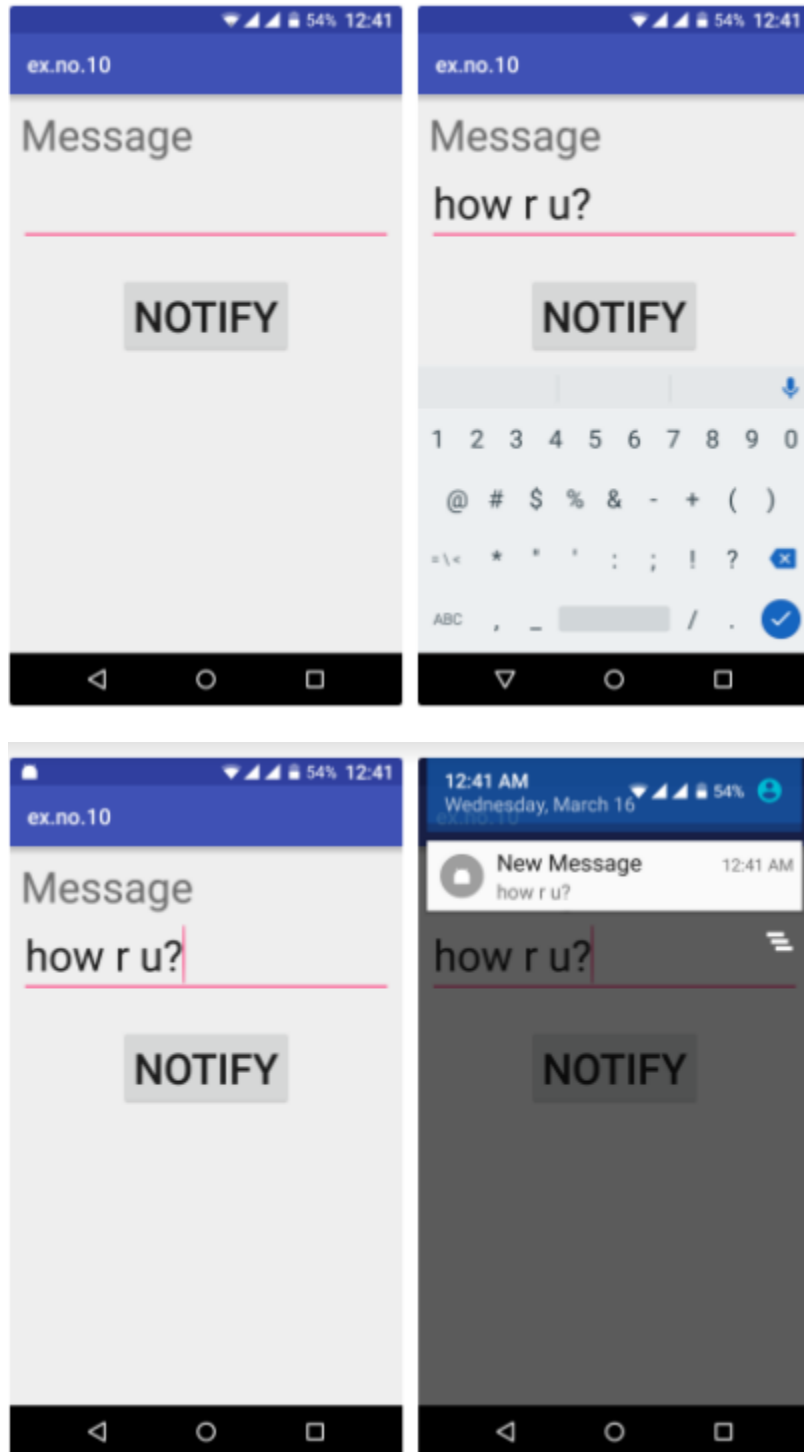


Experiment: 04**Aim: Implement an application that creates an alert upon receiving a message****Theory:**

Implementation:

```
import android.app.Notification;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Context;
import android.content.Intent;
import androidx.core.app.NotificationCompat;
private static final String CHANNEL_ID = "01";

private void createNotificationChannel() {
    // Create the NotificationChannel, but only on API 26+ because
    // the NotificationChannel class is new and not in the support library
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
        CharSequence name = getString(R.string.channel_name);
        String description = getString(R.string.channel_description);
        int importance = NotificationManager.IMPORTANCE_DEFAULT;
        NotificationChannel channel = new NotificationChannel(CHANNEL_ID, name,
        importance);
        channel.setDescription(description);
        // Register the channel with the system; you can't change the importance
        // or other notification behaviors after this
        NotificationManager notificationManager =
        getSystemService(NotificationManager.class);
        notificationManager.createNotificationChannel(channel);
    }
}

public void notifyNow(){
    Intent intent = new Intent(this, ResultActivity.class);
```

```
intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);

PendingIntent pendingIntent = PendingIntent.getActivity(this, 0, intent, 0);

NotificationCompat.Builder builder = new NotificationCompat.Builder(this,
CHANNEL_ID)

    .setSmallIcon(R.drawable.ic_launcher_foreground)

    .setContentTitle("LocTrack Alert!")

    .setContentText("Latitude and Longitude Saved to Database")

    .setPriority(NotificationCompat.PRIORITY_DEFAULT)

    // Set the intent that will fire when the user taps the notification

    .setContentIntent(pendingIntent)

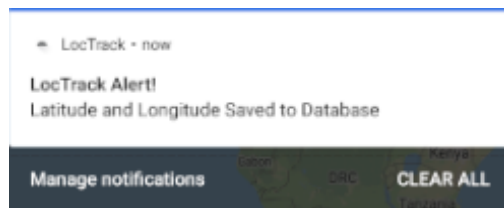
    .setAutoCancel(true);

NotificationManager
notif=(NotificationManager)getSystemService(Context.NOTIFICATION_SERVICE);

    notif.notify(6934,builder.build());

}
```

Output:



Conclusion:

From this experiment, we learnt how to create notifications in android applications. We first create an Intent, Notification channel and a Notification compat_builder. Notification compat builder will set the icon, text and the priority with other details of the notification whereas notification channel will create channel sequence. We use Notification Manager to finally build and notify the notification that we created.