

מעבדה בפיתוח באנדרואיד

מספר קורס: 61985

מרצה: דניאל בואנוס

שירות באנדרואיד – SERVICE

- תהליך שביצעו עלול לקחת הרבה זמן ובכך לשבש את האינטראקציה של היישום עם המשתמש
- לרוב נרצה שהוא ימשיך לרוץ גם כשהיישום אינו מתקשר עם המשתמש
- למשל: הורדת סרטון ארוך, עיבוד אלגוריתמי מסובך...
- ניתן להפעיל ולתקשר עם השירות גם מיישום אחר
- סוגי שירות:



– Foreground Service

– Background Service

– Bound Service (לא יתורגל במסגרת השיעור)

<https://developer.android.com/guide/components/services>

נקודות התייחסות בעבודה עם שירות

- רישום השירות במניפסט
- החלטה על סוג השירות להפעיל
 - ראה דוגמה להפעלה בשקף הבא
- הפעלת Thread בתוך השירות שמבצע את

	Foreground	Background	Bound
startService/stopService/stopSelf	+	+	+
startForeground/Notification	+		
bindService/unbindService			+
Use internal Thread	+	+	+
sendBroadcast (optional)		+	
Override onStartCommand	+	+	+

דוגמה איך להפעיל שירות

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    .....
    if (!isMyServiceRunning(MyBackground.class)) {
        Intent intent = new Intent(this, MyBackground.class);
        startService(intent);
    }

    if (!isMyServiceRunning(MyForeground.class)) {
        Intent intent = new Intent(this, MyForeground.class);
        startForegroundService(intent);
    }
}
```

```
private boolean isMyServiceRunning(Class<?> serviceClass) {
    ActivityManager manager = (ActivityManager) getSystemService(Context.ACTIVITY_SERVICE);
    for (ActivityManager.RunningServiceInfo service : manager.getRunningServices(Integer.MAX_VALUE)) {
        if (serviceClass.getName().equals(service.service.getClassName())) {
            return true;
        }
    }
    return false;
}
```

שליחת notification מתוך שירות

- אחת הדרישות כדי להפוך ל-foreground Service זה לבנות מנגנון של שליחת הודעות עם הפעלת השירות ותוך כדי עבודתו
- זה מבטיח שאנדרואיד לא יעצור את השירות במקרה של מצוקת זיכרון או מצוקת סוללה

<https://developer.android.com/training/notify-user/build-notification>

תרגיל 10



- יישום עם שני TextView גדולים בפעילות הראשית שמציגים:
 - מונה שמתעדכן מתוך background service
 - 0-9 בקצב של 1000 מילישניה
 - מונה שמתעדכן מתוך foreground service
 - 0-999 בקצב של 100 מילישניה
- כל שירות יפעיל Thread שמתוכנן יקודם מונה בלולאה
- העדכון יתבצע ע"י שידור ערך המונה מהשירות לפעילות
- הפעילות תממש מקלט שידורים דינמי שיאזין לעדכון המונים ויעדכן את ה-TextViews
- ה-foreground service גם ישלח notification לכל עדכון
 - ...ראה דוגמה בשקף הבא
- בסיום הספירה ישלח שידור מיוחד שיציג Toast בפעילות הראשית
 - “End of Background counting”
 - “End of Foreground Counting”

קוד דוגמה איך לעבוד עם ה-notification

```
private void initForeground(){
    String CHANNEL_ID = "my_channel_01";
    if (mNotiMgr==null)
        mNotiMgr= (NotificationManager) getSystemService(Context.NOTIFICATION_SERVICE);

    NotificationChannel channel = new NotificationChannel(CHANNEL_ID,
        "My main channel",
        NotificationManager.IMPORTANCE_DEFAULT);

    ((NotificationManager) getSystemService(Context.NOTIFICATION_SERVICE))
        .createNotificationChannel(channel);

    // Create an explicit intent for an Activity in your app
    Intent intent = new Intent(this, MainActivity.class);
    intent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP | Intent.FLAG_ACTIVITY_SINGLE_TOP);
    PendingIntent pendingIntent = PendingIntent.getActivity(this, 0, intent, 0);

    mNotifyBuilder = new Notification.Builder(this, CHANNEL_ID)
        .setContentTitle("Testing Notification...")
        .setSmallIcon(R.drawable.ic_launcher_foreground)
        .setContentIntent(pendingIntent);

    startForeground(NOTIFICATION_ID1, updateNotification(Integer.toString(0)));
}

public Notification updateNotification(String details){
    mNotifyBuilder.setContentText(details).setOnlyAlertOnce(false);
    Notification noti = mNotifyBuilder.build();
    noti.flags = Notification.FLAG_ONLY_ALERT_ONCE;
    mNotiMgr.notify(NOTIFICATION_ID1, noti);
    return noti;
}
```

Call from
onCreate()

Call from
your Thread