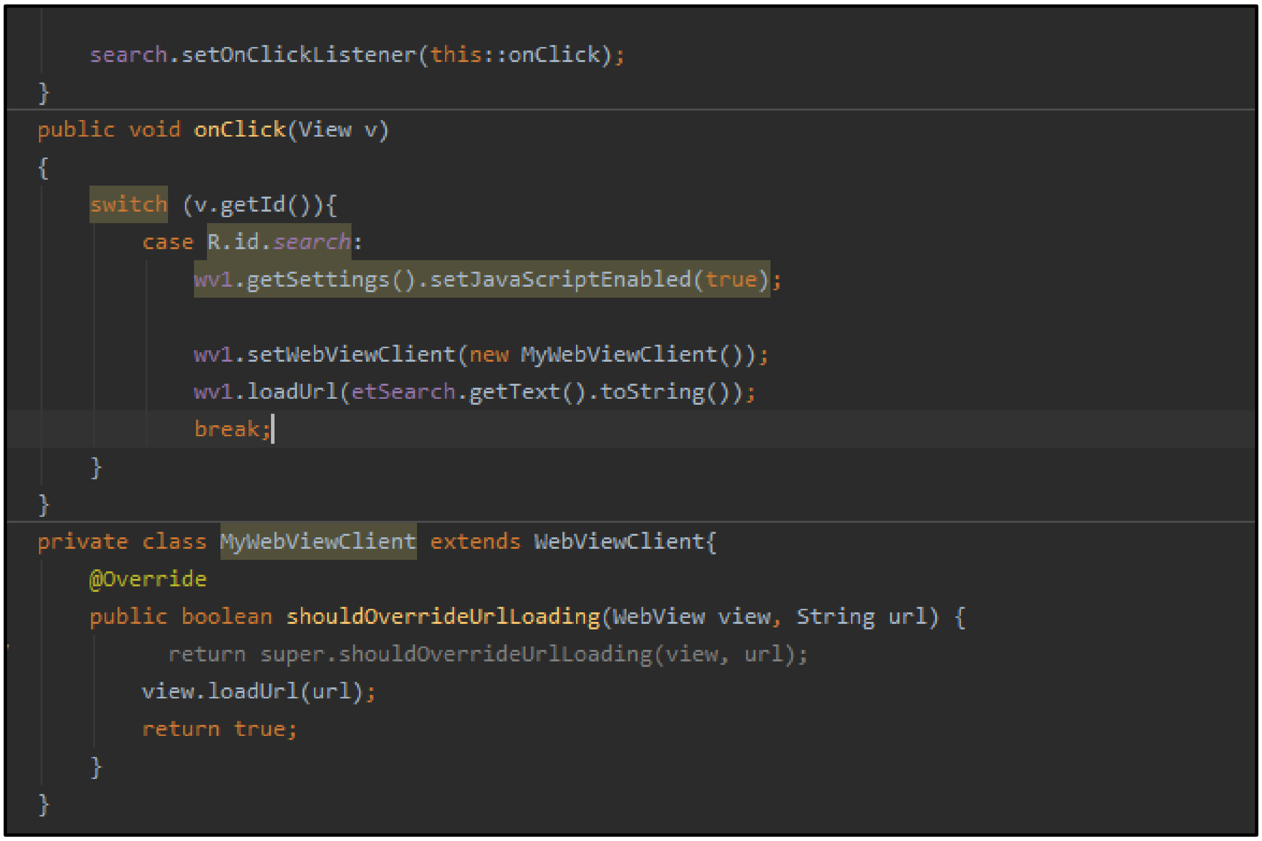
W E
L C O
M E
H O
M E

C

## 7 – Web View, List View, Grid View, Vertical and Horizontal Scrollbar

Code

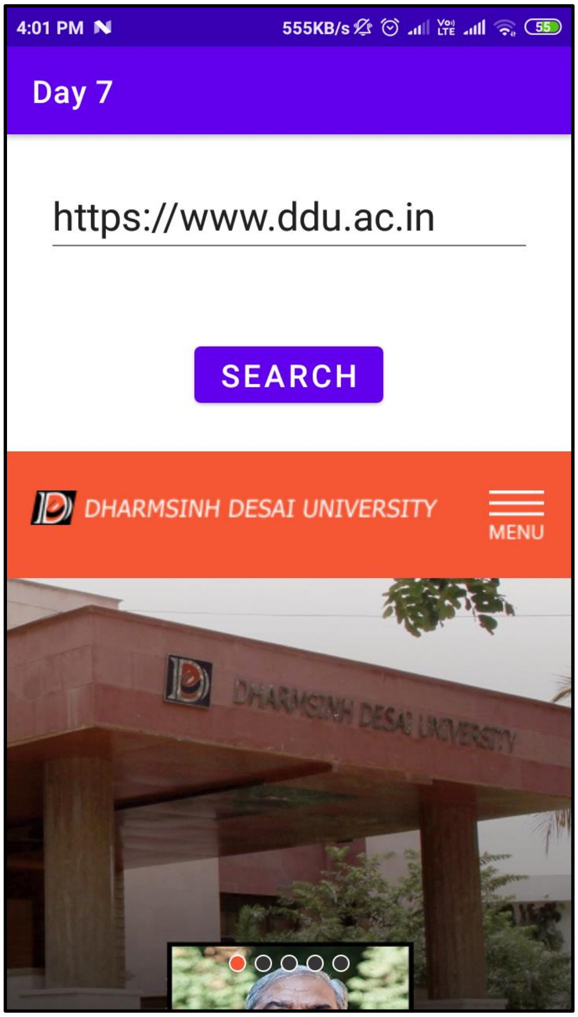
WebViewDemo.java



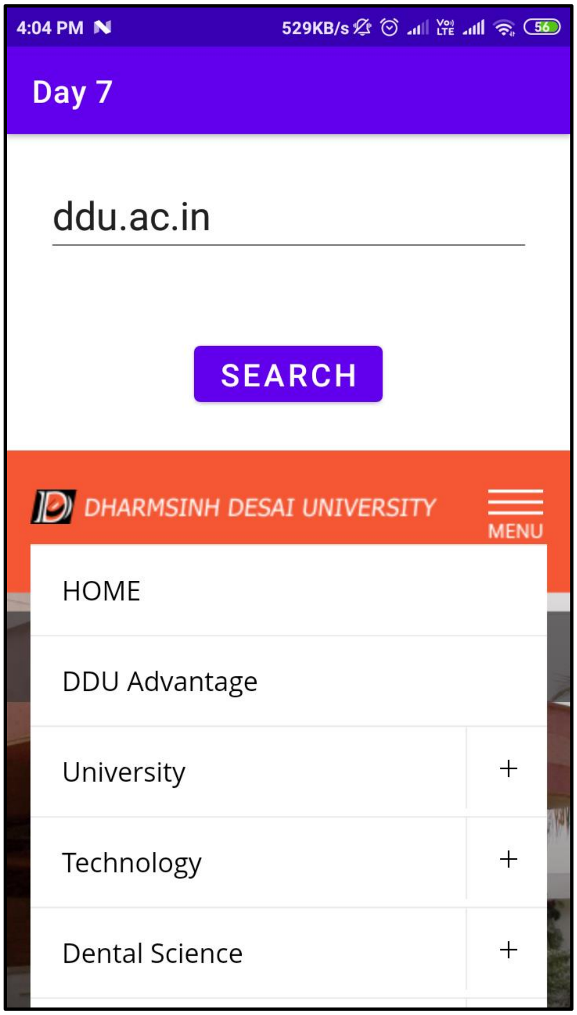
```
    search.setOnClickListener(this::onClick);
}
public void onClick(View v)
{
    switch (v.getId()){
        case R.id.search:
            wv1.getSettings().setJavaScriptEnabled(true);

            wv1.setWebViewClient(new MyWebViewClient());
            wv1.loadUrl(etSearch.getText().toString());
            break;
    }
}
private class MyWebViewClient extends WebViewClient{
    @Override
    public boolean shouldOverrideUrlLoading(WebView view, String url) {
          return super.shouldOverrideUrlLoading(view, url);
        view.loadUrl(url);
        return true;
    }
}
```

- ⇨ Creating general listener and then distinguish them using switch-case.
- ⇨ Extending WebViewClient to open all pages of website in your app not opening it in browser.
- ⇨ You have to just load url (view.loadUrl(url)) and return true;

Output

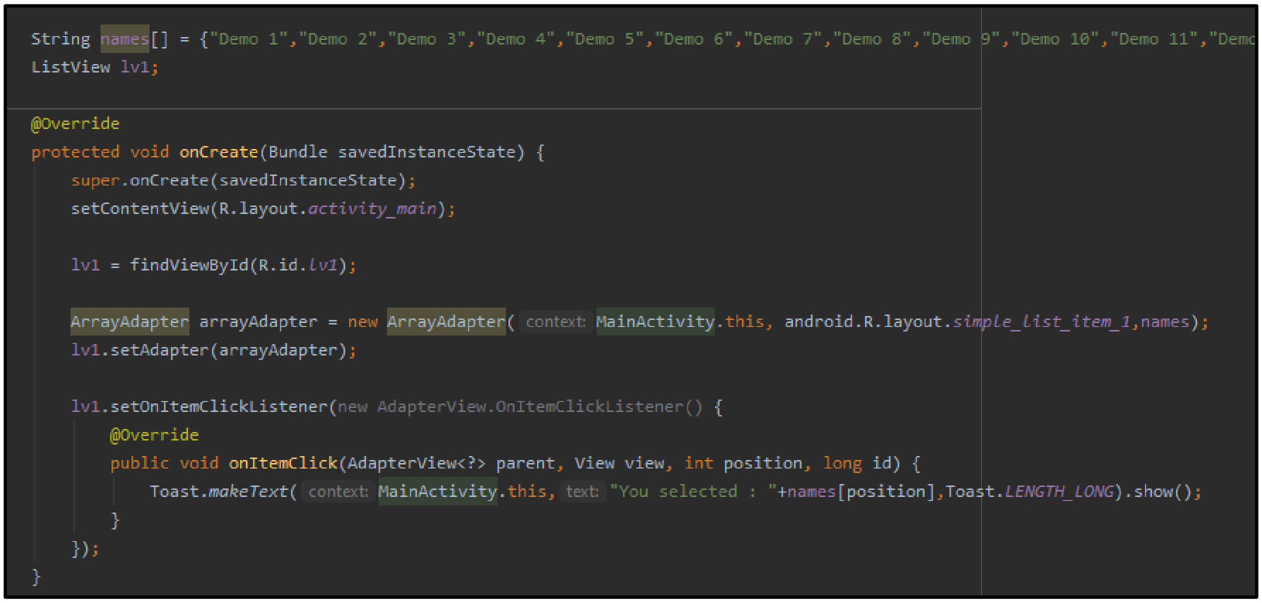


https://www.ddu.ac.in



ddu.ac.in

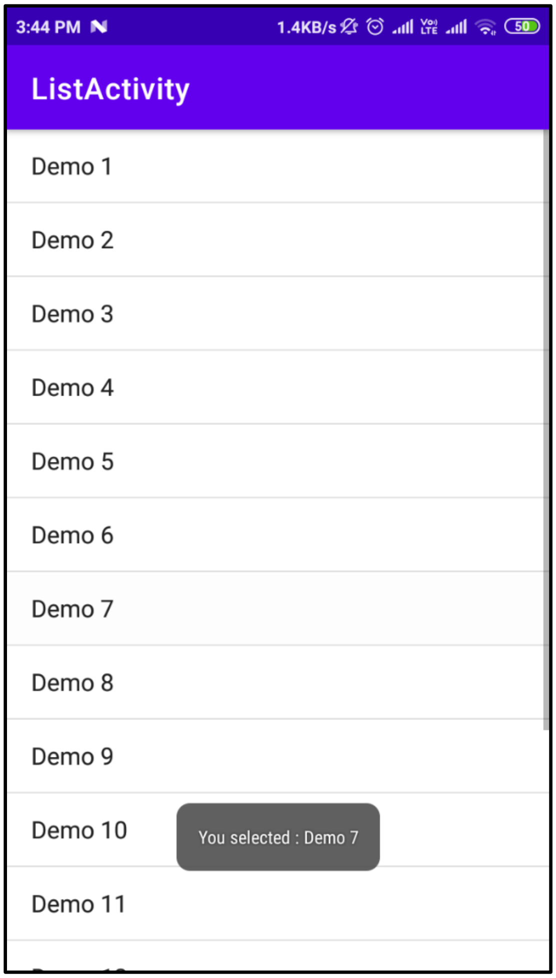
## Code



```
String names[] = {"Demo 1","Demo 2","Demo 3","Demo 4","Demo 5","Demo 6","Demo 7","Demo 8","Demo 9","Demo 10","Demo 11","Demc
ListView lv1;

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

    lv1 = findViewById(R.id.lv1);

    ArrayAdapter arrayAdapter = new ArrayAdapter( context: MainActivity.this, android.R.layout.simple_list_item_1,names);
    lv1.setAdapter(arrayAdapter);

    lv1.setOnItemClickListener(new AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
            Toast.makeText( context: MainActivity.this, text: "You selected : "+names[position],Toast.LENGTH_LONG).show();
        }
    });
}
```

- Requires the data in form of array
- Requires ArrayAdapter which works as a bridge between Screen List UI and array.
- Override setOnItemClickListener() method to handle click event on listview.

Output



ListActivity

Demo 1

Demo 2

Demo 3

Demo 4

Demo 5

Demo 6

Demo 7

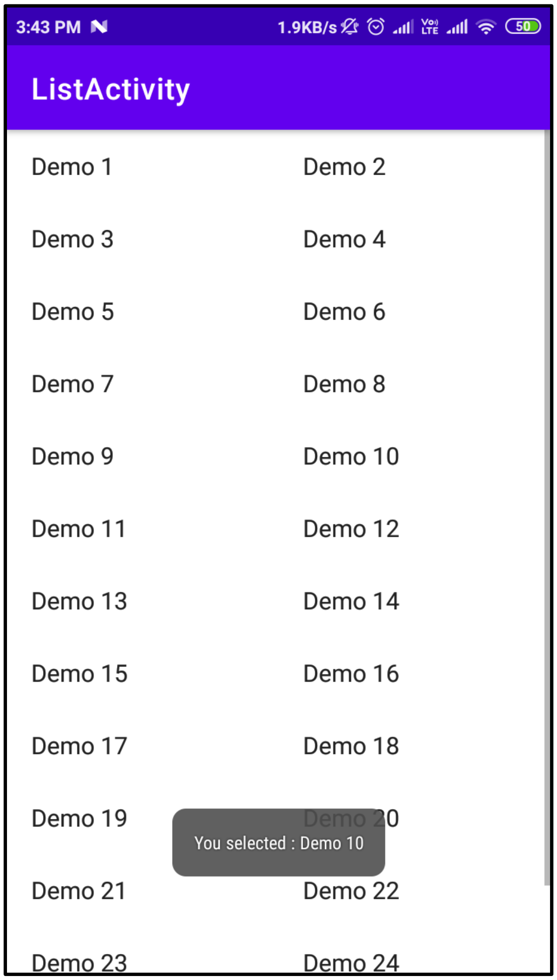
Demo 8

Demo 9

Demo 10

Demo 11

You selected : Demo 7



ListActivity

Demo 1 Demo 2

Demo 3 Demo 4

Demo 5 Demo 6

Demo 7 Demo 8

Demo 9 Demo 10

Demo 11 Demo 12

Demo 13 Demo 14

Demo 15 Demo 16

Demo 17 Demo 18

Demo 19 Demo 20

Demo 21 Demo 22

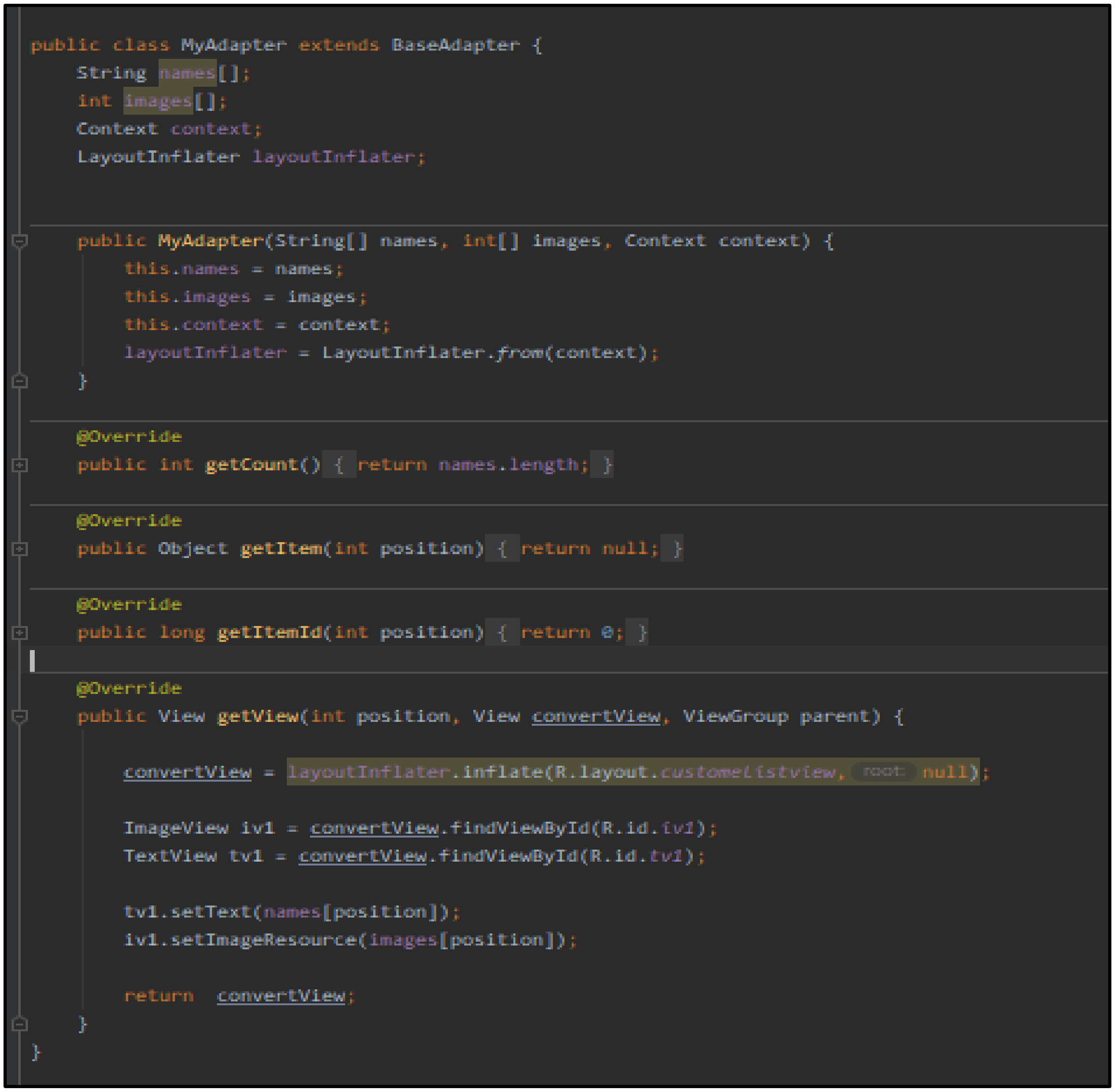
Demo 23 Demo 24

You selected : Demo 10

## 8 - Custom List View and Grid View

Code

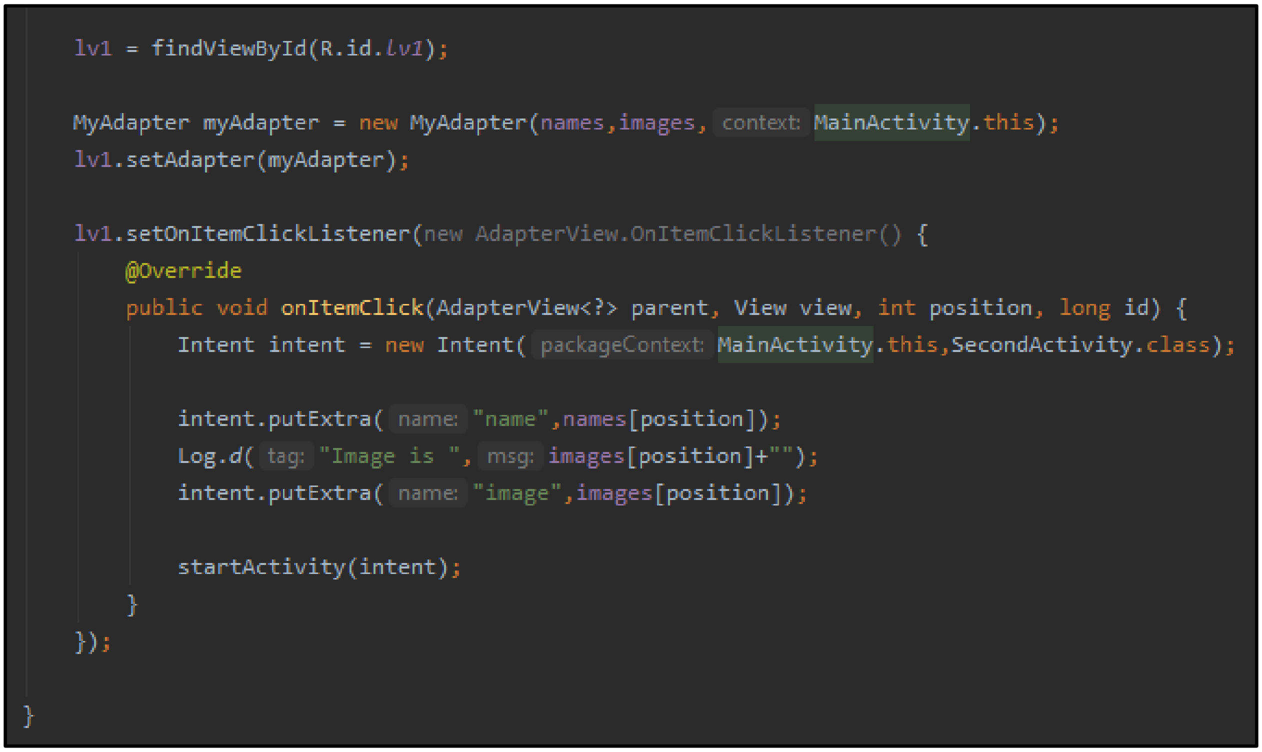
MyAdapter.java



```java
public class MyAdapter extends BaseAdapter {
    String names[];
    int images[];
    Context context;
    LayoutInflater layoutInflater;


    public MyAdapter(String[] names, int[] images, Context context) {
        this.names = names;
        this.images = images;
        this.context = context;
        layoutInflater = LayoutInflater.from(context);
    }

    @Override
    public int getCount() { return names.length; }

    @Override
    public Object getItem(int position) { return null; }

    @Override
    public long getItemId(int position) { return 0; }

    @Override
    public View getView(int position, View convertView, ViewGroup parent) {

        convertView = layoutInflater.inflate(R.layout.customelistview, root: null);

        ImageView iv1 = convertView.findViewById(R.id.iv1);
        TextView tv1 = convertView.findViewById(R.id.tv1);

        tv1.setText(names[position]);
        iv1.setImageResource(images[position]);

        return  convertView;
    }
}
```

- ⇨ For custom listview you need to create separate class which extends BaseAdapter class and override its abstract methods.
- ⇨ getView in which you have to return a view that makes a single item.
- ⇨ Mainly override getCount() and getView() method.
- ⇨ You can get access to layout UI using LayoutInflator.

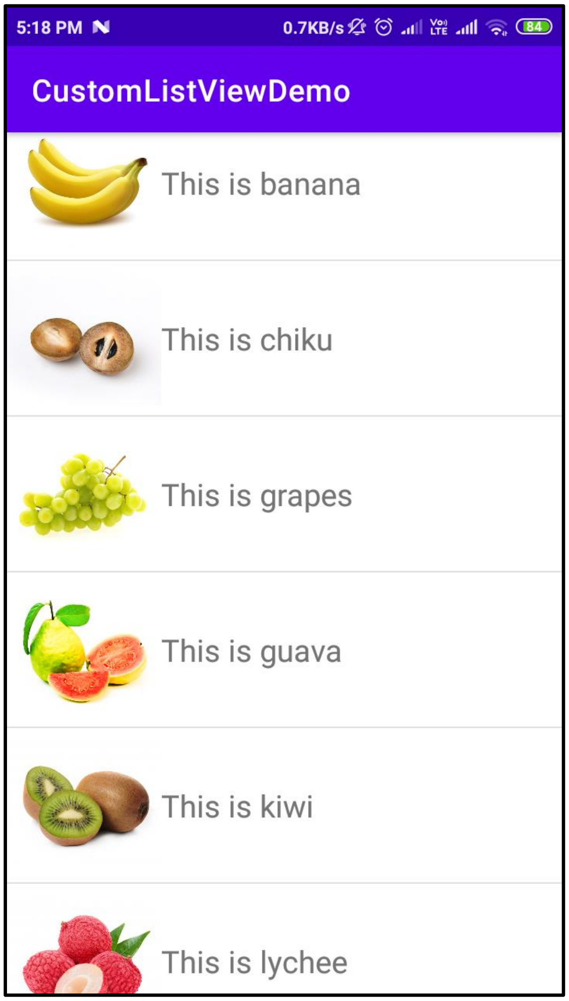
MainActivity.java



```
    lv1 = findViewById(R.id.lv1);

    MyAdapter myAdapter = new MyAdapter(names,images, context: MainActivity.this);
    lv1.setAdapter(myAdapter);

    lv1.setOnItemClickListener(new AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
            Intent intent = new Intent( packageContext: MainActivity.this,SecondActivity.class);

            intent.putExtra( name: "name",names[position]);
            Log.d( tag: "Image is ", msg: images[position]+"");
            intent.putExtra( name: "image",images[position]);

            startActivity(intent);
        }
    });

}
```

- Create an object of MyAdapter
- Pass string array names and int array images and context.
- Set the adapter (lv1.setAdapter(myAdapter))
- Finally, implement setOnItemClickListener().
- In which create an intent and put extra information like- name and image id form array using position.

Output



CustomListViewDemo

This is banana

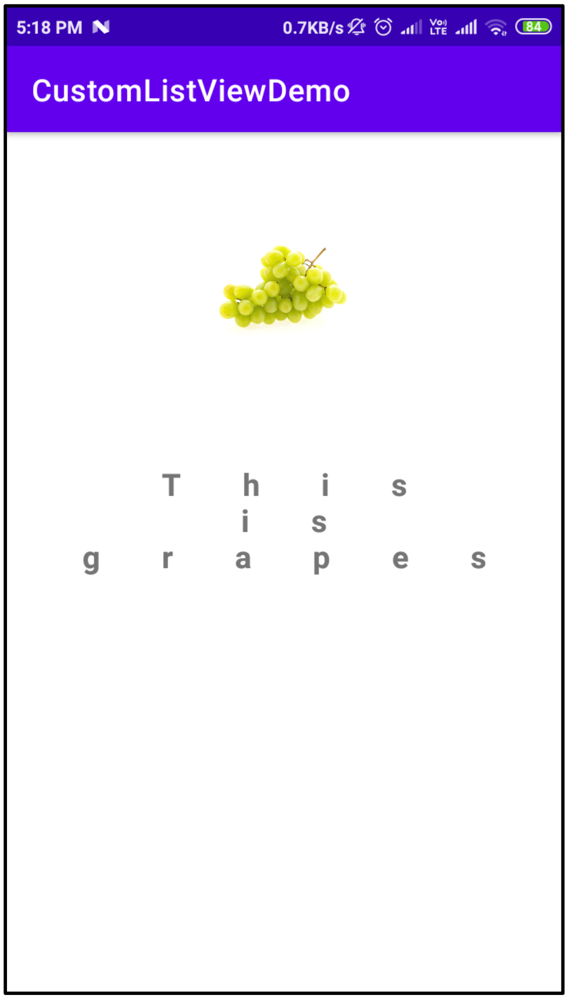
This is chiku

This is grapes

This is guava

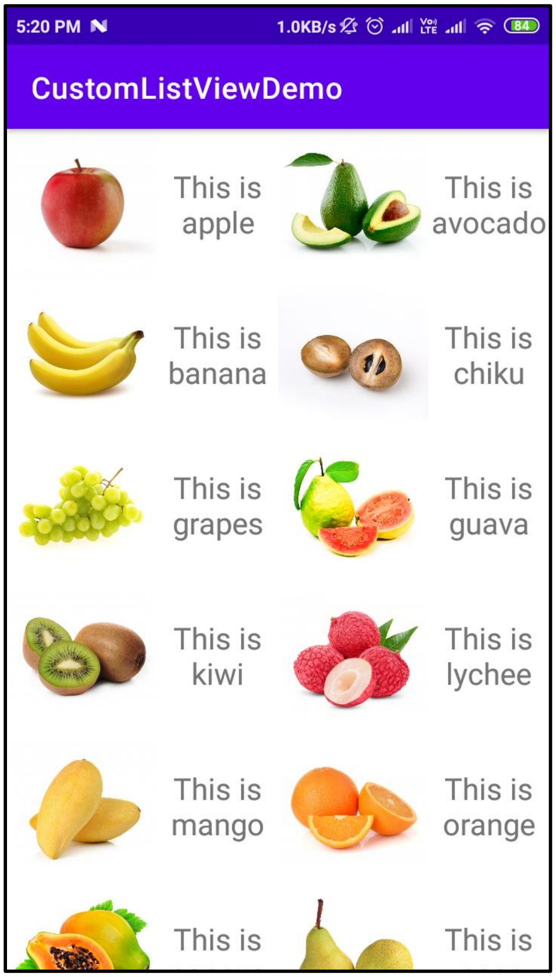
This is kiwi

This is lychee



CustomListViewDemo

```
    T   h   i   s
        i   s
g   r   a   p   e   s
```



This is apple

This is avocado

This is banana

This is chiku

This is grapes

This is guava

This is kiwi

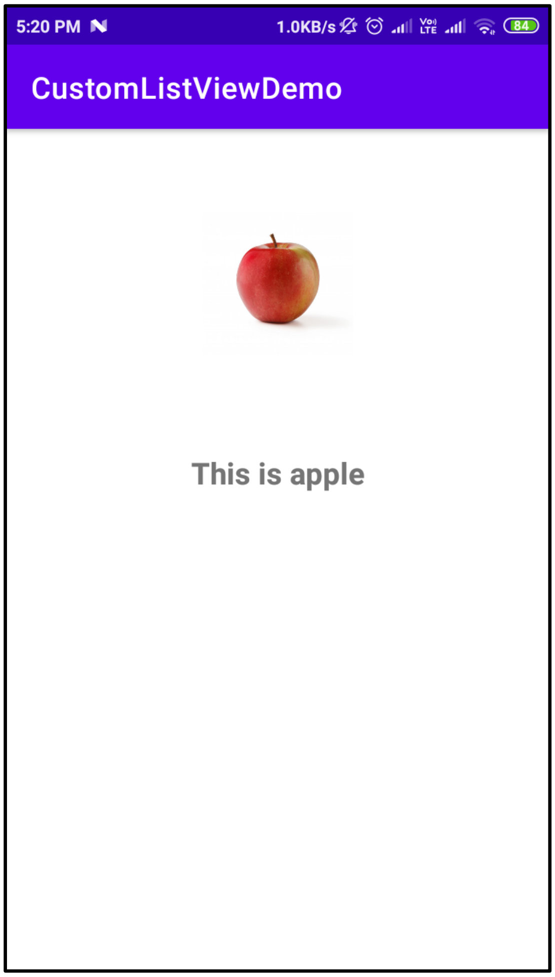
This is lychee

This is mango

This is orange

This is

This is

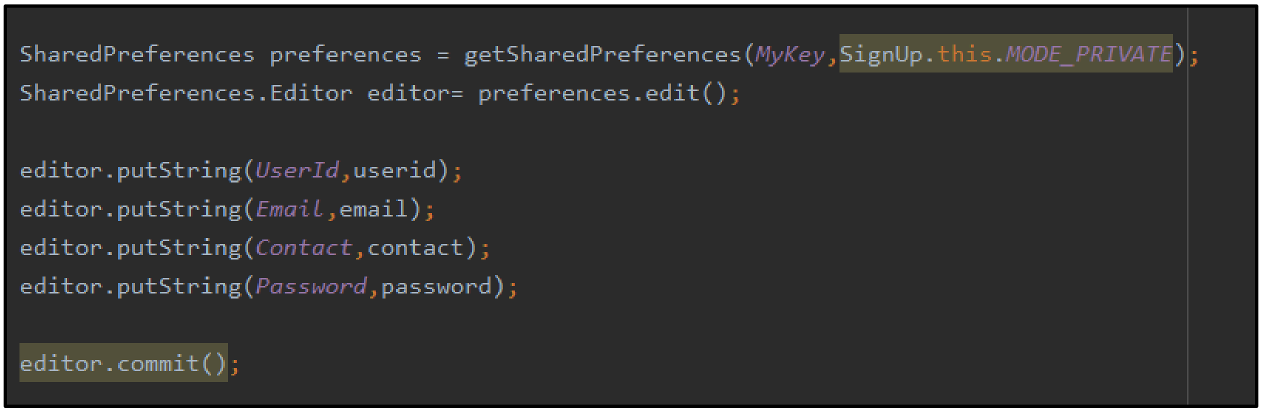


This is apple

## 9 - SharedPreferences - getSharedPreferences and Editor

Code

SignUp.java

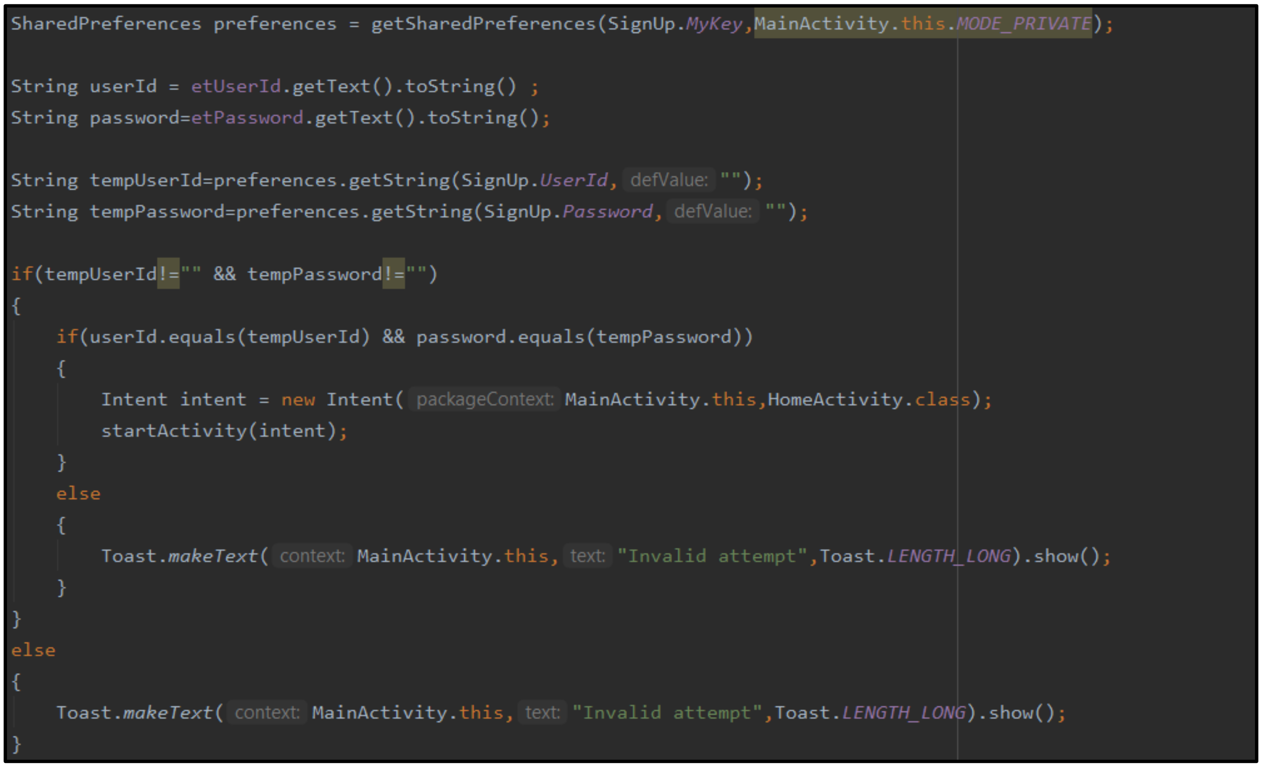


```
SharedPreferences preferences = getSharedPreferences(MyKey,SignUp.this.MODE_PRIVATE);
SharedPreferences.Editor editor= preferences.edit();

editor.putString(UserId,userid);
editor.putString(Email,email);
editor.putString(Contact,contact);
editor.putString(Password,password);

editor.commit();
```

- First grab the SharedPreference object using getSharedPreferences passing the argument key name and mode.
- Then if you want to add or delete values from it you need to use editor which is sub class of sharedpreference.
- SharedPreferences.Editor editor = preference.edit() = edit() method will give you an object of Editor class
- Then using putString method you can put various values in it.
- Finally, you have to call editor.commit() method to save changes.

Login.java

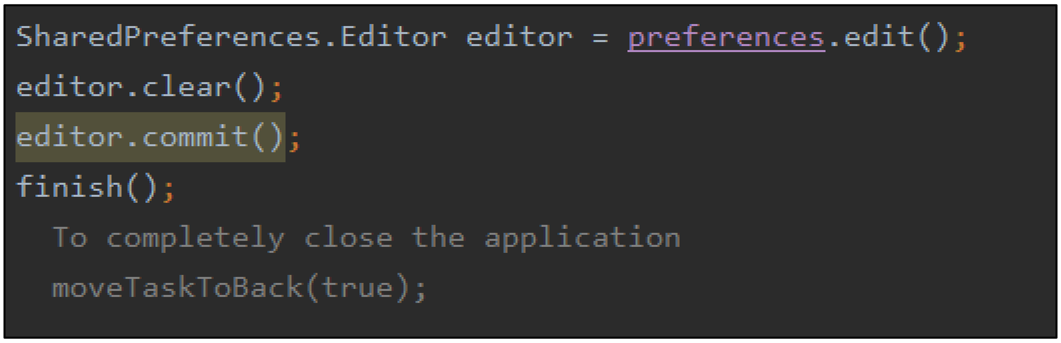
- Here also you need to grab the object first.
- But here you do not need Editor class's object unless you want to add anything
- For reading values from SharedPreferences only object of SharedPreference is sufficient.
- For that you need to use getString(key,defValue) method of SharedPreference.



```
SharedPreferences preferences = getSharedPreferences(SignUp.MyKey,MainActivity.this.MODE_PRIVATE);

String userId = etUserId.getText().toString() ;
String password=etPassword.getText().toString();

String tempUserId=preferences.getString(SignUp.UserId, defValue: "");
String tempPassword=preferences.getString(SignUp.Password, defValue: "");

if(tempUserId!="" && tempPassword!="")
{
    if(userId.equals(tempUserId) && password.equals(tempPassword))
    {
        Intent intent = new Intent( packageContext: MainActivity.this,HomeActivity.class);
        startActivity(intent);
    }
    else
    {
        Toast.makeText( context: MainActivity.this, text: "Invalid attempt",Toast.LENGTH_LONG).show();
    }
}
else
{
    Toast.makeText( context: MainActivity.this, text: "Invalid attempt",Toast.LENGTH_LONG).show();
}
```

HomeScreen.java

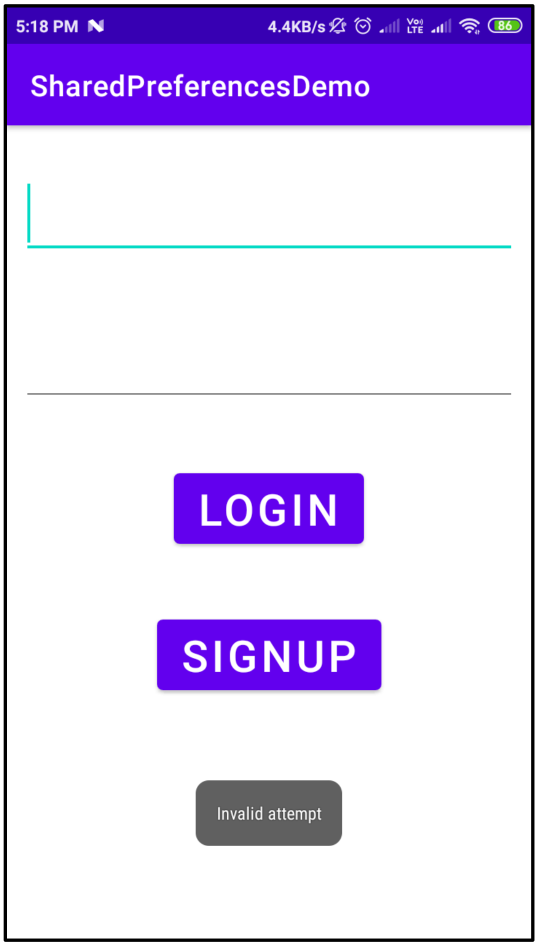
Logout Functionality



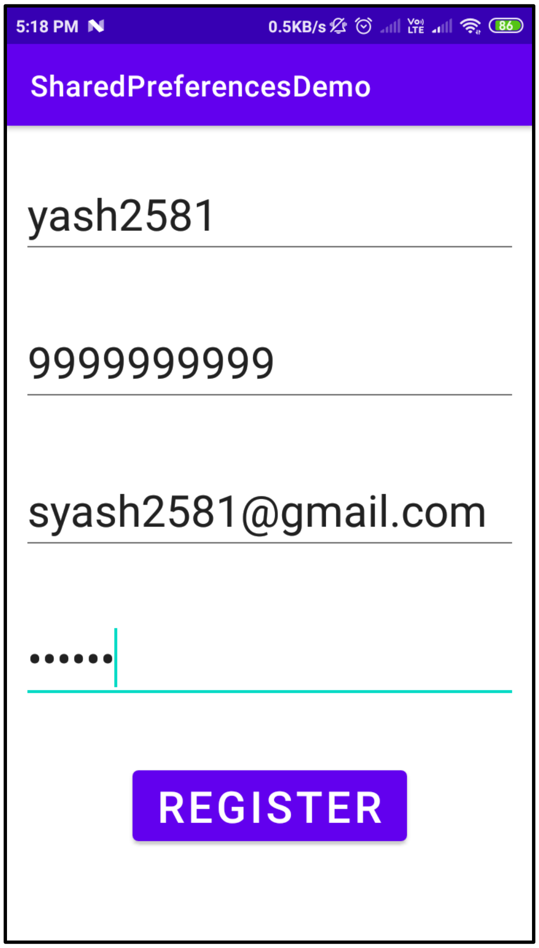
```
SharedPreferences.Editor editor = preferences.edit();
editor.clear();
editor.commit();
finish();
  To completely close the application
  moveTaskToBack(true);
```

- ⇨ Again, if you want to remove sharedPreference. You need to use Editor’s object.
- ⇨ Simply calling editor.clear() will clear the values but don’t forget to call **commit()** method otherwise **impact will not be done**.
- ⇨ To remove entire application you may use moveTaskToBack(true).

Output



Invalid attempt

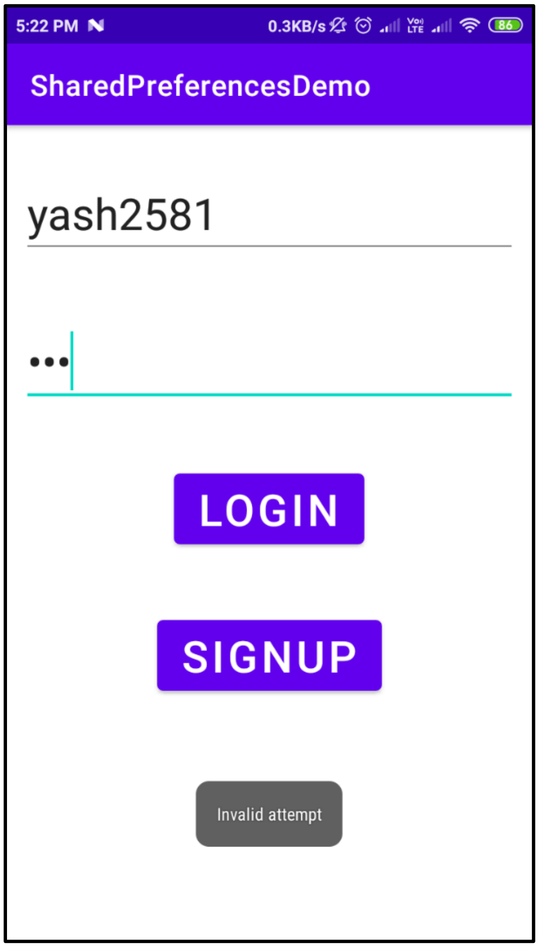


yash2581

9999999999

syash2581@gmail.com

••••••

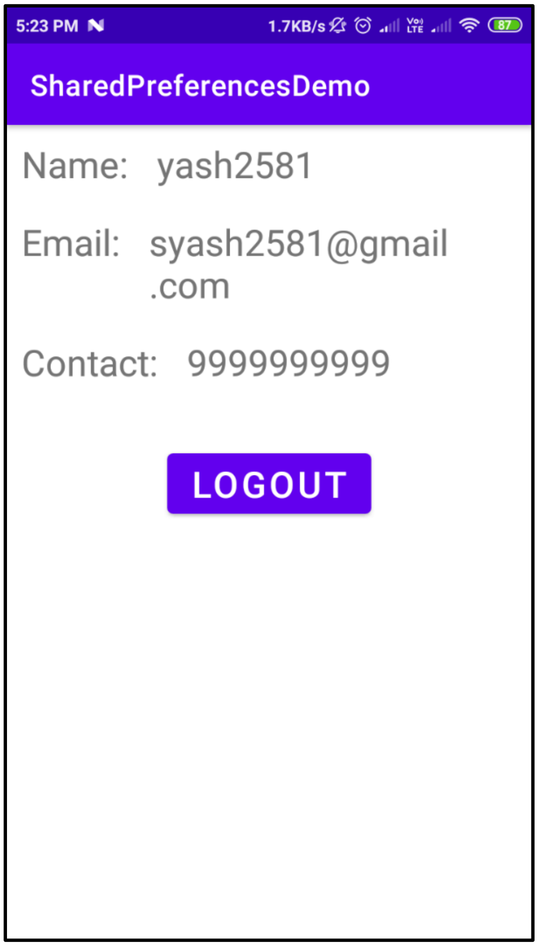


SharedPreferencesDemo

yash2581

•••

Invalid attempt



SharedPreferencesDemo

Name: yash2581

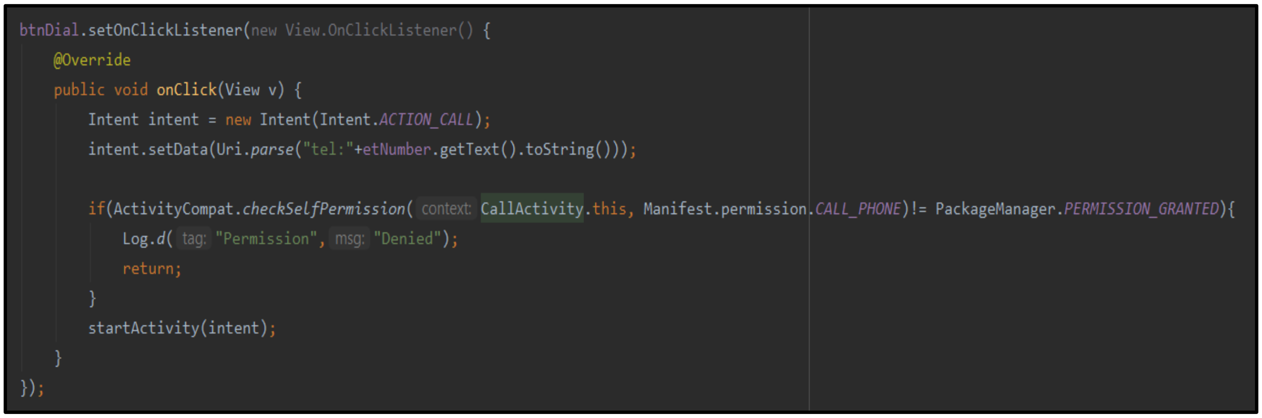
Email: syash2581@gmail.com

Contact: 9999999999

## 10 - Making a call, sending a mail, sending a sms, playing an audio using MediaPlayer, playing a video using VideoView and MediaController

Code

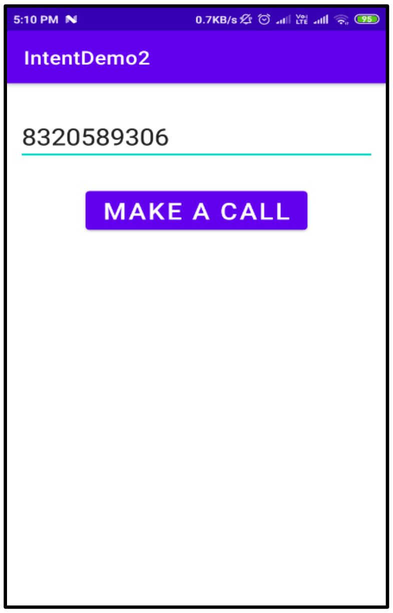
CallActivity.java



```
btnDial.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(Intent.ACTION_CALL);
        intent.setData(Uri.parse("tel:"+etNumber.getText().toString()));

        if(ActivityCompat.checkSelfPermission( context: CallActivity.this, Manifest.permission.CALL_PHONE)!= PackageManager.PERMISSION_GRANTED){
            Log.d( tag: "Permission", msg: "Denied");
            return;
        }
        startActivity(intent);
    }
});
```

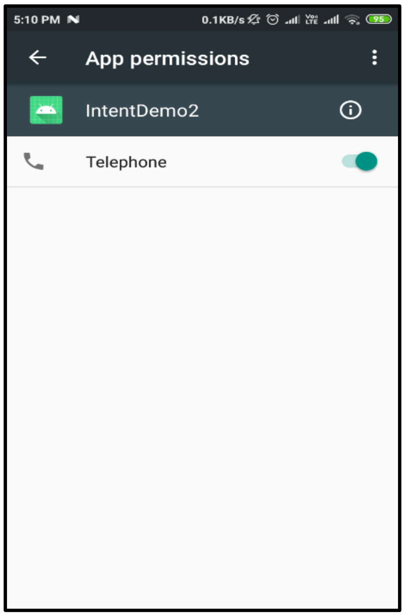
- ⇨ Create an intent of type ***ACTION_CALL*** then set tel:+number as data in setData method.
- ⇨ Check for telephone permission
- ⇨ If granted then start the activity else show the message.

Output



IntentDemo2

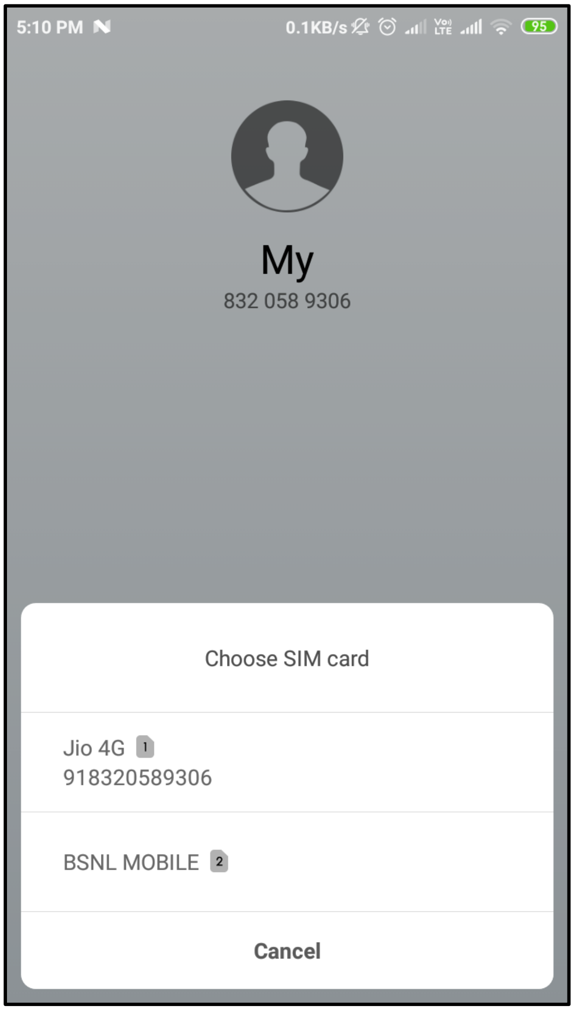
8320589306



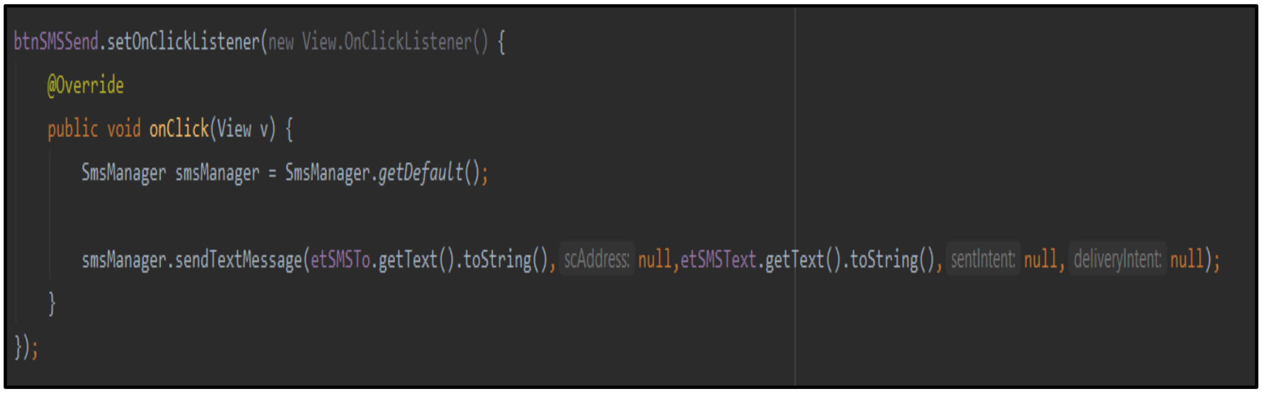
App permissions

IntentDemo2

Telephone ●


5:10 PM
0.1KB/s
95
My
832 058 9306
Choose SIM card
Jio 4G 1
918320589306
BSNL MOBILE 2
Cancel

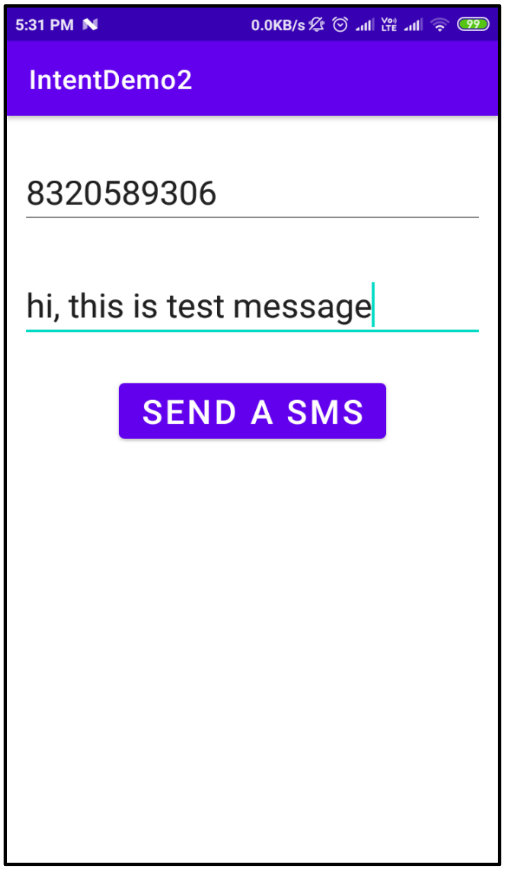
## Code



```
btnSMSSend.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        SmsManager smsManager = SmsManager.getDefault();

        smsManager.sendTextMessage(etSMSTo.getText().toString(), scAddress: null,etSMSText.getText().toString(), sentIntent: null, deliveryIntent: null);
    }
});
```

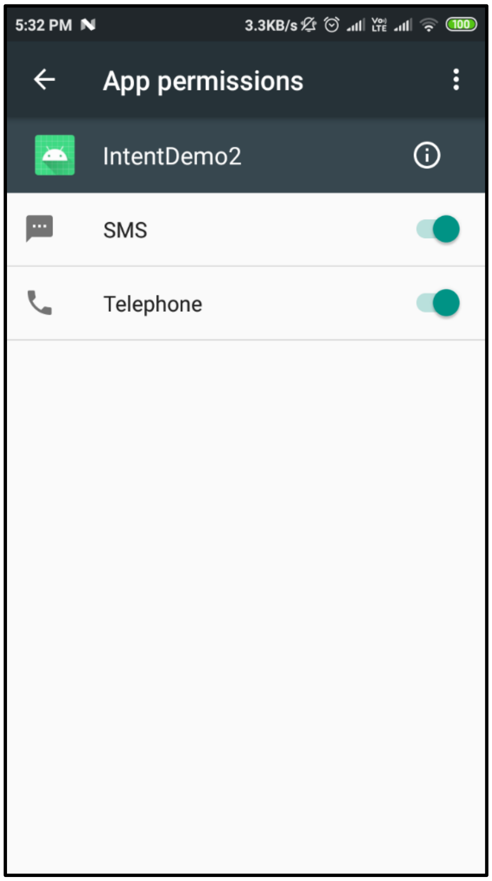
- ⇨ Using smsManager API to send a sms.
- ⇨ Get the object and use sendTextMessage and it's done.

## Output



8320589306

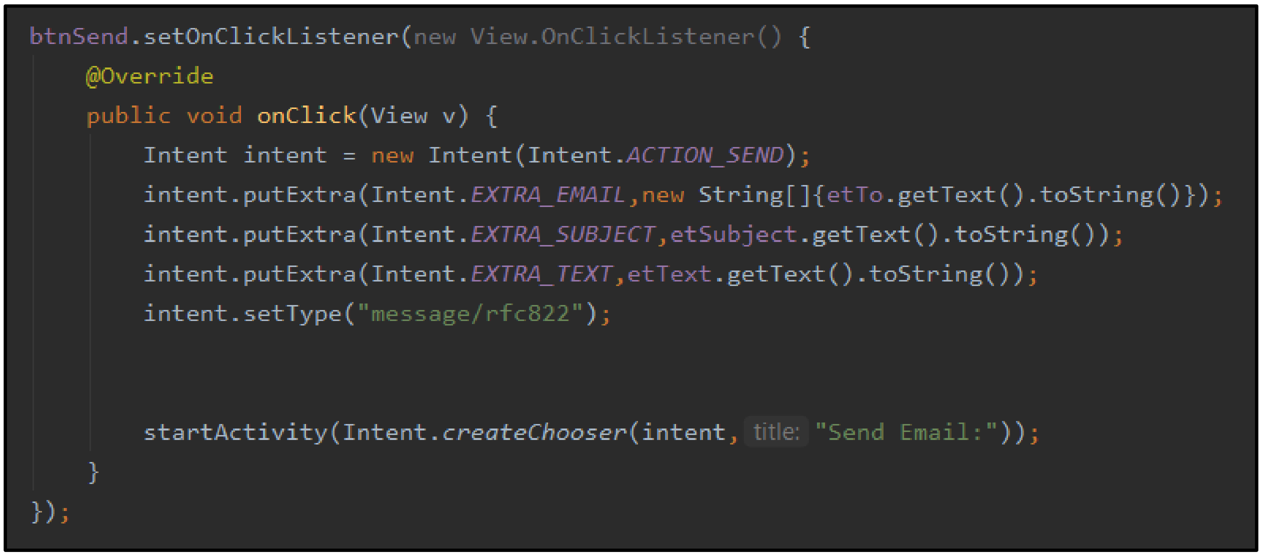
hi, this is test message



SMS ●

Telephone ●

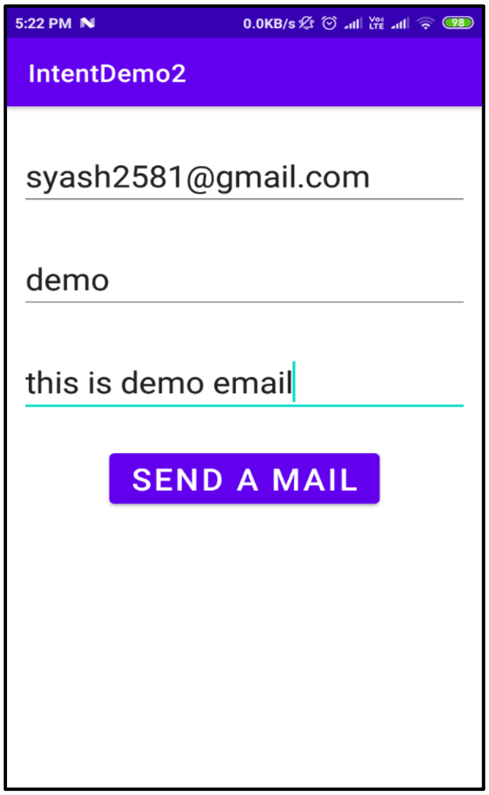
Code



```
btnSend.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(Intent.ACTION_SEND);
        intent.putExtra(Intent.EXTRA_EMAIL,new String[]{etTo.getText().toString()});
        intent.putExtra(Intent.EXTRA_SUBJECT,etSubject.getText().toString());
        intent.putExtra(Intent.EXTRA_TEXT,etText.getText().toString());
        intent.setType("message/rfc822");


        startActivity(Intent.createChooser(intent, title: "Send Email:"));
    }
});
```

- ⇨ Use Intent.ACTION_SEND
- ⇨ Then add basic four fields
  - ○ Intent.EXTRA_EMAIL, new String[]{array values}
  - ○ Intent.EXTRA_SUBJECT, add subject as string
  - ○ Intent.EXTRA_TEXT, add message as string
  - ○ Specifying email type => setType(“message/rfc822”);
- ⇨ Start the Activity and you can use Intent.createChooser(intent,”Send Email:”)) method to give options to user to send email like- gmail , phone’s inbuilt main app.

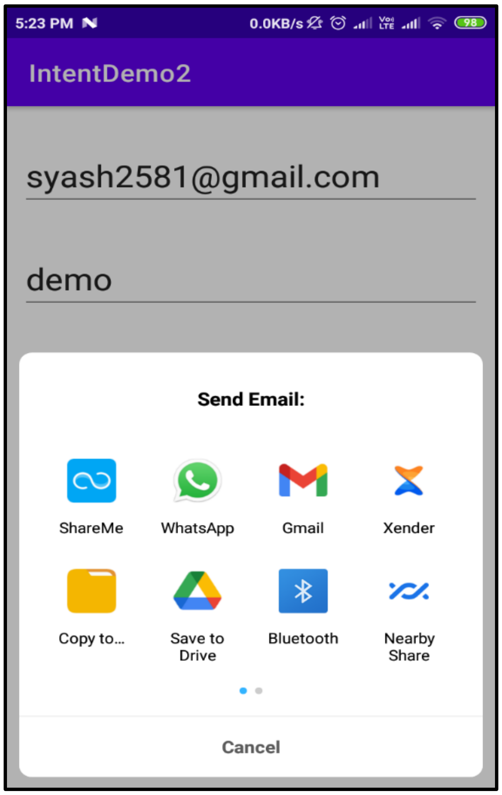
## Output



syash2581@gmail.com

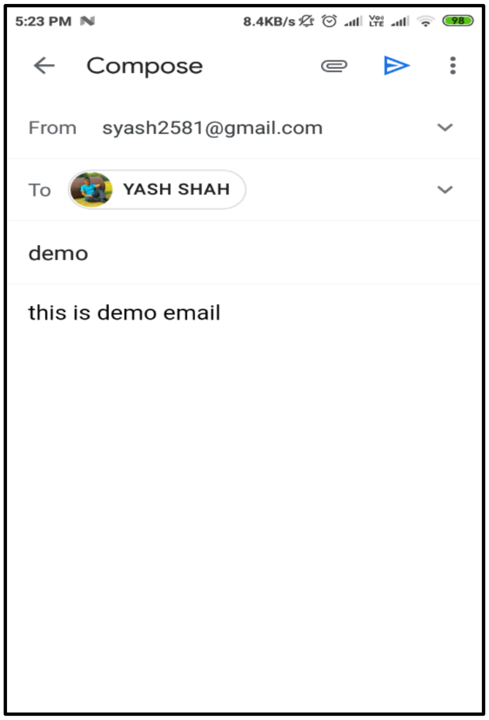
demo

this is demo email



syash2581@gmail.com

demo



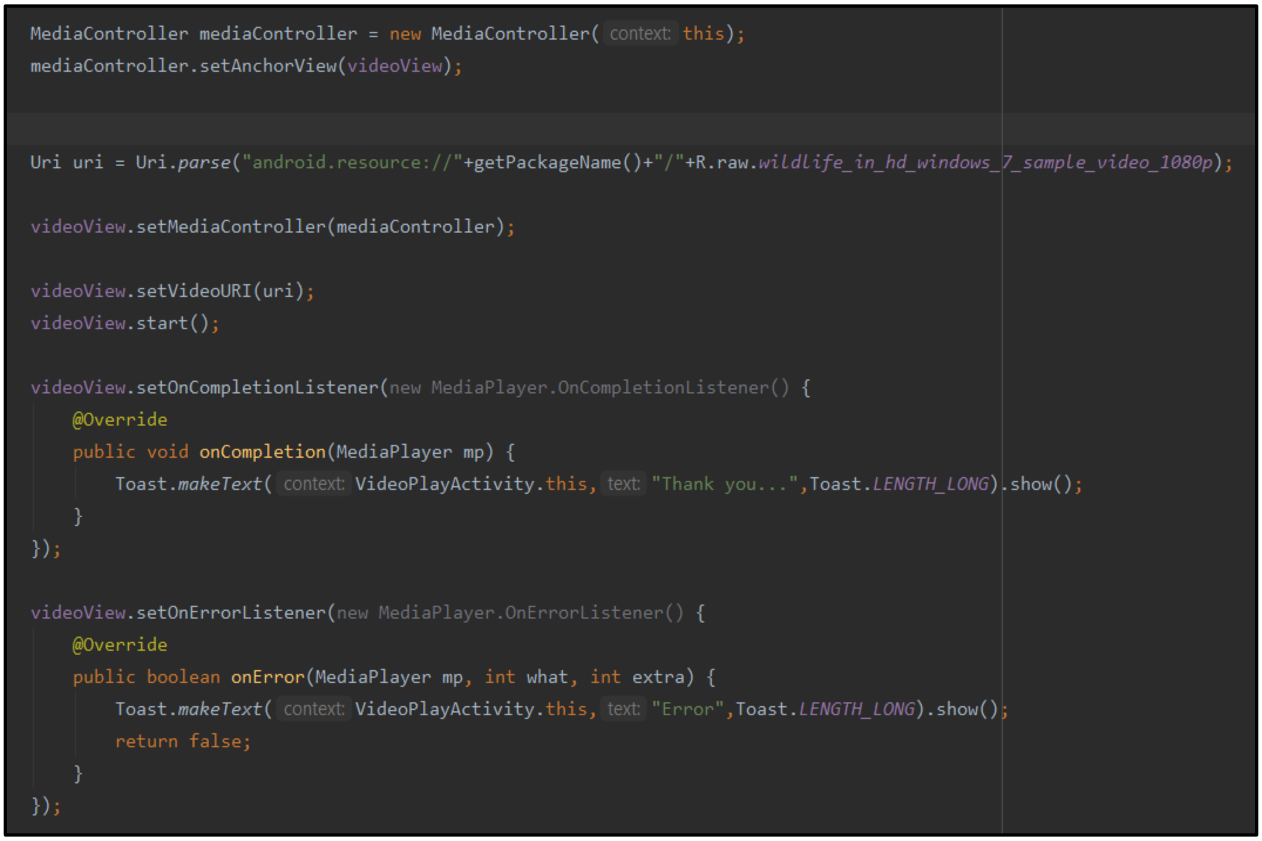
From syash2581@gmail.com

To YASH SHAH

demo

this is demo email

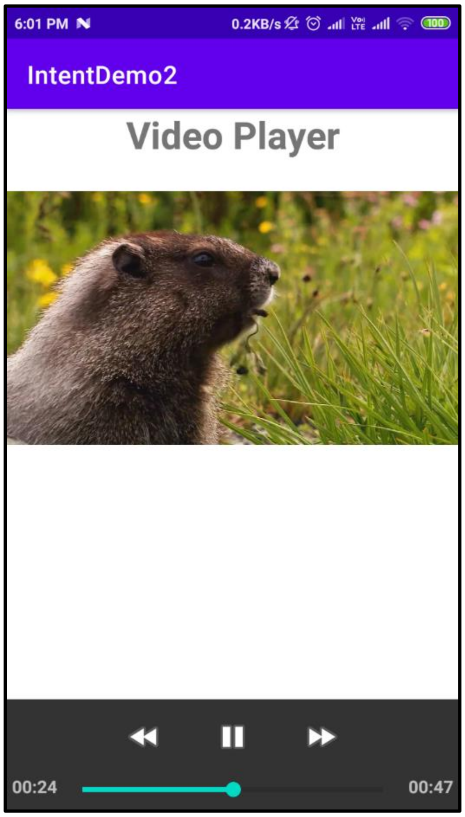
Code



```
MediaController mediaController = new MediaController( context: this);
mediaController.setAnchorView(videoView);


Uri uri = Uri.parse("android.resource://"+getPackageName()+"/"+R.raw.wildlife_in_hd_windows_7_sample_video_1080p);

videoView.setMediaController(mediaController);

videoView.setVideoURI(uri);
videoView.start();

videoView.setOnCompletionListener(new MediaPlayer.OnCompletionListener() {
    @Override
    public void onCompletion(MediaPlayer mp) {
        Toast.makeText( context: VideoPlayActivity.this, text: "Thank you...",Toast.LENGTH_LONG).show();
    }
});

videoView.setOnErrorListener(new MediaPlayer.OnErrorListener() {
    @Override
    public boolean onError(MediaPlayer mp, int what, int extra) {
        Toast.makeText( context: VideoPlayActivity.this, text: "Error",Toast.LENGTH_LONG).show();
        return false;
    }
});
```

- ⇨ Create mediaController for video
- ⇨ And use setAnchorView(videoView)
- ⇨ Use uri to create argument for videoView
- ⇨ Use setVideoURI(uri)
- ⇨ VIdeoView.start()
- ⇨ Add listeners setOnCompletionListener and setOnErrorListener.

Output


6:01 PM

0.2KB/s

100

IntentDemo2

Video Player

00:24

00:47

## 11 – Android Application Project

- ⇨ Developed Bhakti Application using Flutter.

- ⇨ Nowadays, everyone wants that there morning should be energetic and for that they pray god and for that they require bhajans.

- ⇨ Now if they use youtube then they have to turn the screen on while playing the song and it cause battery drain.

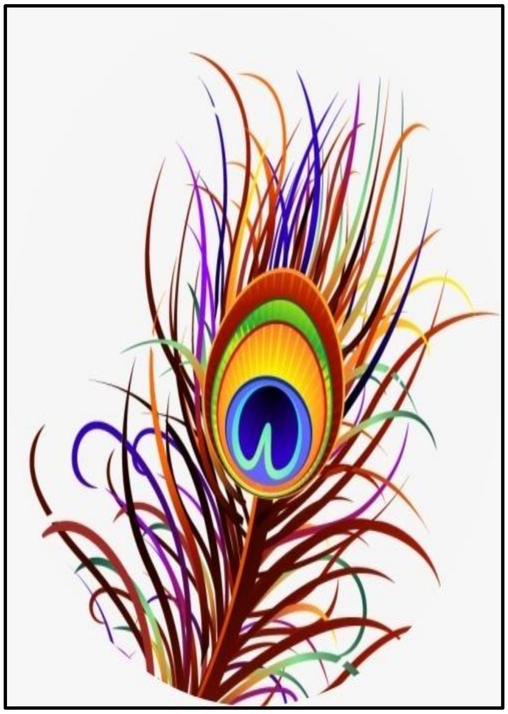
- ⇨ Secondly, it they download apps like gaana , savan then they have to deal with advertisement coming between songs.

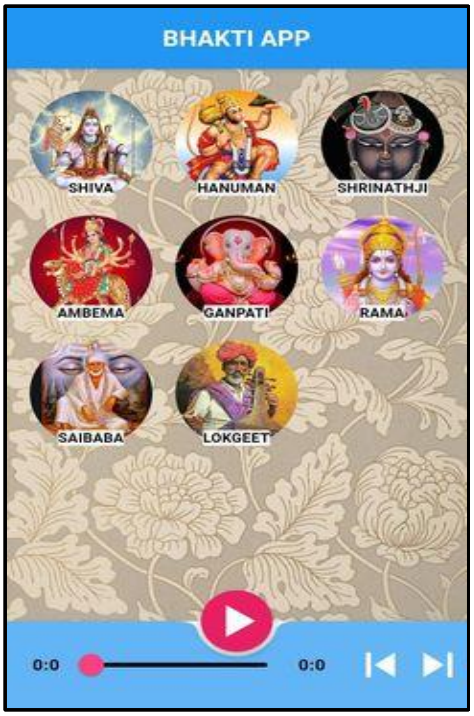
- ⇨ And finally, both ways require high speed internet.

- ⇨ My application has resolved all this issues just providing offline songs through application not only that but it does not use internet any way.

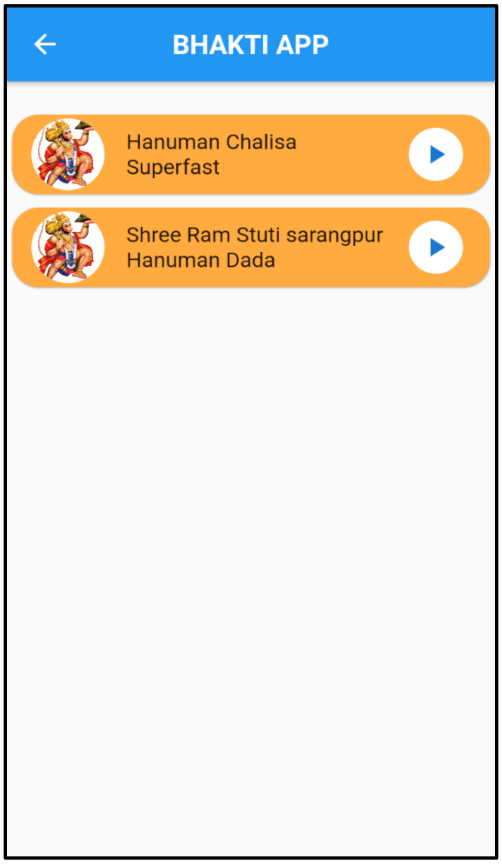
- ⇨ Link: https://www.tinyurl.com/BhaktiApplication

- ⇨ Snapshots





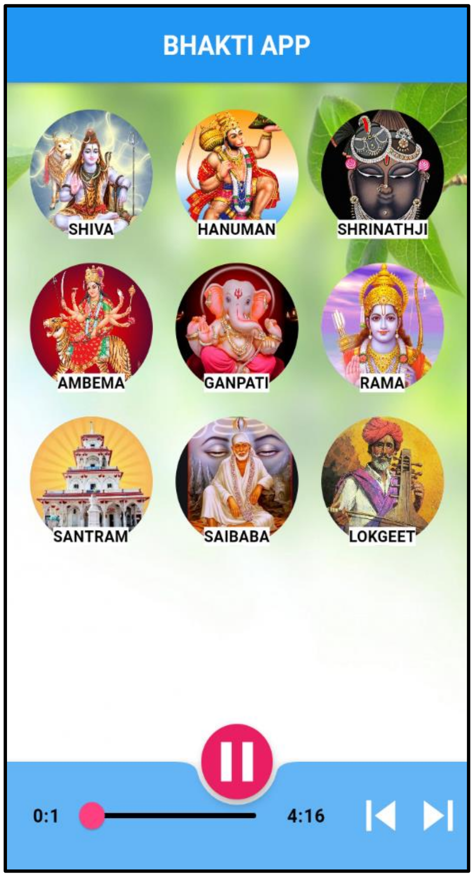
BHAKTI APP



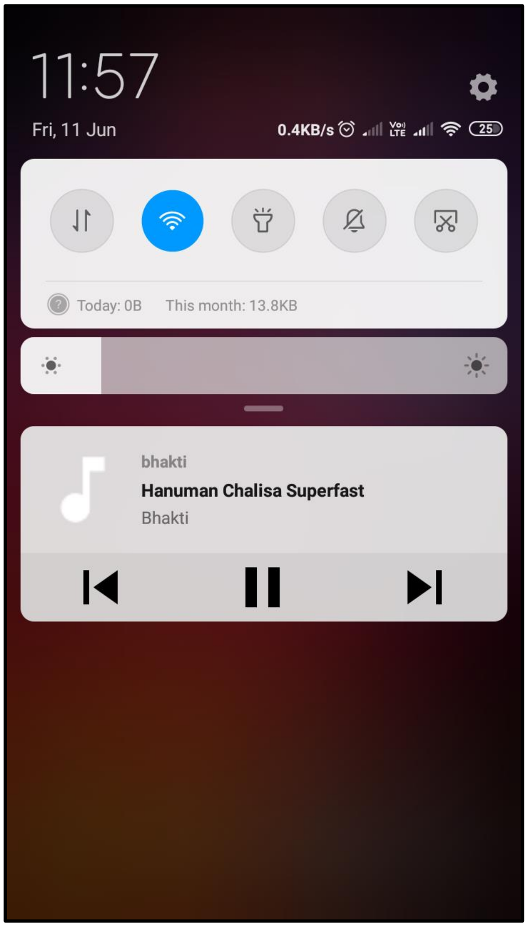
BHAKTI APP

Hanuman Chalisa Superfast

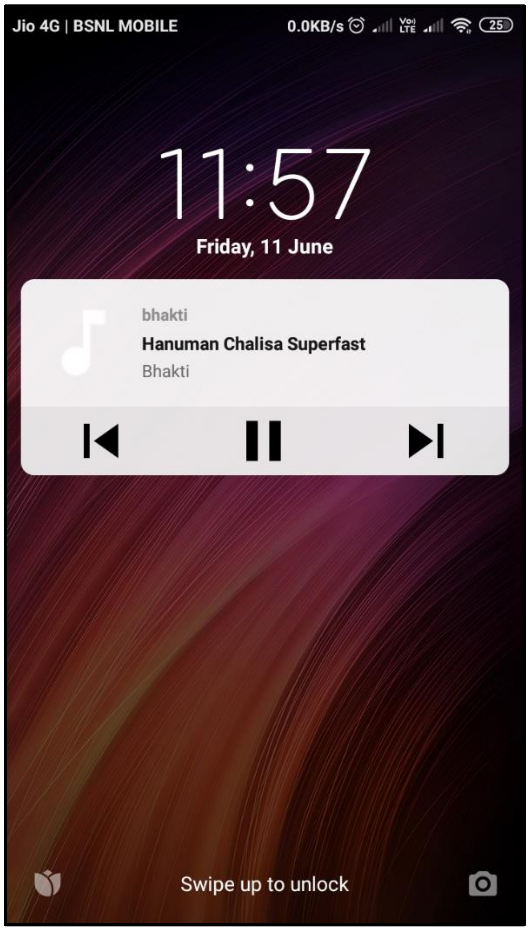
Shree Ram Stuti sarangpur Hanuman Dada



BHAKTI APP



11:57
Fri, 11 Jun
0.4KB/s
25
Today: 0B
This month: 13.8KB
bhakti
Hanuman Chalisa Superfast
Bhakti



Jio 4G | BSNL MOBILE
0.0KB/s
25
11:57
Friday, 11 June
bhakti
Hanuman Chalisa Superfast
Bhakti
Swipe up to unlock