



Πανεπιστήμιο Πειραιώς Τμήμα Ψηφιακών Συστημάτων

Διαδικτυακά και Φορητά Πληροφοριακά Συστήματα (Δικτυακές Υπηρεσίες)

6^η Διάλεξη

Δημοσθένης Κυριαζής

Δευτέρα 16 Απριλίου 2018



Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Android
 - Εκτέλεση εφαρμογών
 - Βασικά στοιχεία εφαρμογών
 - Εγκατάσταση περιβάλλοντος υλοποίησης εφαρμογών
 - Layouts
 - Βασικά widgets: TextView, EditText
 - Προσθήκη αντικειμένων
 - Προσθήκη λειτουργικότητας
 - Εισαγωγή
 - Δημιουργία αντικειμένων
 - Πρόσβαση στους πόρους του project
 - Events & Listeners



jQuery UI

□ Δημιουργία pop-up παραθύρου - dialog()

■ Μη αυτόματο άνοιγμα

▣ `autoOpen: false`

■ Προσαρμογή κουμπιών

`buttons: [`

`{`

`text: "Ok",`

`click: function() {`

`$(this).dialog("close"); }`

`},`

`{`

`text: "Cancel",`

`click: function() {`

`$(this).dialog("close"); }`

`}`

`]`



Τι είναι το Android

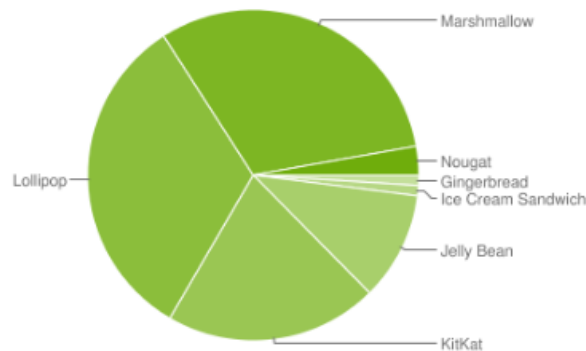
- ❑ Το Android είναι μια «στοίβα» λογισμικού (software stack) για κινητές συσκευές. Περιλαμβάνει
 - Λειτουργικό σύστημα
 - Μεσολογισμικό
 - Βασικές εφαρμογές
- ❑ Βασίζεται στη JAVA και όλες οι εφαρμογές υλοποιούνται με JAVA
- ❑ Ένα JAVA VM (Dalvik – [Android Runtime](#)), έχει δημιουργηθεί και τροποποιηθεί για εκτέλεση σε κινητές συσκευές
- ❑ Το Android SDK προσφέρει πλούσια εργαλεία για την υλοποίηση εφαρμογών καθώς και πολλά χρήσιμα APIs



Συσκευές και εκδόσεις λειτουργικού



Version	Codename	API	Distribution
2.3.3 - 2.3.7	Gingerbread	10	1.0%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	1.0%
4.1.x	Jelly Bean	16	3.7%
4.2.x		17	5.4%
4.3		18	1.5%
4.4	KitKat	19	20.8%
5.0	Lollipop	21	9.4%
5.1		22	23.1%
6.0	Marshmallow	23	31.3%
7.0	Nougat	24	2.4%
7.1		25	0.4%



Data collected during a 7-day period ending on March 6, 2017.

Any versions with less than 0.1% distribution are not shown.

Δεδομένα από περίοδο 7 ημερών – March 6, 2017



Χαρακτηριστικά (1/2)

- Πλαίσιο εφαρμογών (application framework) για την επαναχρησιμοποίηση και αντικατάσταση αντικειμένων
- Βελτιστοποίηση επεξεργασίας γραφικών - υποστήριξη γραφικών 2D και 3D (OpenGL ES)
- Ενσωματωμένος web browser (WebKit)
- Αποθήκευση δεδομένων (SQLite)



Χαρακτηριστικά (2/2)

- Υποστήριξη πολυμέσων – διαφορετικές μορφές βίντεο, ήχου και εικόνων (MPEG4, H.264, MP3, AAC, AMR, JPG, and PNG)
- Τηλεφωνία GSM
- Bluetooth, 4G και Wi-Fi
- Camera, GPS
 - επιπρόσθετε υπηρεσίες αισθητήρων
- Πλούσιο περιβάλλον υλοποίησης εφαρμογών
 - Emulator
 - Εργαλεία debugging, log
 - Εξωτερικά plugins

Συγκεκριμένα
για το
υλισμικό

APPLICATIONS

Home

Contacts

Phone

Browser

...

APPLICATION FRAMEWORK

Activity Manager

Window
Manager

Content
Providers

View
System

Notification
Manager

Package Manager

Telephony
Manager

Resource
Manager

Location
Manager

GTalk Service

LIBRARIES

Surface Manager

Media
Framework

SQLite

OpenGL | ES

FreeType

WebKit

SSL

SSL

libc

ANDROID RUNTIME

Core Libraries

Dalvik Virtual
Machine

LINUX KERNEL

Display
Driver

Camera Driver

Bluetooth
Driver

Flash Memory
Driver

Binder (IPC)
Driver

USB Driver

Keypad Driver

WiFi Driver

Audio
Drivers

Power
Management



Linux Kernel

- ❑ Το Android βασίζεται σε ένα Linux kernel, όχι ένα λειτουργικό Linux
- ❑ Παρέχει ασφάλεια, διαχείριση μνήμης, διαχείριση διεργασιών, οδηγούς συσκευών, διαχείριση δικτυακών πόρων
- ❑ Δρα ως ένα ενδιάμεσο επίπεδο μεταξύ του υλισμικού και της υπόλοιπης στοίβας του λογισμικού





Libraries

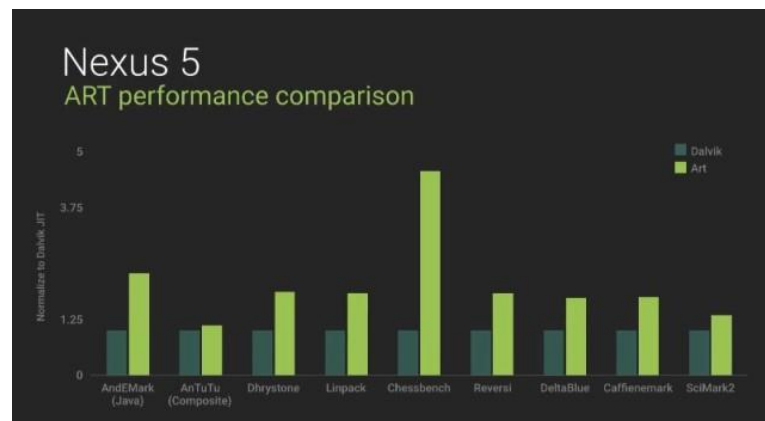
- Εκτελούνται στο system background
- Έχουν υλοποιηθεί με C/C++
- Βιβλιοθήκες
 - Συστήματος
 - Πολυμέσων
 - Surface Manager, για τη διαχείριση της πρόσβασης στο υποσύστημα της οθόνης και τη σύνθεση γραφικών 2D και 3D
 - WebKit
 - SGL, μηχανή γραφικών 2D
 - OpenGL ES, βιβλιοθήκες 3D
 - FreeType, rendering γραμματοσειρών
 - SQLite





Android Runtime

- Το βασικό συστατικό της πλατφόρμας Android
- Dalvik Virtual Machine
 - Πολλαπλά Dalvik VMs μπορεί να εκτελούνται ταυτόχρονα
 - Κάθε εφαρμογή εκτελείται στο δικό της περιβάλλον (Dalvik VM)
- Core Libraries
 - Βασικές βιβλιοθήκες της Java
- Αντικατάσταση με Android Runtime (ART) στο Android 5





Application Framework (1/2)

- Απλοποιεί την επαναχρησιμοποίηση των αντικειμένων
 - Οι εφαρμογές μπορούν να «δημοσιεύουν» τις δυνατότητές τους και άλλες εφαρμογές να τις χρησιμοποιούν
- Οι εφαρμογές είναι ένα σύνολο υπηρεσιών και συστημάτων που περιλαμβάνουν
 - Σύστημα παρουσίασης, διαχείριση πόρων, υπηρεσίες τοποθεσίας, κλπ





Application Framework (2/2)

□ Activity Manager

- Διαχειρίζεται το κύκλο ζωής των εφαρμογών και προσφέρει ένα κοινό σημείο πλοήγησης

□ Notification Manager

- Επιτρέπει στις εφαρμογές να εμφανίζουν ειδοποιήσεις στο status bar

□ Resource Manager

- Παρέχει πρόσβαση σε πόρους όπως λεκτικά συγκεκριμένων τοποθεσιών, γραφικά, αρχεία μορφοποίησης, κλπ

□ Content Providers

- Παρέχουν δεδομένα σε άλλες εφαρμογές (π.χ. επαφές)

□ Views & Widgets (android.widget package)

- Χρησιμοποιούνται για την υλοποίηση εφαρμογών (π.χ. λίστες, κουμπιά, πλαίσια κειμένου, browser, κλπ)



Applications

- Ένα σύνολο βασικών εφαρμογών που συμπεριλαμβάνονται στην πλατφόρμα Android
 - Πρόγραμμα email, SMS, ημερολόγιο, χάρτες, browser, επαφές, κλπ
- Όλες έχουν υλοποιηθεί με Java
- Οι εφαρμογές που υλοποιούνται από τους προγραμματιστές βρίσκονται σε αυτό το επίπεδο





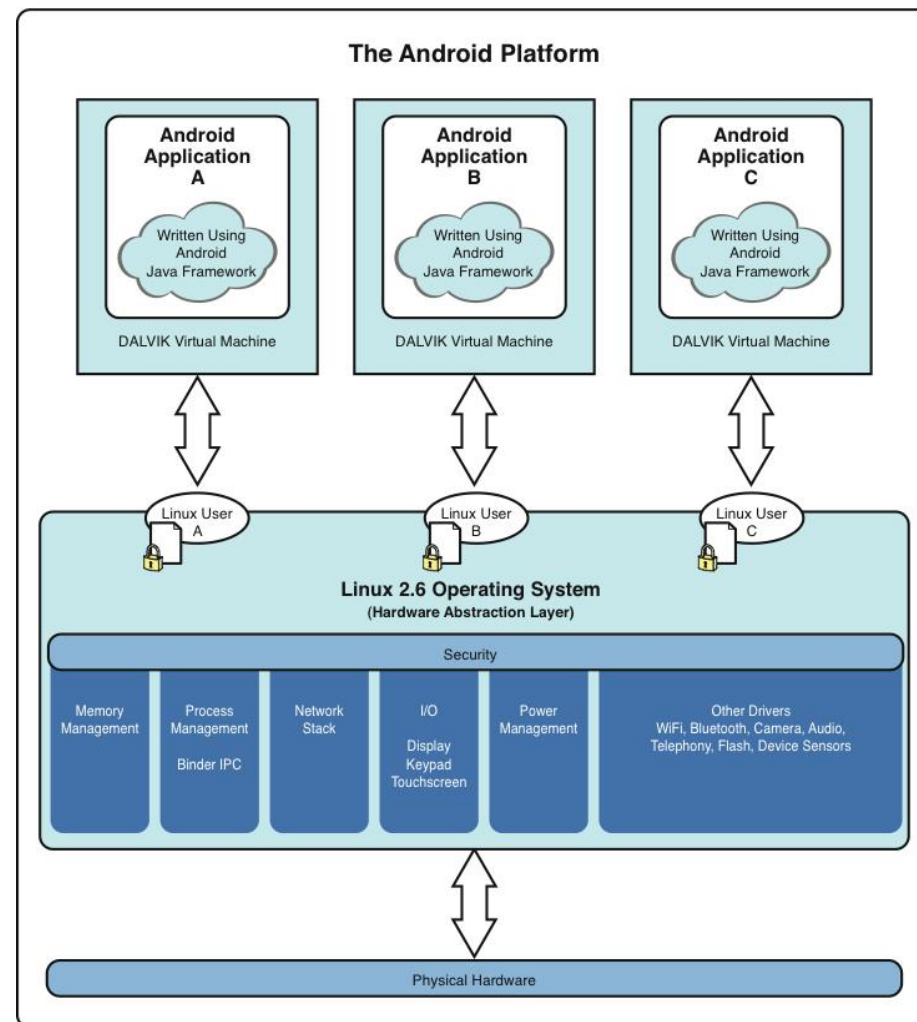
Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Android
 - Εκτέλεση εφαρμογών
 - Βασικά στοιχεία εφαρμογών
 - Εγκατάσταση περιβάλλοντος υλοποίησης εφαρμογών
 - Layouts
 - Βασικά widgets: TextView, EditText
 - Προσθήκη αντικειμένων
 - Προσθήκη λειτουργικότητας
 - Εισαγωγή
 - Δημιουργία αντικειμένων
 - Πρόσβαση στους πόρους του project
 - Events & Listeners

Οι εφαρμογές εκτελούνται σε «κουτιά»



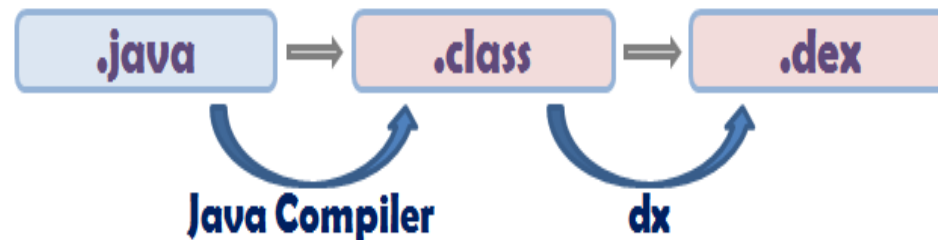
- Κάθε εφαρμογή εκτελείται στη δική της διεργασία Linux
 - Η διεργασία ξεκινάει τη στιγμή που εκτελείται η εφαρμογή
 - Χρησιμοποιούνται threads για λειτουργίες που απαιτούν χρόνο
- Κάθε διεργασία έχει το δικό της περιβάλλον εκτέλεσης
- Κάθε εφαρμογή έχει μοναδικό Linux ID
 - Τα δικαιώματα των αρχείων δεν επιτρέπουν πρόσβαση από άλλες εφαρμογές





Εκτέλεση εφαρμογών (1/2)

- Με χρήση του Android Runtime
 - Γίνεται εκτέλεση ενός Dalvik Executable (.dex)
 - Βελτιστοποίηση χρήσης μνήμης



- Βασίζεται στο Linux Kernel για
 - Threading
 - Διαχείριση μνήμης



Εκτέλεση εφαρμογών (2/2)

Υλοποίηση εφαρμογής σε Java

Compile σε Java

Μετατροπή σε Dalvik bytecode

Εκτέλεση στο ART

Linux OS



Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Android
 - Εκτέλεση εφαρμογών
 - Βασικά στοιχεία εφαρμογών
 - Εγκατάσταση περιβάλλοντος υλοποίησης εφαρμογών
 - Layouts
 - Βασικά widgets: TextView, EditText
 - Προσθήκη αντικειμένων
 - Προσθήκη λειτουργικότητας
 - Εισαγωγή
 - Δημιουργία αντικειμένων
 - Πρόσβαση στους πόρους του project
 - Events & Listeners



Βασικά στοιχεία εφαρμογών (1/2)

- Η επαναχρησιμοποίηση στοιχείων άλλων εφαρμογών δε γίνεται με χρήση όλης της εφαρμογής. Το Android εκκινεί την αντίστοιχη διεργασία της άλλης εφαρμογής όταν χρειάζεται
 - Οι εφαρμογές Android δεν έχουν μοναδικό σημείο εισόδου
 - Έχουν βασικά αντικείμενα που το σύστημα μπορεί να εκκινήσει όταν χρειαστεί
- 5 τύποι αντικειμένων
 - Activity
 - Service
 - Broadcast
 - Content
 - Intent



Βασικά στοιχεία εφαρμογών (2/2)

- Activity
 - Διεπαφή χρήστη με αντικείμενα – π.χ. λίστα με επιλογές μενού, φωτογραφίες με λεζάντα
 - Μπορεί να υπάρχει παραπάνω από 1 ανά εφαρμογή – π.χ. εφαρμογή μηνυμάτων (εύρεση ονομάτων, σύνταξη μηνύματος, κλπ)
- Services
 - Δεν υπάρχει διεπαφή με το χρήστη
 - Εκτελούνται στο background – π.χ. αναπαραγωγή μουσικής
- Broadcast (Receivers)
 - Λήψη και ανταπόκριση σε ειδοποιήσεις – π.χ. αλλαγή ζώνης ώρας
- Content (Providers)
 - Αποθήκευση και ανάκτηση δεδομένων για άλλες εφαρμογές – π.χ. επαφές
- Intents
 - Ασύγχρονα μηνύματα – π.χ. αίτηση μιας activity να παρουσιάσει μια εικόνα στο χρήστη



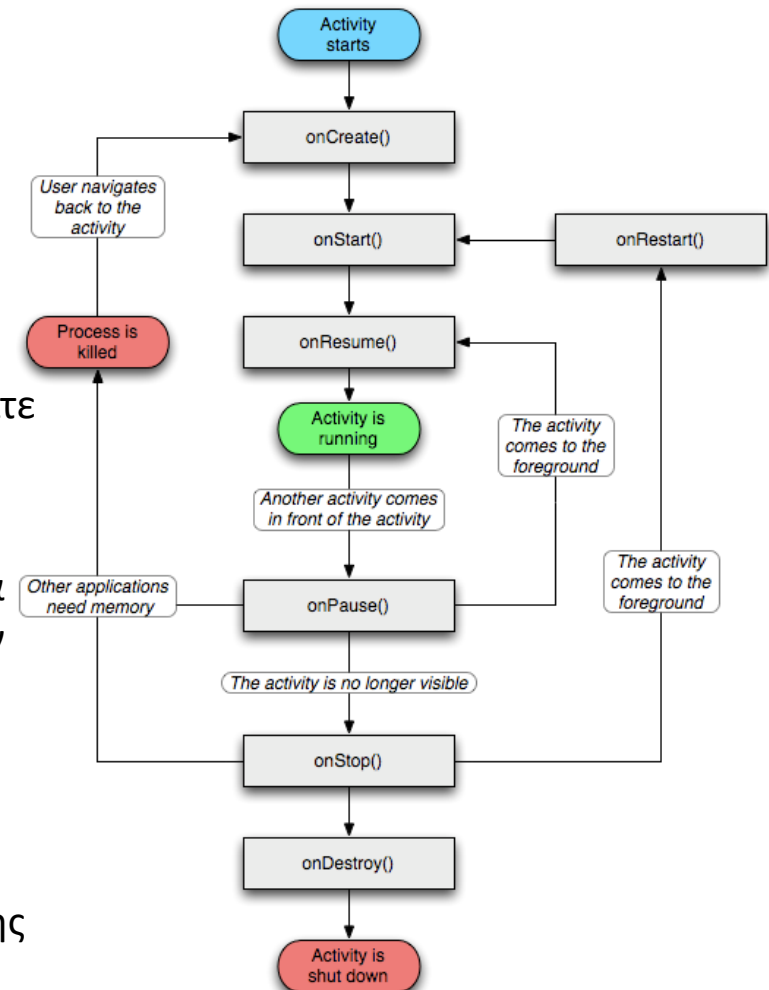
Activity

- Κάθε Activity θα πρέπει να δηλώνεται στο αρχείο `AndroidManifest.xml`
- Έχει ένα συγκεκριμένα παράθυρο (full screen ή μικρότερο) στο οποίο εμφανίζεται το περιεχόμενο
- Το περιεχόμενο παρέχεται ως μια ιεραρχία από `views` - αντικείμενα της κλάσης `View`
 - Η μέθοδος `Activity setContentView()` χρησιμοποιείται για να ορίσει μια συγκεκριμένη ιεραρχία από αντικείμενα `view`



Κύκλος ζωής ενός Activity (1/2)

- onCreate()
 - Καλείται για να δημιουργηθεί ένα activity. Ακολουθείται πάντα από την onStart()
- onStart()
 - Καλείται όταν ένα activity θέλουμε να γίνει ορατό στο χρήστη. Ακολουθείται είτε από την onResume() για να εκτελεστεί στο προσκήνιο είτε από την onStop() για να μπει στο παρασκήνιο
- onRestart()
 - Καλείται για ένα activity που έγινε stop ώστε να το επανεκκινήσει. Ακολουθείται πάντα από την onStart()
- onResume()
 - Καλείται όταν ένα activity θέλει να αλληλεπιδράσει με το χρήστη. Με αυτή τη μέθοδο το activity μεταφέρεται στην κορυφή της στοίβας και αναμένει input από το χρήστη. Ακολουθείται πάντα από την onPause()





Κύκλος ζωής ενός Activity (2/2)

□ onPause()

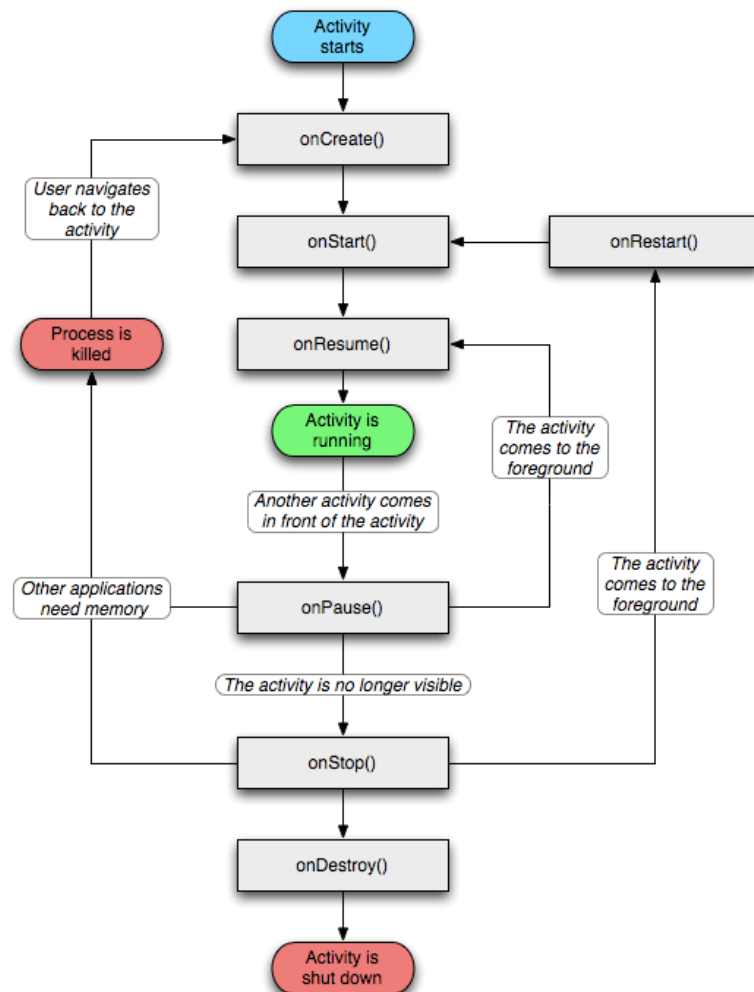
- Καλείται όταν το σύστημα επιθυμεί να συνεχίσει ένα προηγούμενο activity και πρέπει να κάνει pause το συγκεκριμένο activity

□ onStop()

- Καλείται στη περίπτωση που ένα δεύτερο activity έχει εκκινήσει και «επισκιάζει» το τρέχον activity, άρα πλέον δεν θα είναι ορατό στο χρήστη

□ onDestroy()

- Η τελευταία κλήση προτού τερματιστεί ένα activity. Καλείται είτε για να καταστραφεί το activity είτε όταν το σύστημα τερματίζει προσωρινά ένα activity για εξοικονόμηση χώρου





Intents

- Σκεφτείτε τα Intents ως ένα ρήμα, μια περιγραφή του τι θέλετε να γίνει
 - Π.χ. VIEW, CALL, PLAY κλπ..
- Το σύστημα αντιστοιχεί το Intent με κάποιο Activity που μπορεί να προσφέρει τη συγκεκριμένη υπηρεσία
- Τα Activities και τα IntentReceivers («ακούει» για κάποιο Intent) περιγράφουν τα Intents που μπορούν να εξυπηρετήσουν
- Αν υπάρχουν πολλοί Receivers για το ίδιο Intent, απαιτείται επιλογή από το χρήστη (π.χ. επιλογή εικόνας, αποστολή με mail / μήνυμα)
 - Διαφορετικά προκύπτει exception



Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Android
 - Εκτέλεση εφαρμογών
 - Βασικά στοιχεία εφαρμογών
 - Εγκατάσταση περιβάλλοντος υλοποίησης εφαρμογών
 - Layouts
 - Βασικά widgets: TextView, EditText
 - Προσθήκη αντικειμένων
 - Προσθήκη λειτουργικότητας
 - Εισαγωγή
 - Δημιουργία αντικειμένων
 - Πρόσβαση στους πόρους του project
 - Events & Listeners

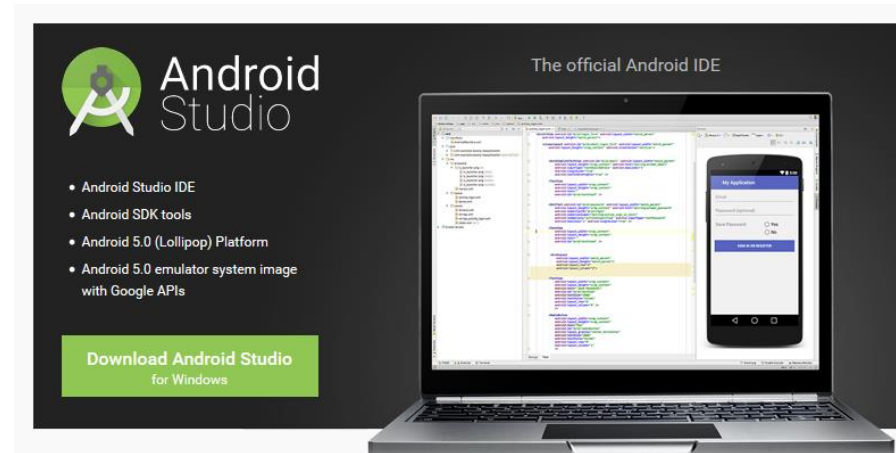
Εγκατάσταση περιβάλλοντος υλοποίησης εφαρμογών (1/3)



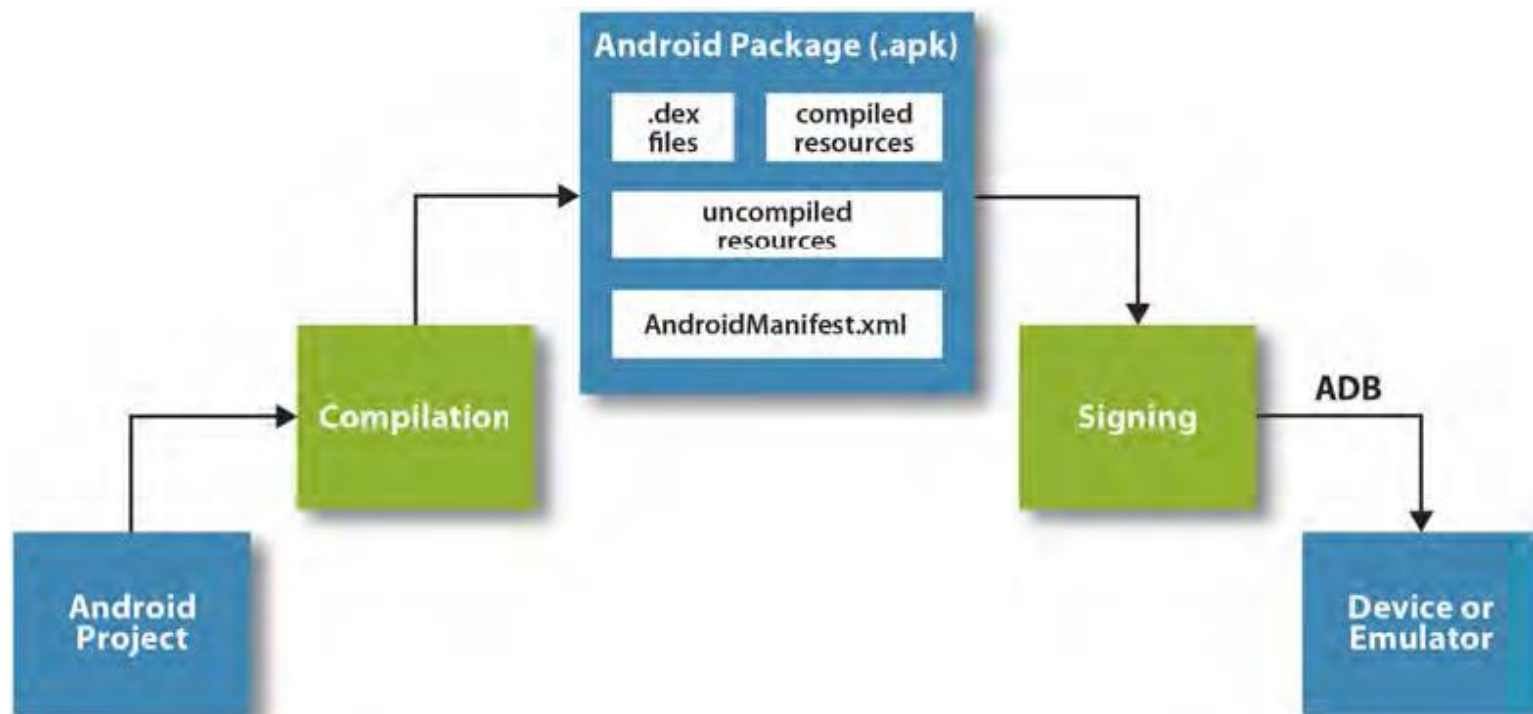
Android Studio

- SDK
- Platform
- Emulator
- Gradle

- ▣ Εγκατάσταση Android Package Files (.apk)
- ▣ Δημιουργία πολλαπλών apk για την ίδια εφαρμογή
- ▣ Παρέχει δυνατότητες debugging



Εγκατάσταση περιβάλλοντος υλοποίησης εφαρμογών (2/3)



Εγκατάσταση περιβάλλοντος υλοποίησης εφαρμογών (3/3)



□ Διαθέσιμο online

■ <http://developer.android.com/sdk/index.html>

□ JDK 7

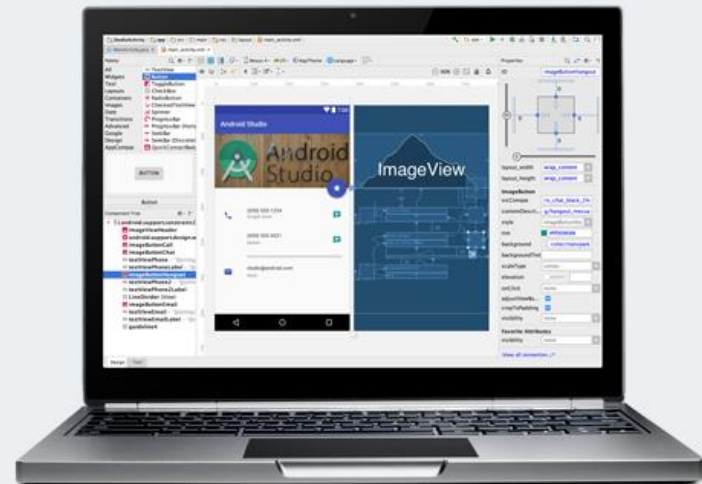
Android Studio

The Official IDE for Android

Android Studio provides the fastest tools for building apps on every type of Android device.

World-class code editing, debugging, performance tooling, a flexible build system, and an instant build/deploy system all allow you to focus on building unique and high quality apps.

DOWNLOAD ANDROID STUDIO
2.3.1 FOR WINDOWS (1,876 MB)





Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Android
 - Εκτέλεση εφαρμογών
 - Βασικά στοιχεία εφαρμογών
 - Εγκατάσταση περιβάλλοντος υλοποίησης εφαρμογών
 - Layouts
 - Βασικά widgets: TextView, EditText
 - Προσθήκη αντικειμένων
 - Προσθήκη λειτουργικότητας
 - Εισαγωγή
 - Δημιουργία αντικειμένων
 - Πρόσβαση στους πόρους του project
 - Events & Listeners



Layouts (1/3)

- Επιτρέπει τη διάταξη των στοιχείων στην οθόνη
 - **LinearLayout**: Διάταξη στοιχείων σε μια γραμμή (οριζόντια ή κάθετα – καθορίζεται από το `android:orientation`). Στην οριζόντια διάταξη υπάρχει μια γραμμή από Views, όλα του ίδιου ύψους («υψηλότερο» παιδί). Η ιδιότητα **layout_gravity** καθορίζει τη διάταξη βάσει του πατέρα, π.χ. `center_horizontal`, `center_vertical`, `top`, `left`, `right`, κλπ. Η ιδιότητα **gravity** καθορίζει την «εσωτερική διάταξη»
 - **RelativeLayout**: Οι τοποθεσίες των στοιχείων περιγράφονται σε σχέση με κάποιο άλλο στοιχείο ή με τον πατέρα
 - **FrameLayout**: Χρησιμοποιείται για να περιορίσει τη χρήση της οθόνης για την προβολή ενός μόνο στοιχείου. Μπορούν να προστεθούν πολλά παιδιά σε ένα **FrameLayout**, αλλά όλα θα βρίσκονται στο επάνω αριστερά μέρος της οθόνης
 - **TableLayout**: Τοποθέτηση στοιχείων σε γραμμές και στήλες
 - **TabHost**
 - ...



Layouts (2/3)

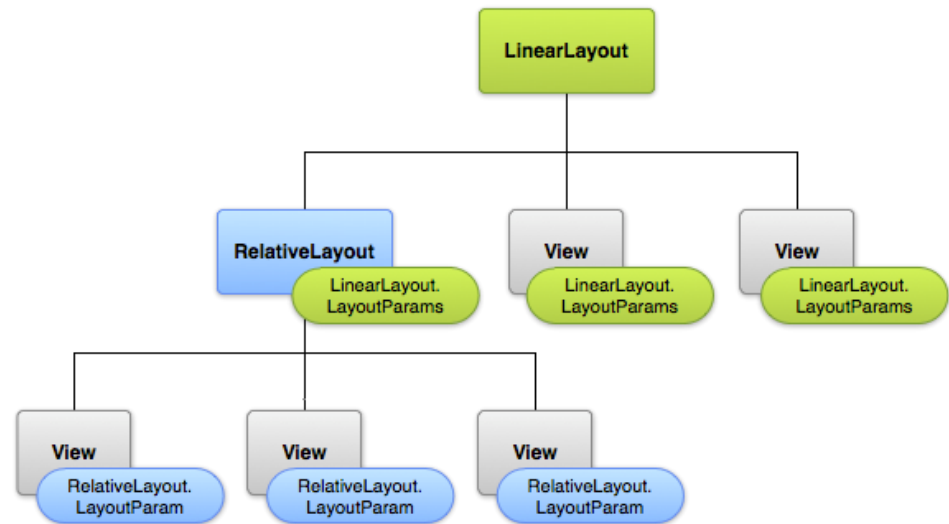
□ LayoutParams

- Πλάτος: `layout_width` και ύψος: `layout_height`
- Απαιτείται ο ορισμός τους!
- Μπορούν να χρησιμοποιηθούν
 - Συγκεκριμένες τιμές
 - Γενικά δεν προτείνεται!
 - Χρήση `dp` (density-independent) για `margins`, `padding`, κλπ
 - Χρήση `sp` (scale-independent) για κείμενα
 - «Σχετικές» τιμές
 - `wrap_content`: Προσαρμόζεται στα περιεχόμενα του (`autosize`)
 - `fill_parent` (από το API 8 και μετά ονομάζεται `match_parent`): Χρησιμοποιεί όλο το διαθέσιμο χώρο από τον πατέρα του



Layouts (3/3)

- Διαφορετικά layout μπορούν να συνδυαστούν
 - Κάθε παιδί θα πρέπει να ορίζει τα LayoutParams που είναι κατάλληλα για τον πατέρα του





Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Android
 - Εκτέλεση εφαρμογών
 - Βασικά στοιχεία εφαρμογών
 - Εγκατάσταση περιβάλλοντος υλοποίησης εφαρμογών
 - Layouts
 - Βασικά widgets: TextView, EditText
 - Προσθήκη αντικειμένων
 - Προσθήκη λειτουργικότητας
 - Εισαγωγή
 - Δημιουργία αντικειμένων
 - Πρόσβαση στους πόρους του project
 - Events & Listeners

Βασικά widgets: TextView, EditText

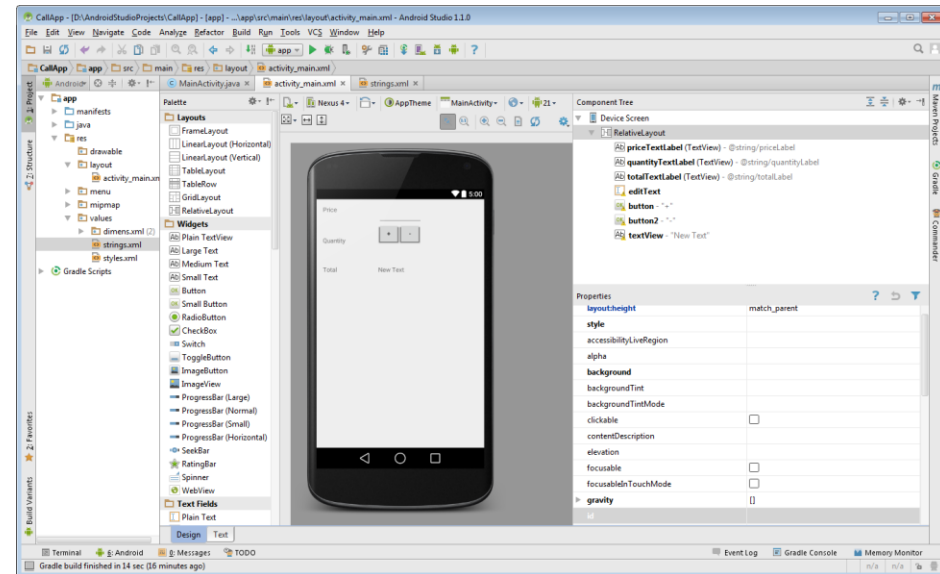


- TextView: προβολή αλφαριθμητικών χαρακτήρων
 - Βασικές ιδιότητες: ύψος, πλάτος, γραμματοσειρά, χρώμα κειμένου, κλπ
 - Επιπλέον ιδιότητες
 - autoLink: Εντοπίζει URLs στο κείμενο και τα μετατρέπει αυτόματα σε συνδέσμους
 - autoText: Εντοπίζει και διορθώνει απλά ορθογραφικά λάθη
 - editable: Δυνατότητα επεξεργασίας κειμένου
- EditText: προβολή και επεξεργασία αλφαριθμητικών χαρακτήρων (υποκλάση του TextView)



Προσθήκη αντικειμένων (1/2)

- ❑ Πλοήγηση στο res -> layout -> activity_main.xml
- ❑ Drag & Drop αντικείμενα στο Activity
- ❑ Ορισμός ID για κάθε αντικείμενο
 - Χρήση “label” για στατικό κείμενο (π.χ. priceLabel)
 - Χρήση “τύπου widget” για «δυναμικά» widgets (π.χ. priceEditText, quantityTextView, totalTextView)



```
MainActivity.java x activity_main.xml x strings.xml x
<?xml version="1.0" encoding="utf-8" android:namespace="http://schemas.android.com/apk/res/android"
xmlns:android="http://schemas.android.com/apk/res/android" android:layout_width="match_parent"
android:layout_height="match_parent" android:paddingLeft="16dp"
android:paddingRight="16dp" android:paddingTop="16dp" android:paddingBottom="16dp" tools:context=".MainActivity">

<TextView android:text="@string/priceLabel" android:layout_width="wrap_content"
android:layout_height="wrap_content" android:id="@+id/priceTextLabel" />

<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/quantityLabel"
android:id="@+id/quantityTextLabel"
android:layout_below="@+id/priceTextLabel"
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true"
android:layout_marginTop="54dp" />

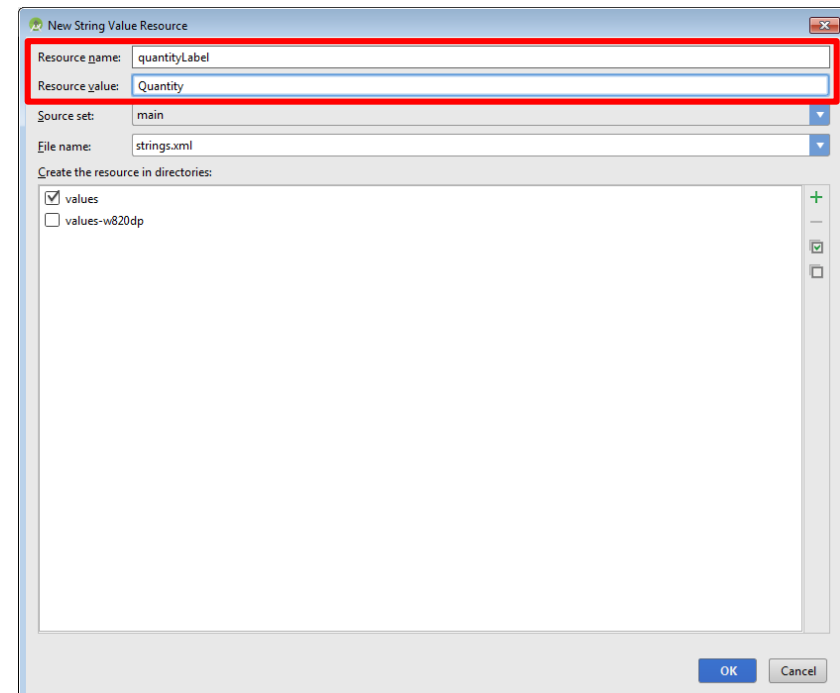
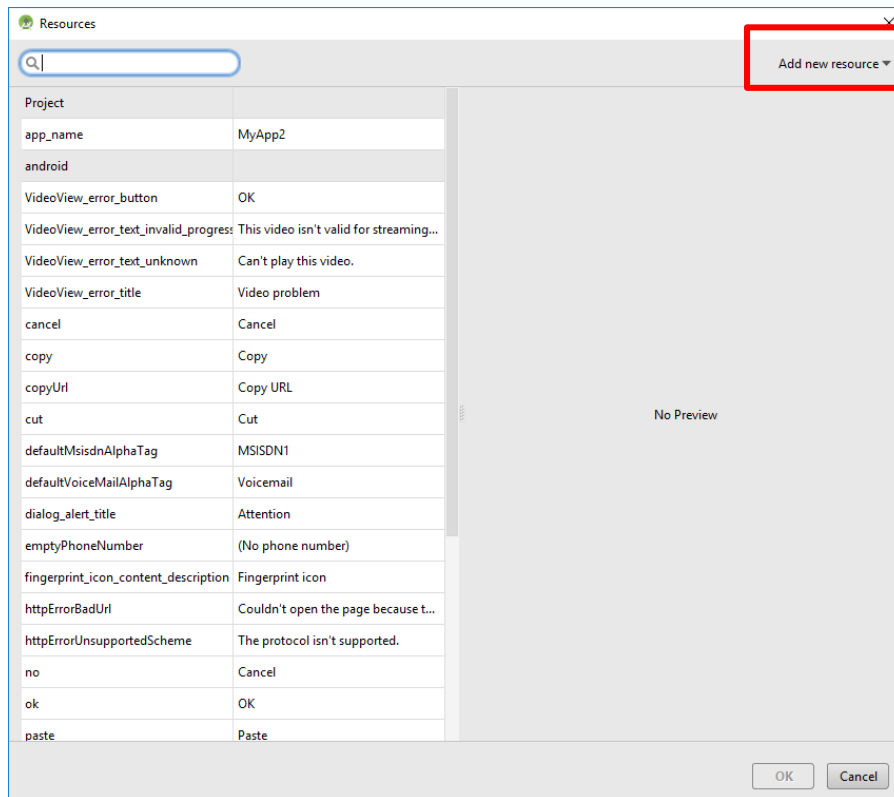
<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/totalLabel"
android:id="@+id/totalTextLabel"
android:layout_below="@+id/quantityTextLabel"
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true"
android:layout_marginTop="50dp" />

<EditText
android:layout_width="100dp"
android:layout_height="wrap_content"
android:id="@+id/editText"
android:layout_alignParentTop="true"
android:layout_centerHorizontal="true" />
```



Προσθήκη αντικειμένων (2/2)

- Ορισμός resource για κάθε αντικείμενο (text property)





Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Android
 - Εκτέλεση εφαρμογών
 - Βασικά στοιχεία εφαρμογών
 - Εγκατάσταση περιβάλλοντος υλοποίησης εφαρμογών
 - Layouts
 - Βασικά widgets: TextView, EditText
 - Προσθήκη αντικειμένων
 - Προσθήκη λειτουργικότητας
 - Εισαγωγή
 - Δημιουργία αντικειμένων
 - Πρόσβαση στους πόρους του project
 - Events & Listeners



Εισαγωγή

- java -> <package> -> MainActivity
- Βασική μέθοδος: onCreate()
 - Bundle savedInstanceState
 - Το bundle είναι ένα key value που αντιστοιχεί κλειδιά σε διάφορους τύπους. Επιτρέπει την μεταφορά πληροφορίας μεταξύ οθονών (διαφορετικών activities)
 - super.onCreate(savedInstanceState)
 - Απαιτείται για την εκτέλεση της εφαρμογής (μέσω της Activity class)!
 - setContentView(R.layout.activity_main)
 - Καθορίζει ποιο interface να εμφανίσει καθώς το activity δε γνωρίζει ποιο είναι το interface του



Δημιουργία αντικειμένων

- Αντιστοιχούν widgets σε συγκεκριμένα αντικείμενα, οι ιδιότητες των οποίων θα «τροποποιηθούν»
 - Π.χ. `private EditText priceEditText;`
 `private TextView quantityTextView;`
 `private Button buttonPlus;`
- Γιατί είναι global?

Πρόσβαση στους πόρους του project



- Με χρήση της μεθόδου `findViewById()`
 - Προσθήκη στην `onCreate()`
 - Επιστρέφει μια κλάση `View`
 - Πρέπει να γίνει `cast` στον κατάλληλο τύπο
 - Π.χ. `priceEditText = (EditText) findViewById(R.id.priceEditText);`
`quantityTextView = (TextView) findViewById(R.id.quantityTextView);`
`buttonPlus = (Button) findViewById(R.id.buttonPlus);`



Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Android
 - Εκτέλεση εφαρμογών
 - Βασικά στοιχεία εφαρμογών
 - Εγκατάσταση περιβάλλοντος υλοποίησης εφαρμογών
 - Layouts
 - Βασικά widgets: TextView, EditText
 - Προσθήκη αντικειμένων
 - Προσθήκη λειτουργικότητας
 - Εισαγωγή
 - Δημιουργία αντικειμένων
 - Πρόσβαση στους πόρους του project
 - Events & Listeners



Events & Listeners (1/3)

- Χρησιμοποιούνται για την υλοποίηση της σχετικής διαδραστικότητας
 - `onClick()` – `OnClickListener`: Όταν ο χρήστης ακουμπήσει οποιοδήποτε αντικείμενο στην οθόνη (ή εστιάσει με κουμπιά πλοήγησης)
 - `onLongClick()` – `OnLongClickListener`: Όταν ο χρήστης ακουμπήσει και κρατήσει το αντικείμενο (επίσης με παρατεταμένο `enter` ή `trackball` για 1 sec)
 - `onFocusChange()` – `OnFocusChangeListener`: Όταν ένα αντικείμενο λαμβάνει / χάνει focus
 - `onKey()` – `OnKeyListener`: Από hardware κουμπιά συσκευών
 - `onCreateContextMenu()` – `OnCreateContextMenuListener`
 - ...



Events & Listeners (2/3)

- Η κλάση υλοποιεί (implements) τον αντίστοιχο Listener
- Απαιτείται το σχετικό import και η υλοποίηση όλων των μεθόδων
- Π.χ.

... implements OnEditorActionListener

```
public boolean onEditorAction(TextView v, int actionId, KeyEvent event)
{
    quantityTextView.setText("7");
    ...
}
```



Input Method Editor (IME)

- Το soft keyboard επιτρέπει στους χρήστες την εισαγωγή περιεχομένου. Η ολοκλήρωση επεξεργασίας σχετίζεται με αντίστοιχες ενέργειες
 - IME_ACTION_DONE: Λειτουργία "done" – συνήθως κλείνει το πληκτρολόγιο
 - IME_ACTION_GO: Λειτουργία "go" – συνήθως οδηγεί το χρήστη στο συγκεκριμένο «στόχο»
 - IME_ACTION_NEXT: Λειτουργία "next" – συνήθως οδηγεί το χρήστη στο επόμενο πεδίο που περιμένει input
 - IME_ACTION_SEARCH
 - ...



Events & Listeners (3/3)

- Διασύνδεση listeners με τα αντίστοιχα widgets
- Π.χ. `priceEditText.setOnEditorActionListener(this);`
`buttonPlus.setOnClickListener(this);`