Source code :

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
android:id="@+id/relativeLayout1"
android:layout_width="fill_parent"
android:layout_height="fill_parent" >
<LinearLayout
android:id="@+id/linearLayout1"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentLeft="true"
android:layout_alignParentRight="true"
android:layout_alignParentTop="true" >
<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_gravity="center"
android:text="ADDITION"
android:textSize="20dp" >
</TextView>
</LinearLayout>
<LinearLayout
android:id="@+id/linearLayout2"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentLeft="true"
android:layout_alignParentRight="true"
android:layout_below="@+id/linearLayout1" >
<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="ENTER NO 1" >
</TextView>
<EditText
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="0.20"
android:id="@+id/edittext1"
android:inputType="number">
</EditText>
</LinearLayout>
<LinearLayout
android:id="@+id/linearLayout3"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentLeft="true"
android:layout_alignParentRight="true"
android:layout_below="@+id/linearLayout2" >
<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
```

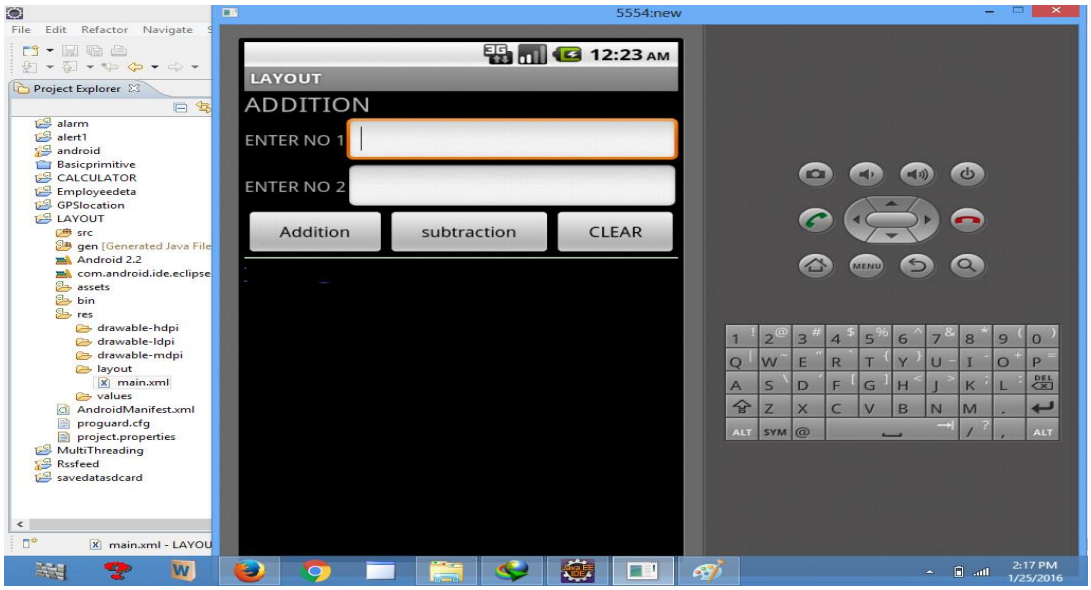
```
android:text="ENTER NO 2" >
</TextView>
<EditText
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="0.20"
android:id="@+id/edittext2"
android:inputType="number">
</EditText>
</LinearLayout>
<LinearLayout
android:id="@+id/linearLayout4"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentLeft="true"
android:layout_alignParentRight="true"
android:layout_below="@+id/linearLayout3" >
<Button
android:layout_width="wrap_content"
android:id="@+id/button1"
android:layout_height="wrap_content"
android:text="Addition"
android:layout_weight="0.50" />
<Button
android:layout_width="wrap_content"
android:id="@+id/button3"
android:layout_height="wrap_content"
android:text="subtraction"
android:layout_weight="0.50" />
<Button
android:layout_width="wrap_content"
android:id="@+id/button2"
android:layout_height="wrap_content"
android:text="CLEAR"
android:layout_weight="0.50" />
</LinearLayout>
<View
android:layout_height="2px"
android:layout_width="fill_parent"
android:layout_below="@+id/linearLayout4"
android:background="#DDFFDD"/>
</RelativeLayout>
```

```
package layout.ne;
import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
```

```
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
public class LAYOUTActivity extends Activity {
/** Called when the activity is first created. */
EditText txtData1,txtData2;
float num1,num2,result1,result2;
@Override
public void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.main);
Button add = (Button) findViewById(R.id.button1);
add.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
try
{
txtData1 = (EditText) findViewById(R.id.edittext1);
txtData2 = (EditText) findViewById(R.id.edittext2);
num1 = Float.parseFloat(txtData1.getText().toString());
num2 = Float.parseFloat(txtData2.getText().toString());
result1=num1+num2;
Toast.makeText(getBaseContext(),"ANSWER:"+result1,Toast.LENGTH_SHORT).show();
}
catch(Exception e)
{
Toast.makeText(getBaseContext(), e.getMessage(),
Toast.LENGTH_SHORT).show();
}
}
});
Button sub = (Button) findViewById(R.id.button3);
sub.setOnClickListener(new OnClickListener() {
public void onClick(View v) {
try
{
txtData1 = (EditText) findViewById(R.id.edittext1);
txtData2 = (EditText) findViewById(R.id.edittext2);
num1 = Float.parseFloat(txtData1.getText().toString());
num2 = Float.parseFloat(txtData2.getText().toString());
result2=num1-num2;
Toast.makeText(getBaseContext(),"ANSWER:"+result2,Toast.LENGTH_SHORT).show();
}
catch(Exception e)
{
Toast.makeText(getBaseContext(), e.getMessage(),
Toast.LENGTH_SHORT).show();
}
}
});
Button clear = (Button) findViewById(R.id.button2);
clear.setOnClickListener(new OnClickListener() {
```

```
public void onClick(View v) {
try
{
txtData1.setText("");
txtData2.setText("");
}
catch(Exception e)
{
Toast.makeText(getBaseContext(), e.getMessage(),
Toast.LENGTH_SHORT).show();
}
} });
}  }
```

Output:



Source code:

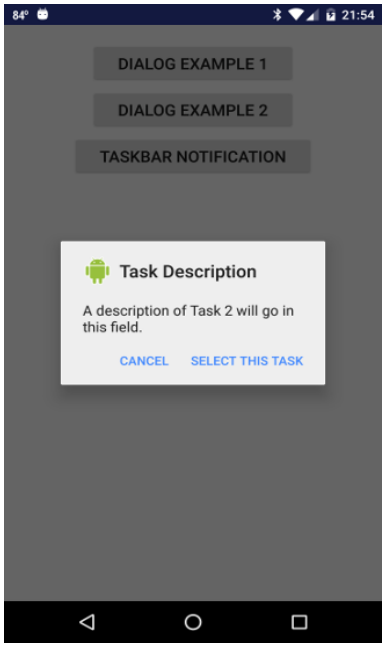
```
Package creatingdialogs;
import android.app.AlertDialog;
import android.app.Dialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Build;
import android.os.Bundle;
import android.support.v4.app.DialogFragment;
import android.support.v4.app.FragmentActivity;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Toast;

public class MainActivity extends FragmentActivity implements OnClickListener {

    private static final int numberTasks = 2;
    private static final String launcherTitle = "Task Description";
    private static final int launcherIcon = R.drawable.dialog_icon;
    private static int buttonPressed;
    private String[] taskDescription = new String[numberTasks];

    private static int dialogTheme;  // Integer defining the dialog theme
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        // Identify buttons in XML layout and attach click  listeners to each

        View button01 = findViewById(R.id.button01);
        button01.setOnClickListener(this);
        View button02 = findViewById(R.id.button02);
        button02.setOnClickListener(this);
        View button03 = findViewById(R.id.button03);
        button03.setOnClickListener(this);

        // Extract  task description strings from strings.xml and place in an array for later use

        taskDescription[0] = getString(R.string.task_description1);
        taskDescription[1] = getString(R.string.task_description2);

        // Use Material Design them if API 23 or later; Holo Light if earlier

        if (Build.VERSION.SDK_INT < Build.VERSION_CODES.LOLLIPOP) {
            dialogTheme = AlertDialog.THEME_HOLO_LIGHT;  // Deprecated with API 23
        } else {
            dialogTheme = R.style.MyDialogTheme;
        }  }
```

```
@Override
public void onPause() {
   super.onPause();
}

@Override
public void onResume() {
   super.onResume();
}

@Override
public void onClick(View v) {

   switch (v.getId()) {
      // For buttons 1 and 2, launch floating dialogs
      case R.id.button01:
         buttonPressed = 1;

         // Set alert content. If icon or title are omitted or set to null, they
         // will not appear in the Alert Dialog window.

         AlertFragment.context = this;
         AlertFragment.iconID = launcherIcon;
         AlertFragment.title = launcherTitle;
         AlertFragment.message = taskDescription[buttonPressed - 1];

         DialogFragment fragment = new AlertFragment();
         fragment.show(getSupportFragmentManager(), "Task 1");

         break;

      case R.id.button02:
         buttonPressed = 2;
         showTaskDialog(launcherTitle, taskDescription[buttonPressed - 1], launcherIcon, this);
         break;

      // For button 3, start a service that will place a notification in the task bar
      case R.id.button03:
         startTheService();
         break;
   }

}


private void showTaskDialog(String title, String message, int icon, Context context) {

   AlertDialog.Builder builder = new AlertDialog.Builder(context, dialogTheme);
   builder.setMessage(message).setTitle(title).setIcon(icon);
   // Add the buttons
```

```
    builder.setPositiveButton("Select this Task", new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialog, int id) {
            launchTask();
        }
    });
    builder.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialog, int id) {
            // Can execute additional code here if desired
            // Default is cancellation of dialog window.
        }
    });

    AlertDialog dialog = builder.create();
    dialog.show();
}
private void launchTask() {

    // Illustrate a Toast notification
    Toast.makeText(this, "launchTask() executed", Toast.LENGTH_SHORT).show();

    switch (buttonPressed) {
        case 1:    // Launch task 1
            Intent i = new Intent(this, TaskActivity1.class);
            startActivity(i);
            break;
        case 2:    // Launch task 2
            Intent j = new Intent(this, TaskActivity2.class);
            startActivity(j);
            break;
    }
}

// Start a service to demo a status bar notification

public void startTheService() {
    Intent serviceIntent = new Intent(this, MyNotificationService.class);
    this.startService(serviceIntent);
}
public static class AlertFragment extends DialogFragment {

    public static Context context;
    public static String message;
    public static String title;
    public static int iconID;
    public static int buttonPressed;

    @Override
    public Dialog onCreateDialog(Bundle savedInstanceState) {

        // Use the Builder class to construct the dialog.  Use the
        // form of the builder constructor that allows a theme to be set
```

```
AlertDialog.Builder builder = new AlertDialog.Builder(getActivity(), MainActivity.dialogTheme);
        if (title != null) builder.setTitle(title);
        if (iconID != 0) builder.setIcon(iconID);
        builder.setMessage(message)
                .setPositiveButton("Select this task", new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog, int id) {
                        launchTask();
                    }
                })
                .setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog, int id) {
                        // Default is to cancel the dialog window.  Can add
                        // additional commands if desired.
                    }
                });
        // Create the AlertDialog object and return it
        return builder.create();
    }

    // Method to launch new activity when button pressed in Alert Dialog
    private void launchTask() {
        Intent i = new Intent(context, TaskActivity1.class);
        startActivity(i);
    }
  }
}
```

Output:



Source code :

```
<AbsoluteLayout xmlns:android="http://schemas.android.com/apk/res/android"
android:id="@+id/myLayout"
android:stretchColumns="0"
android:layout_width="fill_parent"
android:layout_height="fill_parent">
<TextView android:text="@string/title"
android:layout_x="110dp"
android:layout_y="10dp"
android:layout_width="wrap_content"
android:layout_height="wrap_content"/>
<TextView android:text="@string/empid"
android:layout_x="30dp"
android:layout_y="50dp"
android:layout_width="wrap_content"
android:layout_height="wrap_content"/>
<EditText android:id="@+id/editEmpid"
android:inputType="number"
android:layout_x="150dp"
android:layout_y="50dp"
android:layout_width="150dp"
android:layout_height="40dp"/>
<TextView android:text="@string/name"
android:layout_x="30dp"
android:layout_y="100dp"
android:layout_width="wrap_content"
android:layout_height="wrap_content"/>
<EditText android:id="@+id/editName"
android:inputType="text"
android:layout_x="150dp"
android:layout_y="100dp"
android:layout_width="150dp"
android:layout_height="40dp"/>
<TextView android:text="@string/salary"
android:layout_x="30dp"
android:layout_y="150dp"
android:layout_width="wrap_content"
android:layout_height="wrap_content"/>
<EditText android:id="@+id/editsalary"
android:inputType="number"
android:layout_x="150dp"
android:layout_y="150dp"
android:layout_width="150dp"
android:layout_height="40dp"/>
<Button android:id="@+id/btnAdd"
android:text="@string/add"
android:layout_x="30dp"
android:layout_y="200dp"
android:layout_width="130dp"
android:layout_height="40dp"/>
```

```
<Button android:id="@+id/btnDelete"
android:text="@string/delete"
android:layout_x="160dp"
android:layout_y="200dp"
android:layout_width="130dp"
android:layout_height="40dp"/>n
<Button android:id="@+id/btnModify"
android:text="@string/modify"
android:layout_x="30dp"
android:layout_y="250dp"
android:layout_width="130dp"
android:layout_height="40dp"/>
<Button android:id="@+id/btnView"
android:text="@string/view"
android:layout_x="160dp"
android:layout_y="250dp"
android:layout_width="130dp"
android:layout_height="40dp"/>
<Button android:id="@+id/btnViewAll"
android:text="@string/view_all"
android:layout_x="85dp"
android:layout_y="300dp"
android:layout_width="150dp"
android:layout_height="40dp"/>
</AbsoluteLayout>
```

Go to values folder and select string.xml file.Replace the code below

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
<string name="app_name">Employee detail1</string>
<string name="hello">Hello World, Employee detail Activity!</string>
<string name="title">Employee Details</string>
<string name="empid">Enter Employee ID: </string>
<string name="name">Enter Name: </string>
<string name="salary">Enter salary: </string>
<string name="add">Add Employee</string>
<string name="delete">Delete Employee</string>
<string name="modify">Modify Employee</string>
<string name="view">View Employee</string>
<string name="view_all">View All Employee</string>
</resources>
```

8) Now select mainactivity.java file and type the following code.In my coding maniactivity name is EmployeedetailActivity.

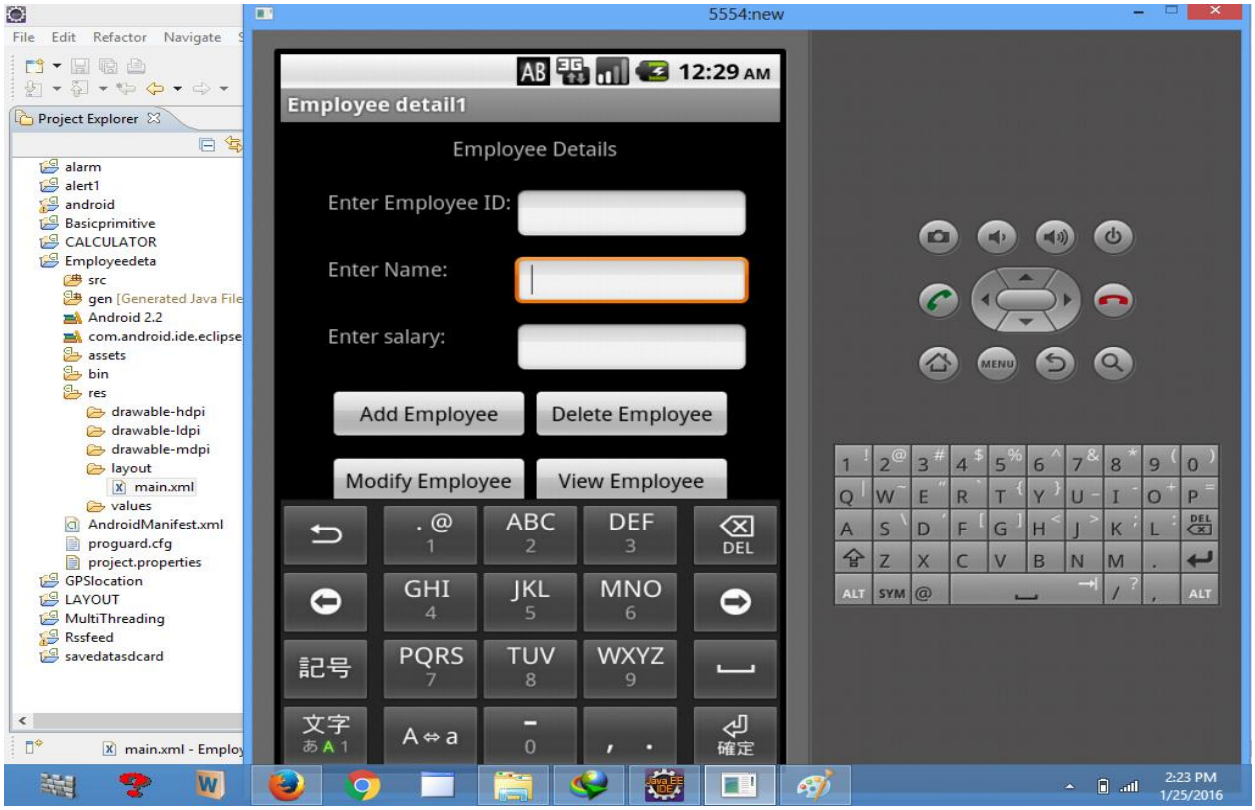
```
package employee.detail;
//import android.R;
import android.app.Activity;
import android.app.AlertDialog.Builder;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
```

```
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;
public class EmployeedetailActivity extends Activity implements OnClickListener {
EditText editEmpid,editName,editsalary;
Button btnAdd,btnDelete,btnModify,btnView,btnViewAll;
SQLiteDatabase db;
/** Called when the activity is first created. */
@Override
public void onCreate(Bundle savedInstanceState)
{
super.onCreate(savedInstanceState);
setContentView(R.layout.main);
editEmpid=(EditText)findViewById(R.id.editEmpid);
editName=(EditText)findViewById(R.id.editName);
editsalary=(EditText)findViewById(R.id.editsalary);
btnAdd=(Button)findViewById(R.id.btnAdd);
btnDelete=(Button)findViewById(R.id.btnDelete);
btnModify=(Button)findViewById(R.id.btnModify);
btnView=(Button)findViewById(R.id.btnView);
btnViewAll=(Button)findViewById(R.id.btnViewAll);
btnAdd.setOnClickListener(this);
btnDelete.setOnClickListener(this);
btnModify.setOnClickListener(this);
btnView.setOnClickListener(this);
btnViewAll.setOnClickListener(this);
db=openOrCreateDatabase("EmployeeDB", Context.MODE_PRIVATE, null);
db.execSQL("CREATE TABLE IF NOT EXISTS employee(empid VARCHAR,name
VARCHAR,salary VARCHAR);");
}
public void onClick(View view)
{
if(view==btnAdd)
{
if(editEmpid.getText().toString().trim().length()==0||
editName.getText().toString().trim().length()==0||
editsalary.getText().toString().trim().length()==0)
{
showMessage("Error", "Please enter all values");
return;
}
db.execSQL("INSERT INTO employee
VALUES('"+editEmpid.getText()+"','"+editName.getText()+
"','"+editsalary.getText()+"');");
showMessage("Success", "Record added");
clearText();
}
if(view==btnDelete)
{
```

```
if(editEmpid.getText().toString().trim().length()==0)
{
showMessage("Error", "Please enter Employee id");
return;
}
Cursor c=db.rawQuery("SELECT * FROM employee WHERE
empid='"+editEmpid.getText()+"'", null);
if(c.moveToFirst())
{
db.execSQL("DELETE FROM employee WHERE
empid='"+editEmpid.getText()+"'");
showMessage("Success", "Record Deleted");
}
else
{
showMessage("Error", "Invalid Employee id");
}
clearText();
}
if(view==btnModify)
{
if(editEmpid.getText().toString().trim().length()==0)
{
showMessage("Error", "Please enter Employee id");
return;
}
Cursor c=db.rawQuery("SELECT * FROM employee WHERE
empid='"+editEmpid.getText()+"'", null);
if(c.moveToFirst())
{
db.execSQL("UPDATE employee SET
name='"+editName.getText()+"',salary='"+editsalary.getText()+
"' WHERE empid='"+editEmpid.getText()+"'");
showMessage("Success", "Record Modified");
}
else
{
showMessage("Error", "Invalid Rollno");
}
clearText();
}
if(view==btnView)
{
if(editEmpid.getText().toString().trim().length()==0)
{
showMessage("Error", "Please enter Employee id");
return;
}
Cursor c=db.rawQuery("SELECT * FROM employee WHERE
empid='"+editEmpid.getText()+"'", null);
if(c.moveToFirst())
```

```
{
editName.setText(c.getString(1));
editsalary.setText(c.getString(2));
}
else
{
showMessage("Error", "Invalid Employee id");
clearText();
}
}
if(view==btnViewAll)
{
Cursor c=db.rawQuery("SELECT * FROM employee", null);
if(c.getCount()==0)
{
showMessage("Error", "No records found");
return;
}
StringBuffer buffer=new StringBuffer();
while(c.moveToNext())
{
buffer.append("Employee id: "+c.getString(0)+"\n");
buffer.append("Name: "+c.getString(1)+"\n");
buffer.append("salary: "+c.getString(2)+"\n\n");
}
showMessage("Employee details Details", buffer.toString());
}
}
public void showMessage(String title,String message)
{
Builder builder=new Builder(this);
builder.setCancelable(true);
builder.setTitle(title);
builder.setMessage(message);
builder.show();
}
public void clearText()
{
editEmpid.setText("");
editName.setText("");
editsalary.setText("");
editEmpid.requestFocus();
}
}
```

OUTPUT:

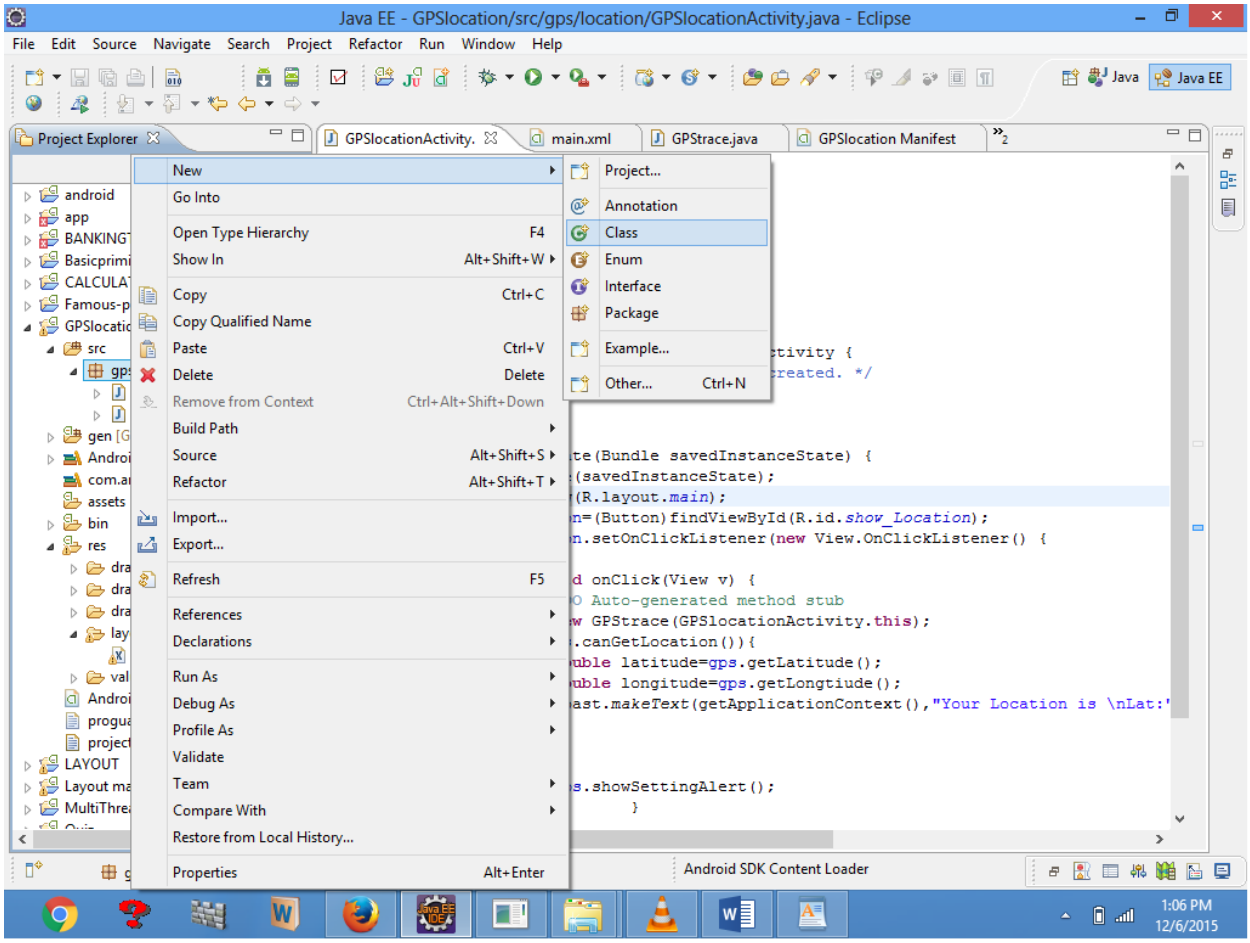

5554:new
12:29 AM
Employee detail1
Employee Details
Enter Employee ID:
Enter Name:
Enter salary:
Add Employee
Delete Employee
Modify Employee
View Employee
Project Explorer
alarm
alert1
android
Basicprimitive
CALCULATOR
Employeedeta
src
gen [Generated Java File
Android 2.2
com.android.ide.eclipse
assets
bin
res
drawable-hdpi
drawable-ldpi
drawable-mdpi
layout
main.xml
values
AndroidManifest.xml
proguard.cfg
project.properties
GPSlocation
LAYOUT
MultiThreading
Rssfeed
savedatasdcard
2:23 PM
1/25/2016

Source code:

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
android:id="@+id/relativeLayout1"
android:layout_width="match_parent"
android:layout_height="match_parent" >
<Button
android:id="@+id/show_Location"
android:layout_width="wrap_content"
android:layout_height="wrap_content
android:text="Show_Location"
android:layout_centerVertical="true"
android:layout_centerHorizontal="true"
/>
</RelativeLayout>

package gps.location;
//import android.R;
import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;
public class GPSlocationActivity extends Activity {
/** Called when the activity is first created. */
Button btnShowLocation;
GPStrace gps;
@Override
public void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.main);
btnShowLocation=(Button)findViewById(R.id.show_Location);
btnShowLocation.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
// TODO Auto-generated method stub
gps=new GPStrace(GPSlocationActivity.this);
if(gps.canGetLocation()){
double latitude=gps.getLatitude();
double longitude=gps.getLongtiude();
Toast.makeText(getApplicationContext(),"Your Location is
\nLat:"+latitude+"\nLong:"+longitude, Toast.LENGTH_LONG).show();
}
else
{
gps.showSettingAlert();
}
} });  } }
```

)Go to src folder and Right Click on your package folder and choose new class and give the

class nams as GPStrace



9)Select the GPStrace.java file and paste the following code.

```
package gps.location;
import android.app.AlertDialog;
import android.app.Service;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.os.IBinder;
import android.provider.Settings;
public class GPStrace extends Service implements LocationListener{
private final Context context;
boolean isGPSEnabled=false;
boolean canGetLocation=false;
boolean isNetworkEnabled=false;
Location location;
double latitude;
double longtitude;
```

```
private static final long MIN_DISTANCE_CHANGE_FOR_UPDATES=10;
private static final long MIN_TIME_BW_UPDATES=1000*60*1;
protected LocationManager locationManager;
public GPStrace(Context context)
{
this.context=context;
getLocation();
}
public Location getLocation()
{
try{
locationManager=(LocationManager) context.getSystemService(LOCATION_SERVICE);
isGPSEnabled=locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER);
isNetworkEnabled=locationManager.isProviderEnabled(LocationManager.NETWORK_PROVI
DER);
if(!isGPSEnabled && !isNetworkEnabled){
}else{
this.canGetLocation=true;
if(isNetworkEnabled){
locationManager.requestLocationUpdates(
LocationManager.NETWORK_PROVIDER,
MIN_TIME_BW_UPDATES,
MIN_DISTANCE_CHANGE_FOR_UPDATES,this);
}
if(locationManager!=null){
location=locationManager.getLastKnownLocation(LocationManager.NETWORK_PROVIDER)
;
if(location !=null){
latitude=location.getLatitude();
longtitude=location.getLongitude();
}
}
}
if(isGPSEnabled){
if(location==null){
locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER,MIN_TIME_B
W_UPDATES, MIN_DISTANCE_CHANGE_FOR_UPDATES, this);
if(locationManager!=null){
location=locationManager.getLastKnownLocation(LocationManager.GPS_PROVIDER);
if(location!=null){
latitude=location.getLatitude();
longtitude=location.getLongitude();
}
}
}
}
}
catch(Exception e)
{
e.printStackTrace();
}
```

```
return location;
}
public void stopUsingGPS(){
if(locationManager!=null){
locationManager.removeUpdates(GPStrace.this);
}
}
public double getLatitude(){
if(location!=null){
latitude=location.getLatitude();
}
return latitude;
}
public double getLongtiude(){
if(location!=null){
longtitude=location.getLatitude();
}
return longtitude;
}
public boolean canGetLocation(){
return this.canGetLocation;
}
public void showSettingAlert(){
AlertDialog.Builder alertDialog=new AlertDialog.Builder(context);
alertDialog.setTitle("GPS is settings");
alertDialog.setMessage("GPS is not enabled.Do you want to go to setting menu?");
alertDialog.setPositiveButton("settings", new DialogInterface.OnClickListener() {
@Override
public void onClick(DialogInterface dialog,int which){
Intent intent=new Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS);
context.startActivity(intent);
}
});
alertDialog.setNegativeButton("cancel", new DialogInterface.OnClickListener() {
@Override
public void onClick(DialogInterface dialog, int which) {
// TODO Auto-generated method stub
dialog.cancel();
}
});
alertDialog.show();
}
@Override
public void onLocationChanged(Location location) {
// TODO Auto-generated method stub
}
@Override
public void onProviderDisabled(String provider) {
// TODO Auto-generated method stub
}
@Override
```

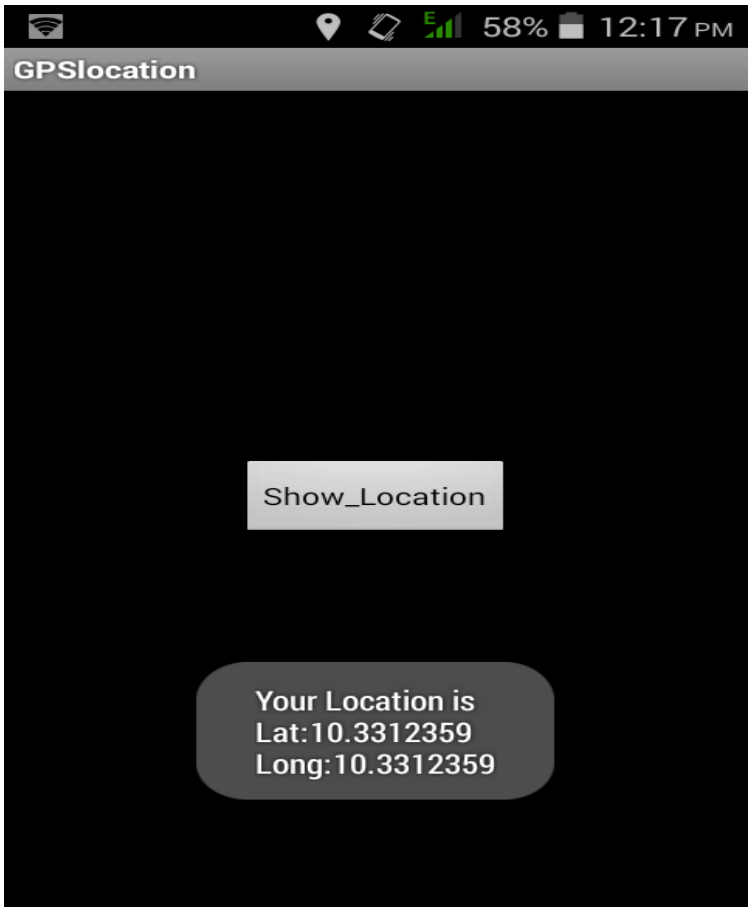
```
public void onProviderEnabled(String provider) {
// TODO Auto-generated method stub
}
@Override
public void onStatusChanged(String provider, int status, Bundle extras) {
// TODO Auto-generated method stub
}
@Override
public IBinder onBind(Intent intent) {
// TODO Auto-generated method stub
return null;
}
}
```

10)Go to manifest.xml file and add the code below

```
<uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION"/>
<uses-permission
android:name="android.permission.INTERNET"/>
```

- Now go to main.xml and right click .select run as option and select run configuration
- Android output is present in the android emulator as shown in below.

Output:

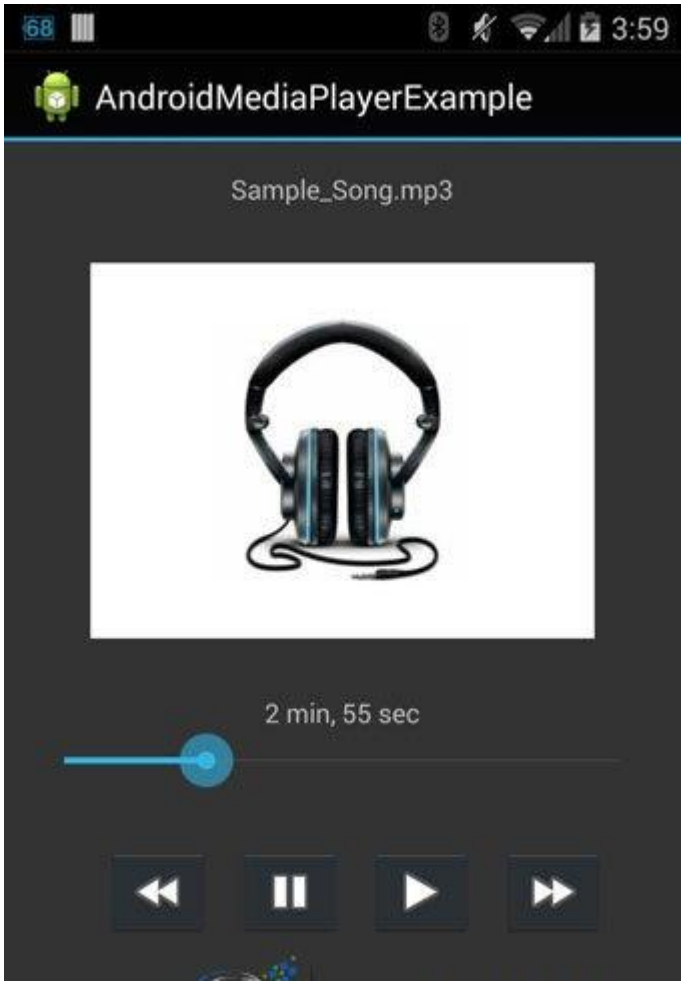


Source code:

```
import java.util.concurrent.TimeUnit;
import android.app.Activity;
import android.media.MediaPlayer;
import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.SeekBar;
import android.widget.TextView;

public class AndroidMediaPlayerExample extends Activity {

   private MediaPlayer mediaPlayer;
   public TextView songName, duration;
   private double timeElapsed = 0, finalTime = 0;
   private int forwardTime = 2000, backwardTime = 2000;
   private Handler durationHandler = new Handler();
   private SeekBar seekbar;

   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);

      //set the layout of the Activity
      setContentView(R.layout.activity_main);

      //initialize views
      initializeViews();
   }

   public void initializeViews(){
      songName = (TextView) findViewById(R.id.songName);
      mediaPlayer = MediaPlayer.create(this, R.raw.sample_song);
      finalTime = mediaPlayer.getDuration();
      duration = (TextView) findViewById(R.id.songDuration);
      seekbar = (SeekBar) findViewById(R.id.seekBar);
      songName.setText("Sample_Song.mp3");

      seekbar.setMax((int) finalTime);
      seekbar.setClickable(false);
   }

   // play mp3 song
   public void play(View view) {
      mediaPlayer.start();
      timeElapsed = mediaPlayer.getCurrentPosition();
      seekbar.setProgress((int) timeElapsed);
      durationHandler.postDelayed(updateSeekBarTime, 100);
   }
```

```
    //handler to change seekBarTime
    private Runnable updateSeekBarTime = new Runnable() {
        public void run() {
            //get current position
            timeElapsed = mediaPlayer.getCurrentPosition();
            //set seekbar progress
            seekbar.setProgress((int) timeElapsed);
            //set time remaing
            double timeRemaining = finalTime - timeElapsed;
            duration.setText(String.format("%d min, %d sec", TimeUnit.MILLISECONDS.toMinutes((long)
timeRemaining), TimeUnit.MILLISECONDS.toSeconds((long) timeRemaining) -
TimeUnit.MINUTES.toSeconds(TimeUnit.MILLISECONDS.toMinutes((long) timeRemaining))));

            //repeat yourself that again in 100 miliseconds
            durationHandler.postDelayed(this, 100);
        }
    };

    // pause mp3 song
    public void pause(View view) {
        mediaPlayer.pause();
    }

    // go forward at forwardTime seconds
    public void forward(View view) {
        //check if we can go forward at forwardTime seconds before song endes
        if ((timeElapsed + forwardTime)  0) {
            timeElapsed = timeElapsed - backwardTime;

            //seek to the exact second of the track
            mediaPlayer.seekTo((int) timeElapsed);
        }
    }

}//handler to change seekBarTime
private Runnable updateSeekBarTime = new Runnable() {
    public void run() {
        //get current position
        timeElapsed = mediaPlayer.getCurrentPosition();
        //set seekbar progress
        seekbar.setProgress((int) timeElapsed);
        //set time remaing
        double timeRemaining = finalTime - timeElapsed;
        duration.setText(String.format("%d min, %d sec", TimeUnit.MILLISECONDS.toMinutes((long)
timeRemaining), TimeUnit.MILLISECONDS.toSeconds((long) timeRemaining) -
TimeUnit.MINUTES.toSeconds(TimeUnit.MILLISECONDS.toMinutes((long) timeRemaining))));

        //repeat yourself that again in 100 miliseconds
        durationHandler.postDelayed(this, 100);
    };
```

Output:


3:59
AndroidMediaPlayerExample
Sample_Song.mp3
2 min, 55 sec

Step 1: Design your Application:

Before you commit to an approach, it is advisable to make sure you have all the requirements ready so that it can meet these requirements.

For example, if you are creating a pure HTML5 mobile application, then you have access to only those features that are available to the browser.

Also, ensure you have the Pre-requisite: – Cordova CLI and the Android SDK tools, ready. For installing Cordova CLI, you first need to install Node.js version 0.10+.

Next run the following command to install the Cordova CLI:

sumo npm install -g cordova

Next, install the Android SDK tools from

http://developer.android.com/sdk/installing/index.html

Step 2 – HTML5 Mobile Framework Application

The foundation stone for building an HTML5 mobile application is the HTML5 mobile frameworks. While using jQuery on a website, it allows you to easily create animations, show and hide things, and so on.

Finally, get to adding your website. Edit your index.html page for content.

Step 3 -Testing Application through Browser

The best thing about an HTML5 framework is that, you can test it directly through your local server browser. You will be able to access your application through a local web server by going to:

http://localhost/MyApp

If you are using the Chrome DevTools, you can even imitate the view by clicking the mobile icon in the top left corner of the screen. From here, you can debug your application to your content using the Console, Network, Time-line etc.

Step 4- Package your Application

Using third party applications like PhoneGap and Xamarin, you can package your application in a native wrapper which act as a bridge between your app and the native API's.

Third party application loads your app in the web view, which in turn displays your application to the user.

Now we can generate a release version of the APK

Cordova build –release

Your APK file should be located here:

platforms/android/ant-build/MyApp-release-unsigned.apk

To submit it to the App Store, we need to sign in.

Step 5 – Testing Application on a Device

As we mentioned before, the best thing about the HTML5 application is that you can test it in any local browser. To test the application using native API's, you will have to run it on an actual device.But do you know to debug without your browser debugging tools?

Luckily, there are a few good options to debug directly on your device. One of the few good options is to use the ***GapDebug***. It allows you to install. ipa files for iOS or .apk for Android onto your device and provides a debugging interface that exactly imitates the Chrome Dev Tools. Any kind of changes you make in this interface will be reflected on the live application.

Step 6- Distribute on App Store

Once you have finished with your debugging and ready for the launch, you will first have to sign your application to the Play Store before you can install it on any device.

To sign your application, you will simply need a key store file

You create a keystore with the Java Keytool utility that comes with any standard JDK distribution and can be located at %JAVA_HOME%\bin. On Windows, this would usually be C:\Program Files\Java\jre7\bin

(Source: https://stackoverflow.com/questions/3997748/how-can-i-create-a-keystore)

Final Process:

Sign up as a Google Play Developer and prepare your app store listings. Thankfully this is the least tiny hurdle you need to pass the test.

Wrapup

You can also opt for converting an existing Android app and adding hybrid components to it. Android Studio has now adapted well to the Android O platform for developing hybrid apps. However, with the release of Android P developer review, the developers will soon have to upgrade to the still newer development codes due to behaviour changes and new features and API that will be brought along. These hybrid apps created using Android Studio help you share data back and forth between the web and native platforms and make attractive use of them.