



# Εισαγωγή στις ΤΠΕ

(για φοιτητές Τμήματος Χημικών Μηχανικών)

## Εργαστηριακή Ενότητα 10 : Βόλτα στο Παρίσι

Διδάσκων: Νικόλαος Τσέλιος

Τμήμα Επιστημών της Εκπαίδευσης και της Αγωγής  
στην Προσχολική Ηλικία



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



# Σκοποί ενότητας

- Να σχεδιάσει μια εφαρμογή ξενάγησης με το App Inventor
- Να τη δοκιμάσει στη συσκευή του
- Να μάθει να καλεί άλλες εφαρμογές μέσα από την εφαρμογή που προγραμματίζει
- Να μάθει να καλεί διαδικτυακές σελίδες μέσα από την εφαρμογή
- Να μάθει να χρησιμοποιεί λίστες

# Περιεχόμενα ενότητας

- Περιγραφή: Δημιουργούμε, δοκιμάζουμε και τροποποιούμε μια εφαρμογή ξενάγησης με τη χρήση του App Inventor
- Λέξεις Κλειδιά: App Inventor, Mobile devices, Activity Starter, Web Viewer, list picker

# Η εφαρμογή Paris MapTour

- Θα σχεδιάσουμε μια εφαρμογή ξενάγησης σε πόλη
- Θα χρησιμοποιήσουμε το αντικείμενο Activity Starter για να εκκινούμε μια άλλη εφαρμογή
- Θα χρησιμοποιήσουμε λίστες
- Θα καλούμε διαδικτυακές σελίδες μέσα από την εφαρμογή

# Η εφαρμογή Paris MapTour (designer)



Component type	Palette group	What you'll name it	Purpose
Image	User Interface	Image1	Show a static image of Paris on screen.
Label	User Interface	Label1	Display the text "Discover Paris with your Android!"
ListPicker	User Interface	ListPicker1	When clicked, a list of destination choices will appear.
ActivityStarter	Connectivity	ActivityStarter1	Launch the Maps app when a destination is chosen.

# Βήμα 1

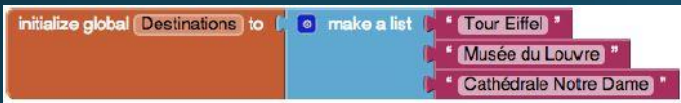
Property	Value
Action	android.intent.action.VIEW
ActivityClass	com.google.android.maps.MapActivity
ActivityPackage	com.google.android.apps.maps

- Επιλέγουμε νέο έργο (project)
- Επιλέγουμε designer
- Προσθέτουμε τα αντικείμενα που είδαμε στην προηγούμενη διαφάνεια
- Στο αντικείμενο ActivityStarter βάζουμε τις ακόλουθες τιμές στις ιδιότητες (χωρίς καμία αλλαγή στα κεφαλαία-μικρά)
- Στην ιδιότητα DataUri θα τοποθετούμε το σύνδεσμο από το Google Maps.
- Αυτό θα γίνεται δυναμικά οπότε θα το δούμε αργότερα.
- Βάζουμε την εικόνα metro.jpg (picture property Image1)
- <http://www.appinventor.org/bookFiles/MapTour/metro.jpg>
- Το Listpicker όταν πατηθεί θα εμφανιστούν οι δυνατές επιλογές. Το όνομα του πλήκτρου να γίνει «Choose Paris Destination»

# Βήμα 2

Block type	Drawer	Purpose
initialize global ("Destinations")	Variables	Create a list of the destinations.
make a list	Lists	Add the items to the list.
text ("Tour Eiffel")	Text	The first destination.
text ("Musée du Louvre")	Text	The second destination.
text ("Cathédrale Notre Dame")	Text	The third destination.

- Στο Blocks Editor δημιουργούμε μια μεταβλητή με τους δυνατούς προορισμούς
- Χρησιμοποιούμε τα block όπως φαίνονται στον πίνακα αριστερά
- Αρχικά το list block θα έχει μόνο 2 υποδοχές, δημιουργούμε μια Τρίτη πατώντας πάνω στο μπλε κουμπί



# Βήμα 3

Block type	Drawer	Purpose
Screen1.Initialize	Screen1	This event is triggered when the app starts.
set ListPicker1.Elements to	ListPicker1	Set this property to the list that you want to appear.
get global destinations	Drag out from variable initialization block	The list of destinations.



- Για να απεικονίζονται οι επιλογές αυτές θα πρέπει να χρησιμοποιήσουμε κατάλληλα blocks
- Δηλαδή θα 'περάσουμε' στο ListPicker1 τις επιλογές που έχουμε στη λίστα Destinations



# Βήμα 4

Block type	Drawer	Purpose
ListPicker1.AfterPicking	ListPicker1	This event is triggered when the user chooses from List Picker.
set ActivityStarter1.DataUri to	ActivityStarter1	The DataUri instructs Maps which map to open on launch.
join	Text	Build the DataUri from two pieces of text.
text ("http://maps.google.com?q=")	Text	The first part of the DataUri expected by Maps.
ListPicker1.Selection	ListPicker1	The item the user chose.
ActivityStarter1.StartActivity	ActivityStarter1	Launch Maps.

Concatenates the Maps URL with the search terms selected in ListPicker1, e.g., "http://maps.google.com/?q=Tour Eiffel".

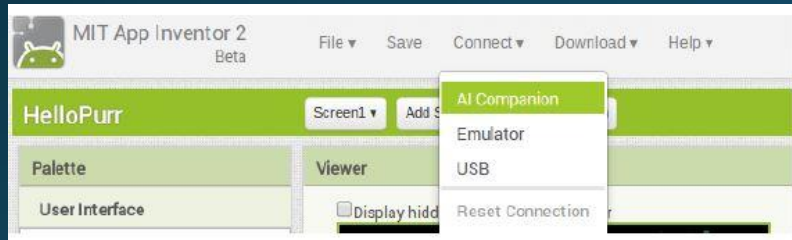
when ListPicker1 .AfterPicking

do

- set ActivityStarter1 .DataUri to
- join "http://maps.google.com/?q=" ListPicker1 . Selection
- call ActivityStarter1 .StartActivity

- Ακολουθως, θα προγραμματίσουμε κατάλληλα την εφαρμογή ώστε όταν ο χρήστης επιλέγει ένα προορισμό να εμφανίζεται στο Google Maps
- Στο Google maps όταν δώσουμε το url <http://maps.google.com?q=Paris> θα εμφανιστεί ο χάρτης του Παρισιού
- προσέξτε το ?, μετά έχει q-query και Paris δηλαδή το σημείο που μας ενδιαφέρει
- Στην εφαρμογή μας θα κατασκευάζουμε δυναμικά το ερώτημα που μας ενδιαφέρει (δηλαδή αντί για Paris, θα βάζουμε ένα στοιχείο της λίστας)
- Έτσι όταν θα επιλέγει ο χρήστης Tour Eiffel το dataURI θα γίνεται "http://maps.google.com/?q= Tour Eiffel"
- Το αντικείμενο ActivityStarter1 θα αναλαμβάνει να ανοίξει τον χάρτη

# Δοκιμή



- Μπορούμε να δοκιμάσουμε την εφαρμογή στο κινητό μας!
- Καλό είναι να το κάνουμε κάθε φορά που προσθέτουμε / τροποποιούμε κάτι στην εφαρμογή μας
- Πρέπει να έχουμε εγκαταστήσει το "MIT AI2 Companion" από το Google Play store
- Κινητό και υπολογιστής πρέπει να είναι στο ίδιο ασύρματο δίκτυο
- Επιλέγουμε Connect -> AI companion

# Δοκιμή (2)



- Ανοίγουμε στο κινητό μας την εφαρμογή “MIT AI2 Companion”
- Μπορούμε να βάλουμε τον 6ψήφιο κωδικό που εμφανίζεται ή
- Να σκανάρουμε το QR code (βάζουμε το κινητό μπροστά στην οθόνη αφού επιλέξουμε τη σχετική επιλογή)
- Γενικές οδηγίες στο <http://appinventor.mit.edu/exploration/ai2/setup.html>

# Τελική μεταφόρτωση εφαρμογής

- Επιλέγουμε εικονίδιο για την εφαρμογή (Designer ->Screen, φορτώνουμε μια εικόνα)
- Θα πρέπει να επιτρέψουμε στη συσκευή μας την εγκατάσταση εφαρμογών εκτός Google Play
  - Settings -> Applications επιλέγουμε το 'Unknown sources'
- Επιλέγουμε Build ->APK
- Πιθανά θα χρειαστούμε ένα QR code scanner (δωρεάν διαθέσιμοι στο Google Play Store)
- Μπορούμε να αποθηκεύσουμε τοπικά το αρχείο .apk και να το στείλουμε και σε άλλους χρήστες ή να το ανεβάσουμε στο δίκτυο (πχ dropbox)

# Η τελική εφαρμογή Paris Map Tour (block)

```
initialize global Destinations to  
  make a list  
    "Tour Eiffel"  
    "Musée du Louvre"  
    "Cathédrale Notre Dame"  
  
when Screen1.Initialize  
do  
  set ListPicker1.Elements to  
    get global Destinations  
  
when ListPicker1.AfterPicking  
do  
  set ActivityStarter1.DataUri to  
    join ("http://maps.google.com/?q=")  
    ListPicker1.Selection  
  call ActivityStarter1.StartActivity
```

The image shows a Scratch script with three main sections. The first section, an orange 'initialize global' block, sets a global variable named 'Destinations' to a list of three Paris landmarks: 'Tour Eiffel', 'Musée du Louvre', and 'Cathédrale Notre Dame'. The second section, a yellow 'when Screen1.Initialize' block, contains a 'do' block that sets the 'Elements' property of a 'ListPicker1' widget to the value of the 'Destinations' global variable. The third section, another yellow 'when ListPicker1.AfterPicking' block, contains a 'do' block that constructs a Google Maps URL by joining the string 'http://maps.google.com/?q=' with the 'Selection' property of 'ListPicker1', and then calls the 'StartActivity' method of an 'ActivityStarter1' widget.

# Βήμα 5 Εναλλακτική υλοποίηση με WebView αντί για ActivityStarter

- Το ActivityStarter είναι ένα εξαιρετικό αντικείμενο καθώς μπορούμε να καλούμε όποια εφαρμογή θέλουμε
- Στην περίπτωσή μας αντίστοιχα χρήσιμο είναι το αντικείμενο WebView καθώς με αυτό δημιουργούμε ένα πλαίσιο μέσα στο οποίο εμφανίζεται οποιαδήποτε σελίδα θέλουμε
- Αφαιρέστε το ActivityStarter και βάλτε ένα αντικείμενο WebView
- Απενεργοποιήστε την ιδιότητα `Screen1.Scrollable`

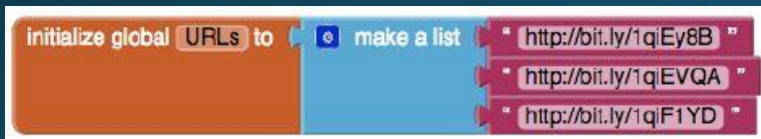
# Βήμα 6 Εναλλακτική υλοποίηση με WebView αντί για ActivityStarter

Landmark	Maps URL
Tour Eiffel	<a href="http://bit.ly/1qiEy8B">http://bit.ly/1qiEy8B</a>
Musée du Louvre	<a href="http://bit.ly/1qiEVQA">http://bit.ly/1qiEVQA</a>
Cathédrale Notre Dame (street view)	<a href="http://bit.ly/1qiF1YD">http://bit.ly/1qiF1YD</a>

- Θα πρέπει να εντοπίσουμε τα αντίστοιχα URLs για να μεταβούμε στα σημεία ενδιαφέροντος
- Αντίστοιχα θα δημιουργήσουμε μια μεταβλητή URLs που θα εμπεριέχει τους σχετικούς συνδέσμους

- <http://bit.ly/1qiEy8B> ,  
<http://bit.ly/1qiEVQA> ,  
<http://bit.ly/1qiF1YD>

- Δημιουργούμε τη σχετική λίστα



# Βήμα 7 Εναλλακτική υλοποίηση με WebViewer αντί για ActivityStarter

Block type	Drawer	Purpose
ListPicker1.AfterPicking	ListPicker1	This event is triggered when the user chooses an item.
ListPicker1.SelectionIndex	ListPicker1	The index (position) of the chosen item.
select list item	Lists	Select an item from the URLs list.
get global URLs	Drag it from the variable initialization	The list of URLs.
WebViewer.GoToURL	WebViewer	Load the URL in the viewer to show the map.



- Θα πρέπει να προσθέσουμε τα κατάλληλα blocks
- Χρειάζεται προσοχή στο εξής: Ο χρήστης επιλέγει από μια λίστα (ListPicker) αλλά η εφαρμογή χρησιμοποιεί το URL από άλλη (URLs)
- Τροποποιούμε το ListPicker1.AfterPicking
- Στην περίπτωση αυτή όταν επιλέγουμε ένα αντικείμενο από το ListPicker1 θα καλούμε το αντίστοιχο url ανάλογα με τη σειρά που τα έχουμε καταχωρήσει
- Έτσι η επιλογή Tour Eiffel θα έχει τιμή SelectionIndex=1 και θα καλείται το πρώτο url



# Η τελική εφαρμογή Paris Map Tour (block, με χρήση του WebView)

The image displays a Scratch script with the following blocks:

- initialize global Destinations to** (orange block) connected to **make a list** (blue block) containing three items: "Tour Eiffel", "Musée du Louvre", and "Cathédrale Notre Dame".
- initialize global URLs to** (orange block) connected to **make a list** (blue block) containing three items: "http://bit.ly/1qiEy8B", "http://bit.ly/1qiEVQA", and "http://bit.ly/1qiF1YD".
- when Screen1.Initialize** (yellow block) with a **do** block containing **set ListPicker1.Elements to** (green block) connected to **get global Destinations** (orange block).
- when ListPicker1.AfterPicking** (yellow block) with a **do** block containing **call WebView1.GoToUrl** (purple block). The **url** input of this block is connected to a **select list item list** (blue block). The **list** input of the select block is connected to **get global URLs** (orange block), and the **index** input is connected to **ListPicker1.SelectionIndex** (green block).

# Τι μάθαμε;

- Λίστες για να αποθηκεύουμε στοιχεία
- Χρησιμοποιούμε το αντικείμενο `ListPicker`. Η ιδιότητα `Elements` εμπεριέχει τις δυνατές επιλογές, η `Selection` το αντικείμενο που έχει επιλεχθεί, η `SelectionIndex` τη θέση του αντικειμένου που έχουμε επιλέξει. Το `AfterPicking` πυροδοτείται με την επιλογή
- Το `ActivityStarter` καλεί οποιαδήποτε εγκατεστημένη εφαρμογή

# Πιθανές επεκτάσεις;

- Δημιουργήστε μια εφαρμογή περιήγησης για το Τμήμα ή για την πόλη καταγωγής σας
- Χρησιμοποιήστε το ActivityStarter για να ανοίξετε όποια εφαρμογή θέλετε, πχ για να στείλετε mail ή να ανοίξετε το Youtube και να παίξετε ένα τραγούδι. (Διαβάστε και το <http://appinventor.mit.edu/explore/ai2/activity-starter.html> )
- Δημιουργήστε έναν τροποποιήσιμο οδηγό όπου ο χρήστης θα μπορεί να εισάγει και νέα σημεία ενδιαφέροντος.
- Αυτά μπορούν να αποθηκεύονται σε μια βάση δεδομένων (TinyWebDB). Για το πώς μπορείτε να χειριστείτε αντικείμενο βάσης δεδομένων μπορείτε να συμβουλευτείτε την επόμενη ενότητα

# Βιβλιογραφία

- Wolber, D., Abelson, H., Spertus, E., & Looney, L. (2014). *App Inventor 2: Create Your Own Android Apps*. O'Reilly Media, Inc.

# Τέλος Ενότητας