



Πανεπιστήμιο Πειραιώς Τμήμα Ψηφιακών Συστημάτων

Διαδικτυακά και Φορητά Πληροφοριακά Συστήματα (Δικτυακές Υπηρεσίες)

7^η Διάλεξη

Δημοσθένης Κυριαζής

Δευτέρα 23 Απριλίου 2018



Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Βασικά widgets
 - EditText
 - CheckBox
 - RadioButton
 - Spinner
- SeekBar
- Portrait / Landscape
- Μενού
- Προσθήκη Activity
- Intents
 - Εισαγωγή
 - Constructor & Μέθοδοι
 - Διαδικασία

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

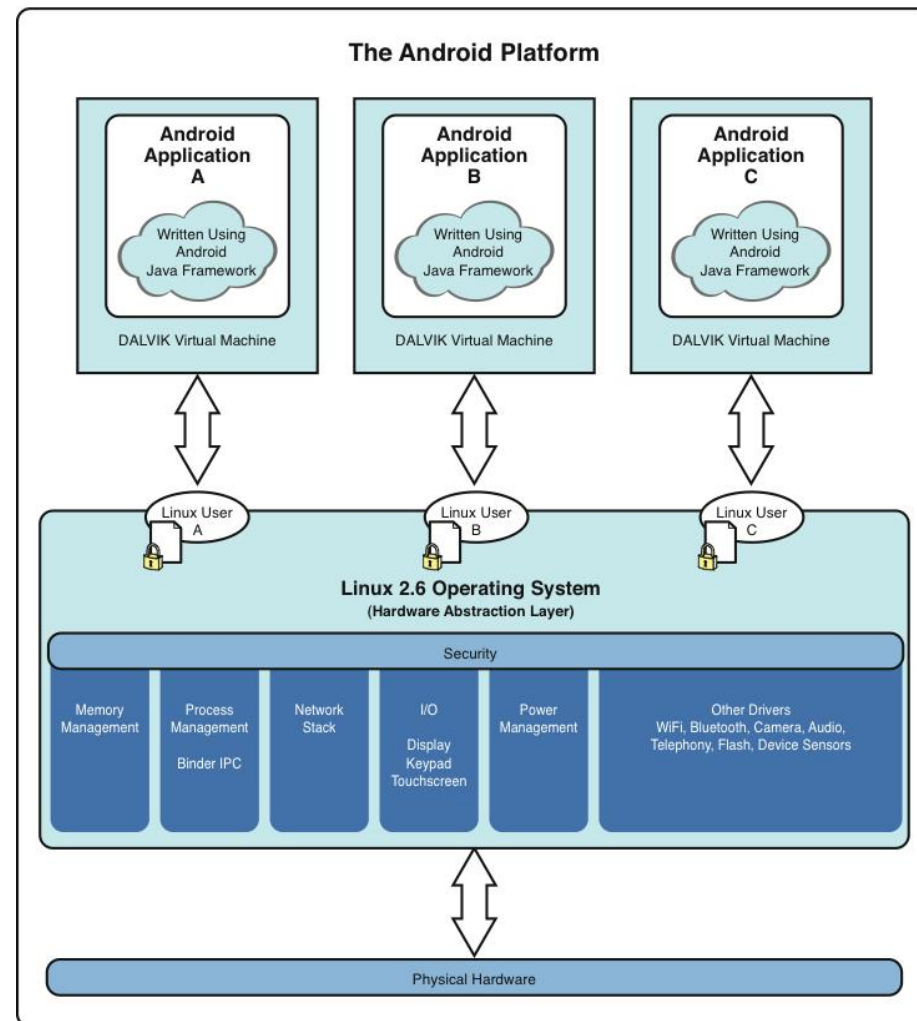
Audio
Drivers

Power
Management

Οι εφαρμογές εκτελούνται σε «κουτιά»



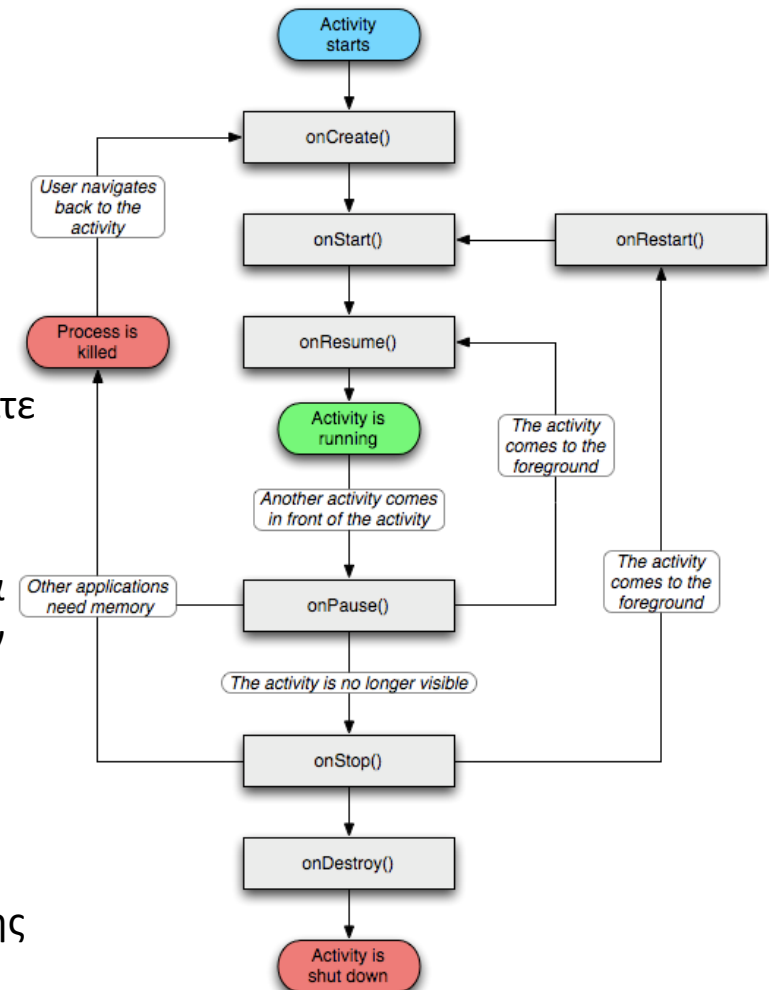
- Κάθε εφαρμογή εκτελείται στη δική της διεργασία Linux
 - Η διεργασία ξεκινάει τη στιγμή που εκτελείται η εφαρμογή
 - Χρησιμοποιούνται threads για λειτουργίες που απαιτούν χρόνο
- Κάθε διεργασία έχει το δικό της περιβάλλον εκτέλεσης
- Κάθε εφαρμογή έχει μοναδικό Linux ID
 - Τα δικαιώματα των αρχείων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





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()

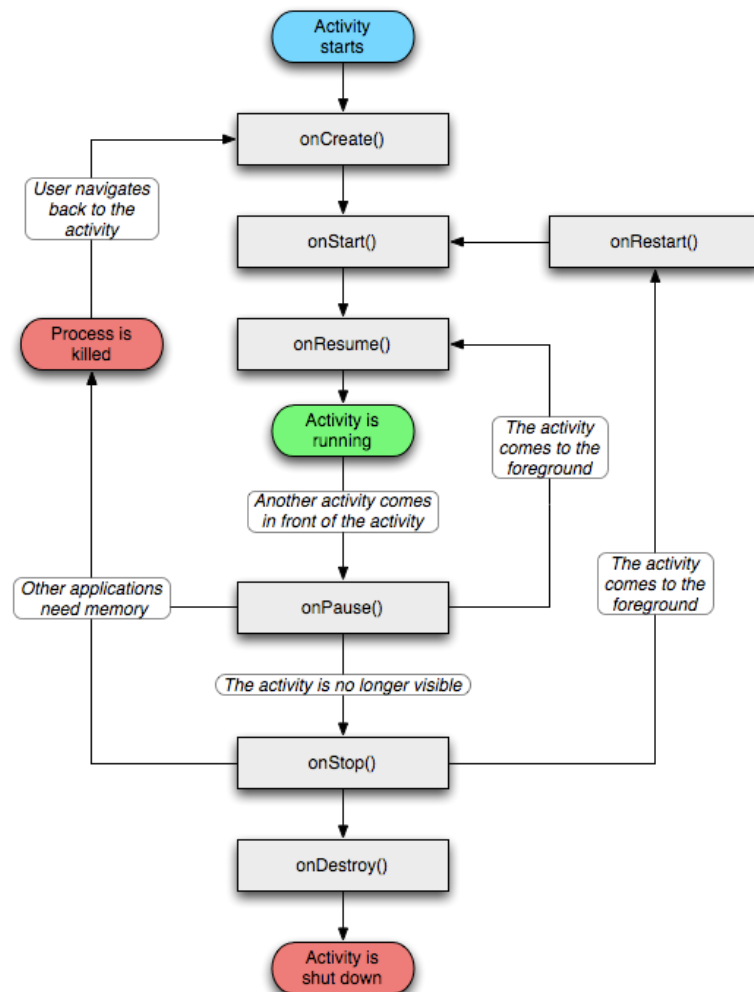
- Καλείται όταν το σύστημα επιθυμεί να συνεχίσει ένα προηγούμενο activity και πρέπει να κάνει pause το συγκεκριμένο activity

□ 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



Βασικά widgets: TextView, EditText

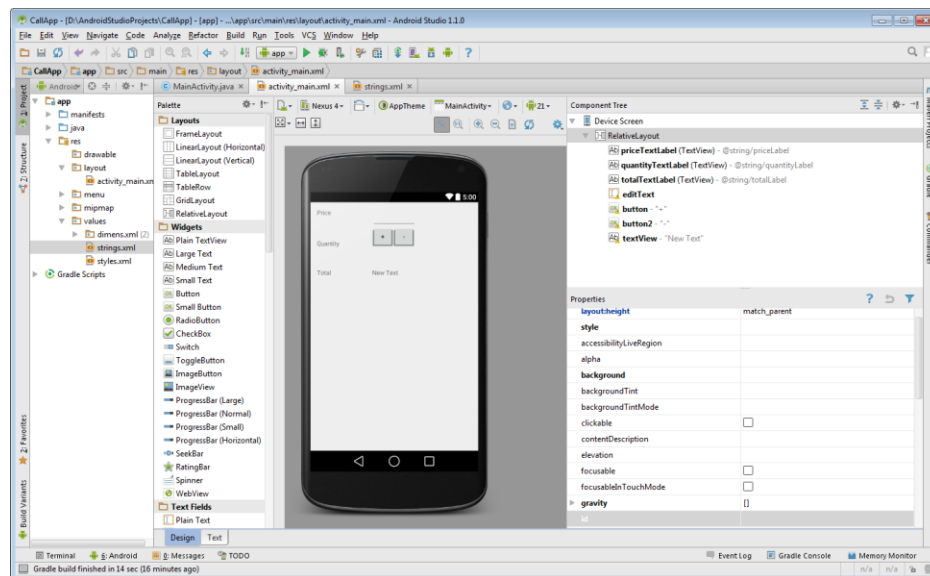


- TextView: προβολή αλφαριθμητικών χαρακτήρων
 - Βασικές ιδιότητες: ύψος, πλάτος, γραμματοσειρά, χρώμα κειμένου, κλπ
 - Επιπλέον ιδιότητες
 - autoLink: Εντοπίζει URLs στο κείμενο και τα μετατρέπει αυτόματα σε συνδέσμους
 - autoText: Εντοπίζει και διορθώνει απλά ορθογραφικά λάθη
 - editable: Δυνατότητα 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
relativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools" 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" />

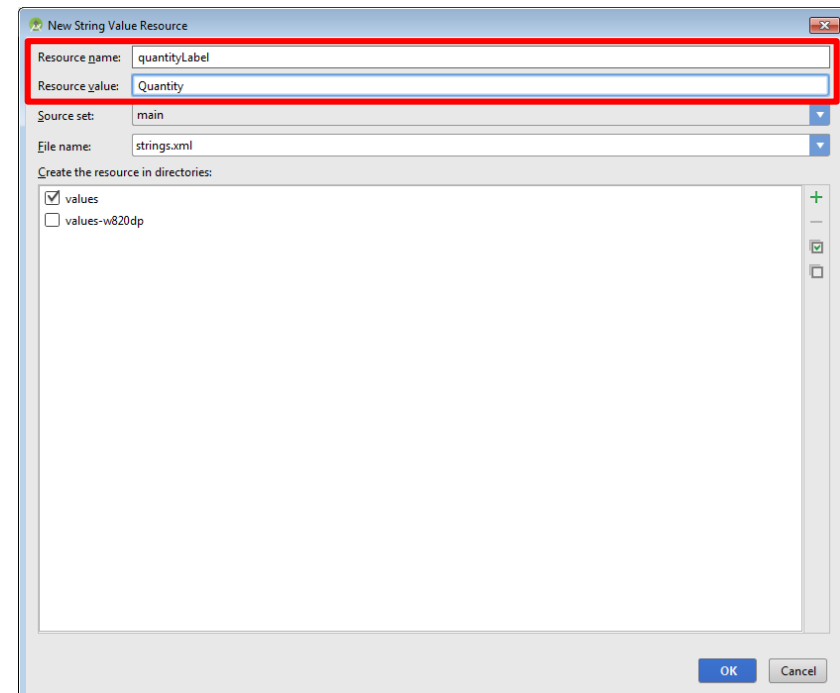
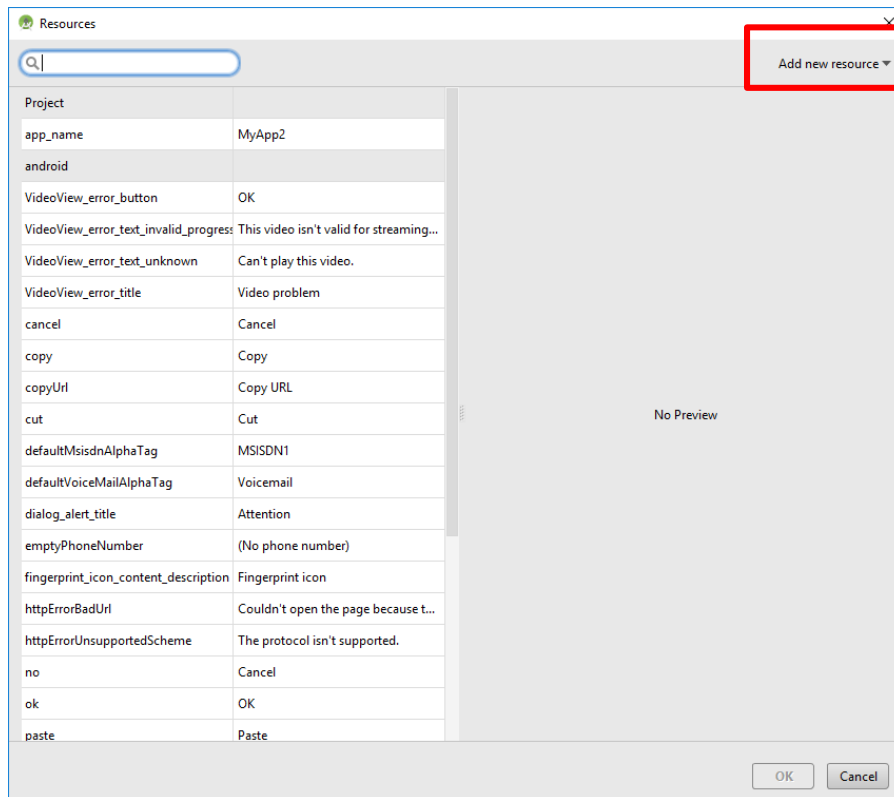
<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/totalLabel"
android:id="@+id/totalTextLabel"
android:layout_below="@+id/quantityTextLabel"
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true"
android:layout_marginTop="50dp"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

<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)





Δημιουργία αντικειμένων

- Αντιστοιχούν widgets σε συγκεκριμένα αντικείμενα, οι ιδιότητες των οποίων θα «τροποποιηθούν»
 - Π.χ.

```
private EditText priceEditText;  
private TextView quantityTextView;  
private Button buttonPlus;
```
- Global ώστε να είναι εφικτή η πρόσβαση από όλες τις μεθόδους και επομένως και από τις μεθόδους των Listeners

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);`



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");
    ...
}
```



Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Βασικά widgets
 - EditText
 - CheckBox
 - RadioButton
 - Spinner
- SeekBar
- Portrait / Landscape
- Μενού
- Προσθήκη Activity
- Intents
 - Εισαγωγή
 - Constructor & Μέθοδοι
 - Διαδικασία



Λήψη κειμένου

- Με χρήση `getText()`

```
public void calc(){  
    priceString = priceEditText.getText().toString();  
    int price;  
    if (priceString.equals("")){  
        price = 0;  
    }  
    else{  
        price = Integer.parseInt(priceString);  
    }  
    int total = price * quantity;  
    totalTextView.setText(Integer.toString(total));  
}
```




Αποθήκευση «κατάστασης» (1/2)

- Για μικρό όγκο δεδομένων (όπως η «κατάσταση» μιας εφαρμογής) μπορεί να γίνει χρήση των APIs του SharedPreferences
- Ένα αντικείμενο SharedPreferences σχετίζεται με ένα αρχείο που περιλαμβάνει ζεύγη key-value και συνοδεύεται με μεθόδους για την ανάγνωση και προσθήκη περιεχομένων στο αρχείο
 - Η εγγραφή γίνεται μέσω SharedPreferences.Editor με κλήση της edit() για το αντικείμενο SharedPreferences
 - Προσθήκη δεδομένων γίνεται με μεθόδους put (π.χ. putInt(), putString(), κλπ) και ακολουθεί η commit() για την αποθήκευση των αλλαγών
 - Ανάκτηση δεδομένων γίνεται με get (π.χ. getInt(), getString(), κλπ)



Αποθήκευση «κατάστασης» (2/2)

- Μεταβλητή αντικειμένου
 - `private SharedPreferences savedValues;`
- Λήψη αντικειμένου στην `onCreate()`
 - `savedValues = getSharedPreferences("SavedValues", MODE_PRIVATE);`
- Χρήση Editor για την προσθήκη περιεχομένου στην `onPause()`
 - `Editor editor = savedValues.edit();`
 - `editor.putString("price", priceString);`
 - `editor.putInt("quantity", quantity);`
 - `editor.commit();`
 - `super.onPause();`
- Ανάκτηση περιεχομένου στην `onResume()`
 - `super.onResume();`
 - `priceString = savedValues.getString("price", "");`
 - `quantity = savedValues.getInt("quantity", 0);`
 - `priceEditText.setText(priceString);`

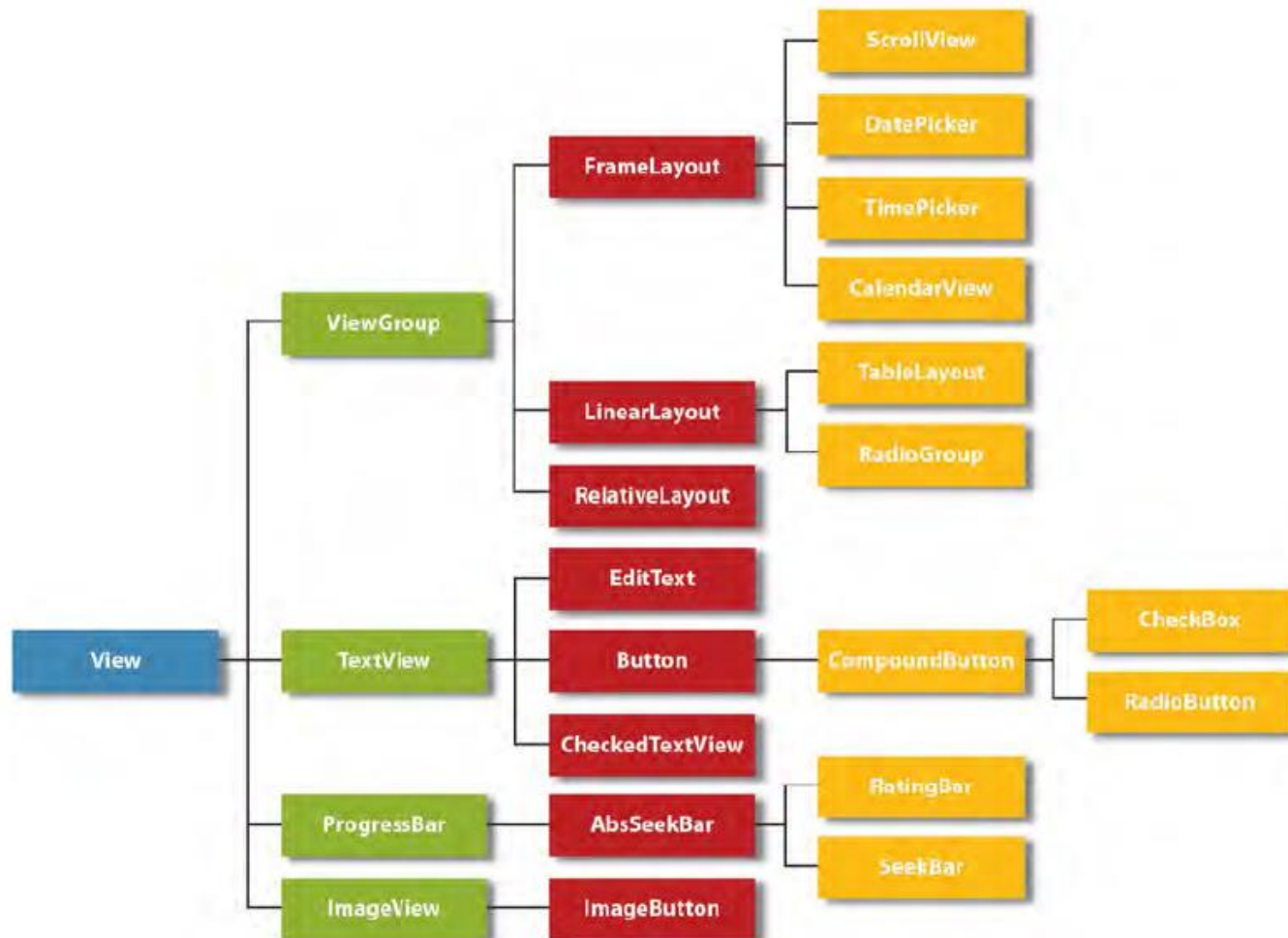


Βασικά widgets

Widget	Περιγραφή
TextView	Παρουσίαση κειμένου (ονομάζεται επίσης label)
EditText	Επεξεργασία κειμένου, mail, password, κλπ (ονομάζεται επίσης editable text view, text box, text field)
Button	Κουμπί
CheckBox	Πολλαπλή επιλογή
RadioButton	Επιλογή ενός αντικειμένου (χρήση και του RadioGroup widget)
Spinner	Επιλογή από λίστα (ονομάζεται επίσης drop-down list)
ProgressBar	Παρουσίαση προόδου διαδικασίας
SeekBar	Επιλογή τιμής μέσω «μπάρας»
RatingBar	Βαθμολόγηση μέσω επιλογής ενός / περισσοτέρων αστεριών
ImageView	Προβολή εικόνας
ImageButton	Κουμπί με εικόνα
DatePicker	Επιλογή ημερομηνίας
TimePicker	Επιλογή ώρας
CalendarView	Επιλογή ημερομηνίας
ScrollView	Κάθετο scroll bar εάν το layout δε χωράει στην οθόνη κατακόρυφα



Κλάσεις widgets και layouts





EditText

□ Βασικά attributes

- `inputType`: Τύπος δεδομένων που εισάγονται (καθορίζει την εμφάνιση του soft keyboard)
 - `text`, `textPersonName`, `textEmailAddress`, `textPassword`, `number`, `numberDecimal`, `phone`, `date`, `time`, κλπ
- `lines`: Αριθμός γραμμών (default: 1)



Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Βασικά widgets
 - EditText
 - CheckBox
 - RadioButton
 - Spinner
- SeekBar
- Portrait / Landscape
- Μενού
- Προσθήκη Activity
- Intents
 - Εισαγωγή
 - Constructor & Μέθοδοι
 - Διαδικασία



CheckBox (1/2)

- Στόχος
 - Πολλαπλή επιλογή αντικειμένων
- Βασικό attribute
 - `android:checked="true"`
 - ▣ επιλογή / απο-επιλογή
- Βασικές μέθοδοι
 - `setChecked(boolean)`: επιλογή / απο-επιλογή αντικειμένου
 - `isChecked()`: επιστρέφει `boolean` για το αν το αντικείμενο είναι επιλεγμένο



CheckBox (2/2)

□ Listener

■ OnCheckedChangeListener

■ Π.χ. @Override

```
public void onCheckedChanged(CompoundButton widget, boolean isChecked) {  
    switch (widget.getId())  
    {  
        case R.id.choice1CheckBox:  
            if (isChecked) {  
                // code  
            }  
        case R.id.choice2CheckBox:  
            if (isChecked) {  
                // code  
            }  
    }  
}
```



RadioButton (1/2)

- Στόχος
 - Επιλογή ενός μόνο αντικειμένου
- Attribute και Μέθοδοι
 - Όπως για το CheckBox
- Βασικές διαφορές
 - Χρήση RadioGroup (για τη διασφάλιση επιλογής ενός μόνο αντικειμένου)
 - Ορισμός προσανατολισμού
 - Π.χ.

```
<RadioGroup  
    android:id="@+id/radioGroup1"  
    ...  
    android:orientation="horizontal" >
```



```
<RadioButton  
    android:id="@+id/radioButtonYes"  
    ...  
    android:text="@string/radioButton_Yes" />
```



Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Βασικά widgets
 - EditText
 - CheckBox
 - RadioButton
 - Spinner
- SeekBar
- Portrait / Landscape
- Μενού
- Προσθήκη Activity
- Intents
 - Εισαγωγή
 - Constructor & Μέθοδοι
 - Διαδικασία



Spinner (1/3)

- Στόχος
 - Επιλογή αντικειμένων από λίστα
- Σημαντικά σημεία
 - Οι ιδιότητες καθορίζονται κυρίως με κώδικα
 - Χρησιμοποιείται το array adapter για την παροχή της λίστας των αντικειμένων στον spinner
 - Γίνεται εύκολα με προσθήκη array στο strings.xml
- Βασικές μέθοδοι ArrayAdapter
 - `createFromResource(context, arrayID, layoutID)`: Δημιουργία array adapter από το array και το αντίστοιχο layout (`android.R.layout.simple_spinner_item`)
 - `setDropDownViewResource()`: Καθορισμός layout για τα αντικείμενα της λίστας (`android.R.layout.simple_spinner_dropdown_item`)



Spinner (2/3)

- Βασικές μέθοδοι Spinner
 - `setAdapter(arrayAdapter)`: Καθορισμός του array adapter για το spinner
 - `setSelection(index)`: Επιλογή αντικειμένου
 - `getSelectedItemPosition()`: Επιστρέφει ακέραιο της επιλογής (αρχική τιμή 0)
 - `getSelectedItem()`: Επιστρέφει το αντικείμενο της επιλογής



Spinner (3/3)

□ Listener

■ OnItemSelectedListener

- ▣ `onItemSelected(AdapterView<?> parent, View view, int position, long id)`
 - `AdapterView` (view που τα περιεχόμενα καθορίζονται από adapter) για τη λίστα
 - `View` για το αντικείμενο στο οποίο έγινε κλικ
 - Θέση του αντικειμένου στο `Adapter`
 - ID του επιλεγμένου αντικειμένου
- ▣ `onNothingSelected(AdapterView<?> arg0)`
 - Εκτελείται όταν η επιλογή «εξαφανίζεται» (π.χ. η λίστα είναι άδεια)



Προσθήκη επιλογών spinner

- Μετάβαση στο strings.xml
- Χρήση String Array
- Προσθήκη ονόματος και τιμών για το array
 - Π.χ. `<string-array name="disc_array">`
 `<item>10%</item>`
 `<item>20%</item>`
 `<item>50%</item>`
 `</string-array>`
- «Σύνδεση» spinner με string array (μέσω του layout)
 - Π.χ. `<Spinner`
 `android:layout_width="match_parent"`
 `android:layout_height="match_parent"`
 `android:id="@+id/spinner"`
 `android:layout_column="1"`
 `android:entries="@array/disc_array" />`

Προσθήκη μεταβλητών για τα widgets



□ Κλάση

- `private RadioGroup radioGroup;`
- `private EditText priceEditText;`
- `private TextView finalAmountTextView;`
- `private TextView discPerc;`
- `private Spinner discSpin;`



Προσθήκη references στα widgets

□ onCreate

- `radioGroup = (RadioGroup) findViewById(R.id.radioGroup1);`
- `priceEditText = (EditText) findViewById(R.id.priceEditText);`
- `finalAmountTextView = (TextView) findViewById(R.id.finalAmountEditText);`
- `discPerc = (TextView) findViewById(R.id.discPercLabel);`
- `discSpin = (Spinner) findViewById(R.id.discountSpinner);`



Ορισμός listeners

□ Κλάση

- ... implements `OnCheckedChangeListener`,
`OnItemSelectedListener`

□ onCreate

- `radioGroup.setOnCheckedChangeListener(this);`
- `discSpin.setOnItemSelectedListener(this);`



Εμφάνιση / απόκρυψη widgets

- Γίνεται με κώδικα
- Μέθοδος της κλάσης View
 - `setVisibility(View.CONSTANT)`
 - `VISIBLE`: Εμφάνιση
 - `INVISIBLE`: Απόκρυψη
 - `GONE`: Απόκρυψη και απελευθέρωση χώρου στην οθόνη



Υλοποίηση listeners (1/2)

```
public void onCheckedChanged(RadioGroup arg0, int checkedId) {  
    switch (checkedId)  
    {  
        case R.id.radioButtonYes:  
            discPerc.setVisibility(View.VISIBLE);  
            discSpin.setVisibility(View.VISIBLE);  
            break;  
        case R.id.radioButtonNo:  
            discPerc.setVisibility(View.GONE);  
            discSpin.setVisibility(View.GONE);  
            break;  
    }  
}
```



Υλοποίηση listeners (2/2)

```
public void onItemSelected(AdapterView<?> parent, View view, int position,  
long id) {  
    if (position==0)  
    {  
        discount = 0.1f;  
        calc();  
    }  
}
```



Συνάρτηση υπολογισμού

```
public void calc() {  
    priceString = priceEditText.getText().toString();  
    int price;  
    if (priceString.equals(""))  
    {  
        price=0;  
    }  
    else {  
        price = Integer.parseInt(priceString);  
    }  
    float total = price * (1 - discount);  
    finalAmountTextView.setText(Float.toString(total));  
}
```




Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Βασικά widgets
 - EditText
 - CheckBox
 - RadioButton
 - Spinner
- SeekBar
- Portrait / Landscape
- Μενού
- Προσθήκη Activity
- Intents
 - Εισαγωγή
 - Constructor & Μέθοδοι
 - Διαδικασία



SeekBar (1/3)

- Στόχος
 - Προβολή «μπάρας» για επιλογή από χρήστη
- Βασικά attributes
 - max: Μέγιστη τιμή (default 100)
 - progress: Τιμή (συνήθως χρησιμοποιείται για αρχικοποίηση)
- Βασικές μέθοδοι
 - setProgress(int): Ορισμός τιμής
 - getProgress(): Λήψη τρέχουσας τιμής



SeekBar (2/3)

□ Listener

■ OnSeekBarChange

- ▣ `public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser)`
 - Σε κάθε αλλαγή από το χρήστη
- ▣ `public void onStartTrackingTouch(SeekBar seekBar)`
 - Στην έναρξη των αλλαγών από το χρήστη
- ▣ `public void onStopTrackingTouch(SeekBar seekBar)`
 - Μετά την ολοκλήρωση των αλλαγών από το χρήστη



SeekBar (3/3)

- ❑ Μεταβλητές για τα widgets
 - `private SeekBar seekBar;`
 - `private TextView seekBarTextView;`
- ❑ References στα widgets
 - `seekBar = (SeekBar) findViewById(R.id.seekBar1);`
 - `seekBarTextView = (TextView) findViewById(R.id.seekBarTextView);`
- ❑ Ορισμός listener
 - `seekBar.setOnSeekBarChangeListener(this);`
- ❑ Υλοποίηση listener
 - `@Override`
`public void onProgressChanged(SearchBar seekBar, int progress,`
`boolean fromUser) {`
`seekBarTextView.setText(progress + "%");`
`}`



Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Βασικά widgets
 - EditText
 - CheckBox
 - RadioButton
 - Spinner
- SeekBar
- Portrait / Landscape
- Μενού
- Προσθήκη Activity
- Intents
 - Εισαγωγή
 - Constructor & Μέθοδοι
 - Διαδικασία



Portrait / Landscape

- Κάθε Activity διαχειρίζεται αλλαγές στον προσανατολισμό της οθόνης
- Με αλλαγή στο `AndroidManifest.xml` μπορεί να οριστεί «συγκεκριμένος» προσανατολισμός
 - Π.χ. `android:screenOrientation="portrait"`
- Για διαφορετικά layout ανάλογα με τον προσανατολισμό
 - Δημιουργία ενός καταλόγου `res/layout-land`
 - Αντιγραφή του `xml` σε αυτόν τον φάκελο
 - Τροποποίηση του



Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Βασικά widgets
 - EditText
 - CheckBox
 - RadioButton
 - Spinner
- SeekBar
- Portrait / Landscape
- Μενού
- Προσθήκη Activity
- Intents
 - Εισαγωγή
 - Constructor & Μέθοδοι
 - Διαδικασία



Εισαγωγή

- Ένα menu μπορεί να περιλαμβάνει 1 ή περισσότερα items
- Οι επιλογές εμφανίζονται είτε με εκκίνηση από τη συσκευή (κουμπί μενού) είτε μέσω της διεπαφής (παρατεταμένο ή απλό κλικ)
- Εκδόσεις πριν το Android 3.0 (API 11) εμφάνιζαν το μενού (options menu) στο κάτω μέρος της οθόνης (6 αντικείμενα)
- Μετά το Android 3.0 (API 11) υπάρχει η δυνατότητα εμφάνισης επιλογών ως action items



Τύποι

- ❑ Options menu: Τυπικό μενού με τις επιλογές να εμφανίζονται ως
 - Κείμενο (by default στη λίστα του μενού)
 - Εικόνες (by default στο Action Bar)
- ❑ Floating context menu: Προκύπτει από παρατεταμένο κλικ σε κάποιο widget
 - Μετά το Android 3.0 (API 11) είναι προτιμότερη η χρήση Contextual Action Menu καθώς εμφανίζει επιλογές που συνδέονται με το επιλεγμένο αντικείμενο
- ❑ Popup menu: Προκύπτει από απλό κλικ σε widget ή action item



Attributes επιλογών μενού

- title: το κείμενο της επιλογής
- icon: το εικονίδιο της επιλογής
 - Συγκεκριμένα εικονίδια στο φάκελο drawable-xhdpi
- showAsAction: καθορίζει εάν η επιλογή θα εμφανίζεται στο action bar. Λαμβάνει τιμές always, never, ifRoom
 - `app:showAsAction="never"`
- orderInCategory: καθορίζει έναν ακέραιο αριθμό για τη σειρά των επιλογών (ο μικρότερος αριθμός εμφανίζεται πρώτος)
 - `android:orderInCategory="100"`



Προσθήκη μενού

- Επιλογή Blank Activity
- Υπάρχει το αρχικό μενού στο φάκελο res/menu
- Αναφέρεται σε ένα XML αρχείο
 - menu_main.xml
 - Περιλαμβάνει τις επιλογές του μενού
- Επιπλέον XML αρχεία μπορούν να προστεθούν στο συγκεκριμένο φάκελο για άλλες activities που επίσης έχουν μενού



Προσθήκη επιλογών (1/2)

- Το Group χρησιμοποιείται για την ομαδοποίηση επιλογών (δεν υπάρχει διαφορά στην εμφάνιση του μενού)
- Επιλογή Item και εισαγωγή στοιχείων
- Strings.xml

```
<resources>
```

```
    <string name="app_name">App1205</string>
```

```
    <string name="hello_world">Hello world!</string>
```

```
    <string name="action_settings">Settings</string>
```

```
    <string name="firstItem_mitem">First Item</string>
```

```
    <string name="secondItem_mitem">Second Item</string>
```

```
</resources>
```



Προσθήκη επιλογών (2/2)

□ menu_main.xml

```
<menu 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" tools:context=".MainActivity">
  <item android:id="@+id/action_settings" android:title="@string/action_settings"
    android:orderInCategory="100" app:showAsAction="never" />
```

<item

```
  android:id="@+id/item1"
  android:title="@string/firstItem_mitem"
  app:showAsAction="always"></item>
```

<item

```
  android:id="@+id/item2"
  android:title="@string/secondItem_mitem"
  app:showAsAction="always"></item>
```

</menu>



Εμφάνιση μενού

- Με χρήση (override) της μεθόδου `onCreateOptionsMenu()`
 - `getMenuInflater().inflate(R.menu.activity_test, menu);`
 - ▣ «Εισαγωγή» στο αντικείμενο `MenuInflater` του μενού και ονομασία «menu»
 - `return true;`
- Το Android καλεί τη μέθοδο μόνο την πρώτη φορά για την εμφάνιση του μενού
 - Για αλλαγές στο μενού γίνεται χρήση της μεθόδου `onPrepareOptionsMenu()`



Διαχείριση events

- Δημιουργούνται όταν ο χρήστης επιλέγει κάποιο αντικείμενο του μενού
- Χρήση της μεθόδου `onOptionsItemSelected()`

- Π.χ. `switch (id) {`

- `case R.id.item1:`

- `Toast.makeText(this, "Item1 selected",
Toast.LENGTH_SHORT).show();`

- `return true;`

- `case R.id.item2:`

- `Toast.makeText(this, "Item2 selected",
Toast.LENGTH_SHORT).show();`

- `return true;`

- `default:`

- `return super.onOptionsItemSelected(item);`

- `}`



Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Βασικά widgets
 - EditText
 - CheckBox
 - RadioButton
 - Spinner
- SeekBar
- Portrait / Landscape
- Μενού
- Προσθήκη Activity
- Intents
 - Εισαγωγή
 - Constructor & Μέθοδοι
 - Διαδικασία



Προσθήκη Activity (1/2)

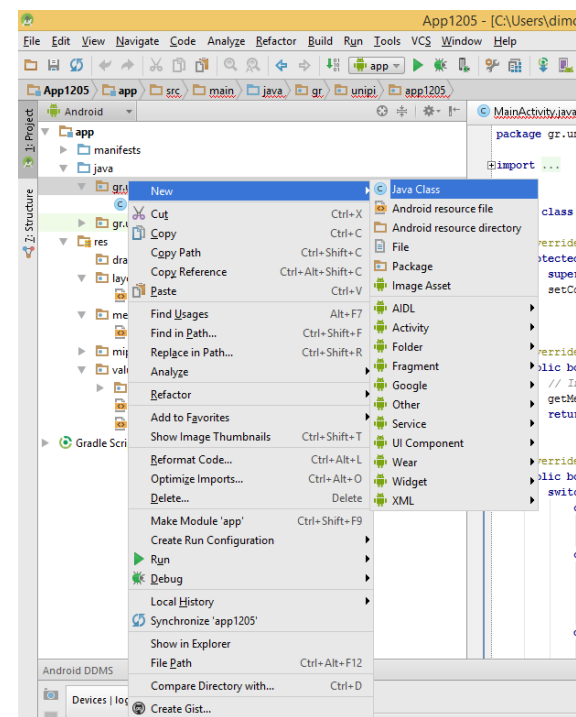
□ Δεξί κλικ στο όνομα του πακέτου -> New

■ -> Java Class

- ... extends Activity
- Προσθήκη layout
- Προσθήκη onCreate()
- Προσθήκη στο AndroidManifest!

■ -> Activity -> Empty Activity

- Προσθέτει το Android Studio το σχετικό κώδικα και το layout
- Π.χ. SecondActivity





Προσθήκη Activity (2/2)

□ Θα πρέπει να δηλωθεί στο **AndroidManifest**

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="gr.unipi.app1205" >
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
            </intent-filter>
            <category android:name="android.intent.category.LAUNCHER" />
        </activity>
        <activity
            android:name=".SecondActivity">
        </activity>
    </application>
</manifest>
```



Εκκίνηση νέου activity

- Επιτυγχάνεται με χρήση intent
 - Αντικείμενο για περιγραφή προς εκτέλεση λειτουργίας
- Δημιουργία αντικειμένου intent
 - `Intent newAct= new Intent(getApplicationContext(), SecondActivity.class);`
- Εκκίνηση activity
 - `startActivity(newAct);`



Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Βασικά widgets
 - EditText
 - CheckBox
 - RadioButton
 - Spinner
- SeekBar
- Portrait / Landscape
- Μενού
- Προσθήκη Activity
- Intents
 - Εισαγωγή
 - Constructor & Μέθοδοι
 - Διαδικασία



Εισαγωγή

- Περιγράφει τη λειτουργία που θα εκτελεστεί
 - Explicit intent: όταν το intent καθορίζει το αντικείμενο στο οποίο θα εκτελεστεί η λειτουργία. Χρησιμοποιείται για τη «μεταφορά» δεδομένων μεταξύ διαφορετικών activities
 - Π.χ. `Intent intent = new Intent(this, OneActivity.class)`
 - Implicit intent: όταν το intent καθορίζει την ενέργεια που θα εκτελεστεί
 - Π.χ. `Intent viewIntent = new Intent (Intent.ACTION_VIEW, uri)`



Constructor & Μέθοδοι (1/2)

□ Explicit Intent

■ Constructor

- Intent(context, class): Δημιουργία intent για τη συγκεκριμένη κλάση

■ Μέθοδοι

- putExtra(name, value): Αποθήκευση συγκεκριμένης τιμής – Π.χ. `Intent.putExtra("title", "android lecture");`
- getStringExtra(name): Ανάκτηση τιμής
- getIntExtra(name, default), getBooleanExtra ..



Constructor & Μέθοδοι (2/2)

□ Implicit Intent

■ Constructor

- Intent (action, uri): Δημιουργία intent για τη συγκεκριμένη ενέργεια (σταθερά) και τα συγκεκριμένα δεδομένα (uri)

■ Μέθοδοι: Αναφέρονται στην ενέργεια κατά τη δημιουργία του intent

- ACTION_VIEW: Προβολή στο browser
- ACTION_CALL: Κλήση αριθμού
- ACTION_DIAL
- ...

□ Για κάθε intent απαιτείται η εκτέλεση της σχετικής activity με χρήση της startActivity()

- Π.χ. startActivity(intent);



Σημερινή διάλεξη

- Σύνοψη προηγούμενης διάλεξης
- Βασικά widgets
 - EditText
 - CheckBox
 - RadioButton
 - Spinner
- SeekBar
- Portrait / Landscape
- Μενού
- Προσθήκη Activity
- Intents
 - Εισαγωγή
 - Constructor & Μέθοδοι
 - Διαδικασία



Διαδικασία (1/8)

- Προσθήκη string array για τα περιεχόμενα του ListView

```
<string-array name="arr_title">  
    <item>University</item>  
    <item>Digital Systems</item>  
    <item>Informatics</item>  
</string-array>
```

- Προσθήκη string για συνδέσμους και τηλέφωνα

```
<string name="sec1_link">http://www.unipi.gr</string>  
<string name="sec1_tel">2104141000</string>  
<string name="sec2_link">http://www.ds.unipi.gr</string>  
<string name="sec2_tel">2104141111</string>
```



Διαδικασία (2/8)

- Προσθήκη ListView στο activity_main.xml
 - Το αντικείμενο βρίσκεται στην ενότητα Containers
 - Επιλογή entries: Πίνακα strings
android:entries="@array/arr_title"



Διαδικασία (3/8)

- Υλοποίηση interface Listener στη **MainActivity.java**
 - ... implements OnItemClickListener
 - protected ListView myList;
- Προσθήκη στην onCreate()
 - myList = (ListView) findViewById(R.id.listView);
 - myList.setOnItemClickListener(this);
- Υλοποίηση Listener
 - Δημιουργία Intent:
 - Intent myIntent = new Intent(this, ItemActivity.class);
 - Τοποθέτηση τιμών (χρήση της getResources() για τη λήψη τιμών από resources του project)
 - myIntent.putExtra("link", getResources().getString(R.string.sec1_link));
 - Έναρξη activity
 - startActivity(myIntent);



Διαδικασία (4/8)

@Override

```
public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  
    Intent myIntent = new Intent(this, ItemActivity.class);  
    if (position ==0)  
    {  
        myIntent.putExtra("link", getResources().getString(R.string.sec1_link));  
        myIntent.putExtra("tel", getResources().getString(R.string.sec1_tel));  
    }  
    if (position ==1)  
    {  
        myIntent.putExtra("link", getResources().getString(R.string.sec2_link));  
        myIntent.putExtra("tel", getResources().getString(R.string.sec2_tel));  
    }  
    startActivity(myIntent);  
}
```



Διαδικασία (5/8)

- Δημιουργία νέας Empty Activity: ItemActivity
- Στο layout για την προβολή νέας «οθόνης» (αφού ο χρήστης επιλέξει ένα αντικείμενο του ListView της αρχικής activity)
 - Προσθήκη 2 TextViews (ορισμός IDs στο xml)



Διαδικασία (6/8)

□ **ItemActivity.class**

private TextView tvLink;

private TextView tvTel;

□ Αλλαγή της onCreate()

■ Δημιουργία Intent και λήψη τιμών

Intent consIntent = getIntent();

String myLink = consIntent.getStringExtra("link");

String myTel = consIntent.getStringExtra("tel");

■ Τοποθέτηση τιμών στα αντίστοιχα αντικείμενα της οθόνης

tvLink = (TextView) findViewById(R.id.textViewLink);

tvTel = (TextView) findViewById(R.id.textViewTel);

tvLink.setText(myLink);

tvTel.setText(myTel);

■ Υλοποίηση interface Listener (της view.View)

... implements OnClickListener



Διαδικασία (7/8)

- Προσθήκη στην onCreate()

 - `tvLink.setOnClickListener(this);`

 - `tvTel.setOnClickListener(this);`

- Προσθήκη στην onClick ()

 - Δημιουργία Intent και λήψη τιμών

 - `Intent consIntent = getIntent();`

 - `String myLink = consIntent.getStringExtra("link");`

 - `Uri myUri = Uri.parse(myLink);`

 - `String myTel = "tel:"+consIntent.getStringExtra("tel");`

 - `Uri callUri = Uri.parse(myTel);`



Διαδικασία (8/8)

□ Προσθήκη στην onClick () - συνέχεια

■ Έναρξη νέας Activity

```
switch (v.getId()) {  
    case R.id.textViewLink:  
        Intent viewIntent = new Intent(Intent.ACTION_VIEW,myUri);  
        startActivity(viewIntent);  
        break;  
    case R.id.textViewTel:  
        Intent callIntent = new Intent(Intent.ACTION_DIAL,callUri);  
        startActivity(callIntent);  
        break;  
}
```




Κλήση αριθμού

- Αλλαγή στην ItemActivity.class

`Intent callIntent = new Intent(Intent.ACTION_CALL,callUri);`

- Δεν επιτρέπεται η κλήση αριθμού

- Αλλαγή του AndroidManifest

`<uses-permission`

`android:name="android.permission.CALL_PHONE"/>`