



Εισαγωγή στις ΤΠΕ

(για φοιτητές Τμήματος Χημικών Μηχανικών)

Εργαστηριακή Ενότητα 7 : Ξυλόφωνο

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



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Σκοποί ενότητας

- Να σχεδιάσει μια εφαρμογή μουσικής με το App Inventor
- Να τη δοκιμάσει στη συσκευή του
- Να μάθει να διαχειρίζεται γεγονότα σχετικά με την επαφή
- Να χρησιμοποιεί το ρολόι της συσκευής για να δημιουργήσει ή να ελέγξει γεγονότα
- Να παίξει διαφορετικούς ήχους με το αντικείμενο sound
- Να δημιουργήσει διαδικασίες που καλούν τον εαυτό τους

Περιεχόμενα ενότητας

- Περιγραφή: Δημιουργούμε, δοκιμάζουμε και τροποποιούμε μια εφαρμογή μουσικής με τη χρήση του App Inventor
- Λέξεις Κλειδιά: App Inventor, Mobile devices, clock, sound, delay, procedure, list

Η εφαρμογή Xylophone

- Θα σχεδιάσουμε μια εφαρμογή μουσικής
- Θα χρησιμοποιήσουμε λίστες
- Θα χρησιμοποιήσουμε το αντικείμενο sound το οποίο θα μπορεί να παίζει διαφορετικούς ήχους
- Θα χρησιμοποιήσουμε λίστες
- Θα χρησιμοποιήσουμε διαδικασίες που καλούν τον εαυτό τους

Η εφαρμογή Xylophone(designer)



| Component type | Palette group | What you'll name it | Purpose |
|-----------------------|----------------|------------------------|--|
| Button | User Interface | Button1 | Play Low C key. |
| Button | User Interface | Button2 | Play D key. |
| Button | User Interface | Button3 | Play E key. |
| Button | User Interface | Button4 | Play F key. |
| Button | User Interface | Button5 | Play G key. |
| Button | User Interface | Button6 | Play A key. |
| Button | User Interface | Button7 | Play B key. |
| Button | User Interface | Button8 | Play High C key. |
| Sound | Media | Sound1 | Play the notes. |
| Button | User Interface | PlayButton | Play back the song. |
| Button | User Interface | ResetButton | Reset the song memory. |
| HorizontalArrangement | Layout | HorizontalArrangement1 | Place the Play and Reset buttons next to each other. |
| Clock | User Interface | Clock1 | Keep track of delays between notes. |

Βήμα 1



- Επιλέγουμε νέο έργο (project)
- Επιλέγουμε designer
- Προσθέτουμε σταδιακά τα 2 αντικείμενα που βλέπουμε στην εικόνα
- Button
 - Name: Button1
 - Background color: Magenta
 - Text property: "C"
 - Width: Fill parent / Height 40 pixels
- Επαναλαμβάνουμε για ένα 2^ο πλήκτρο ("D", Red)
- Αργότερα θα προσθέσουμε άλλα 6 τέτοια buttons (για 8 νότες συνολικά)

Βήμα 2

- Βάζουμε ένα αντικείμενο sound (Minimum interval ο)
- Μεταφορτώνουμε τα αρχεία
- <http://appinventor.org/bookFiles/Xylophone/1.wav>
- <http://appinventor.org/bookFiles/Xylophone/2.wav>

Βήμα 3

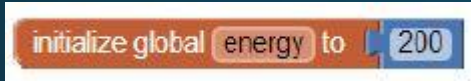
```
when Button1 .Click
do
  set Sound1 . Source to "1.wav"
  call Sound1 . Play
```

```
when Button1 .Click
do
  set Sound1 . Source to "1.wav"
  call Sound1 . Play

when Button2 .Click
do
  set Sound1 . Source to "2.wav"
  call Sound1 . Play
```

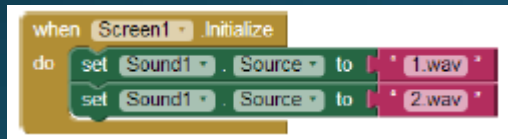
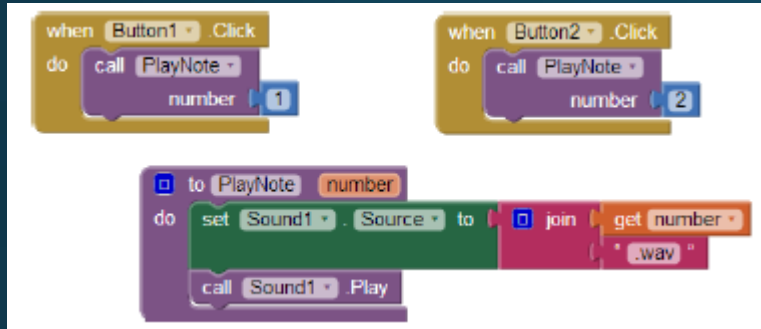
- Πατώντας το Button1 θα πρέπει να αναπαράγεται το αρχείο 1.wav
- Μπορούμε να κάνουμε το ίδιο και για το button2 κλπ
- Παρατηρούμε όμως ότι ο κώδικας επαναλαμβάνεται
- Αυτό είναι μια καλή ένδειξη ότι μπορούμε να χρησιμοποιήσουμε μια διαδικασία

Βήμα 2



- Απεικόνιση του επιπέδου ενέργειας
- Βάζουμε ένα νέο canvas στο designer (ακριβώς κάτω από το προηγούμενο)
- Energycanvas (Height->1, Width->Fillparent)
- Blocks Editor
 - Initializeglobal (text->energy)
 - Βάζουμε τιμή 200 (μέγιστη ενέργεια η ελάχιστη θα είναι 0)
- Με mouseover πάνω στην αρχικοποίηση θα εμφανίζονται τα blocks get (λάβε την τιμή) και set (επανακαθορισμός της τιμής)
- Θα οπτικοποιούμε αυτή τη τιμή ζωγραφίζοντας μια γραμμή ανάλογα με το ύψος του energy (0-200)

Βήμα 3



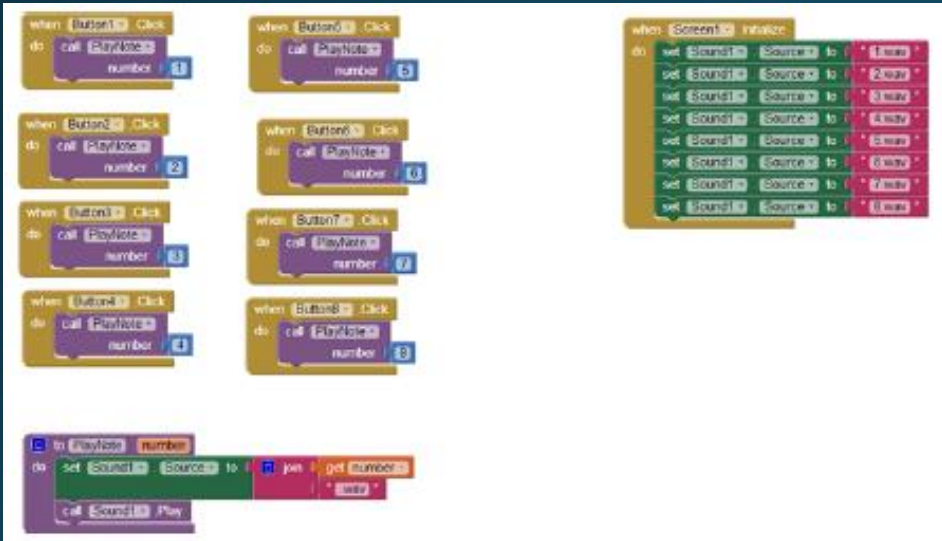
- Οπότε φτιάχνουμε μια ρουτίνα Playnote (η οποία δέχεται ως είσοδο αριθμό ανάλογα με το αν θέλουμε να παίξουμε το 1.wav, 2.wav κλπ)
- Με την εντολή join θα ενοποιούμε τον αριθμό με την επέκταση ".wav"
- Επίσης κατά τη διαδικασία αρχικοποίησης θα πρέπει να δηλώσουμε τις πηγές αρχείων του Sound1

Βήμα 4



- Πρακτικά αυτή η διαδικασία επαναλαμβάνεται και για τα υπόλοιπα 6 πλήκτρα
- Button3 ("E", Pink)
- Button4 ("F", Orange)
- Button5 ("G", Yellow)
- Button6 ("A", Green)
- Button7 ("B", Cyan)
- Button8 ("C", Blue)
- <http://appinventor.org/bookFiles/Xylophone/3.wav>
- <http://appinventor.org/bookFiles/Xylophone/4.wav>
- <http://appinventor.org/bookFiles/Xylophone/5.wav>
- <http://appinventor.org/bookFiles/Xylophone/6.wav>
- <http://appinventor.org/bookFiles/Xylophone/7.wav>
- <http://appinventor.org/bookFiles/Xylophone/8.wav>

Βήμα 5



- Αντίστοιχα τροποποιούμε τον κώδικα ώστε να παίζουμε και τα υπόλοιπα αρχεία

Βήμα 6



- Αναπαραγωγή ηχητικής ακολουθίας
- Θα καταγράψουμε τις νότες που παίξαμε (και το χρόνο) και θα αναπαράγουμε την ακολουθία
- Βάζουμε ένα αντικείμενο clock, ένα αντικείμενο playbutton (Play) και ένα resetbutton (κείμενο Reset)
- Θα δημιουργήσουμε μια λίστα notes (με την ακολουθία των ήχων που παίξαμε) και μια λίστα times (με την καταγραφή του χρόνου)



Βήμα 7



- Τροποποιούμε τον κώδικα ώστε να καταγράφουμε στη λίστα τον ήχο που παίξαμε
- Μια ενδεικτική ακολουθία μπορεί να είναι
 - notes: 1.wav, 1.wav, 1.wav, 2.wav, 3.wav
 - times [dates omitted]: 12:00:01, 12:00:02, 12:00:03, 12:00:03.5, 12:00:04
- Πατώντας το reset θα πρέπει να καθαρίσουμε τη λίστα
- Επειδή δεν θα είναι βέβαιος ο χρήστης ότι έχει συμβεί αυτό, προσθετούμε κ μια μικρή ανάδραση (δόνηση)

Βήμα 8

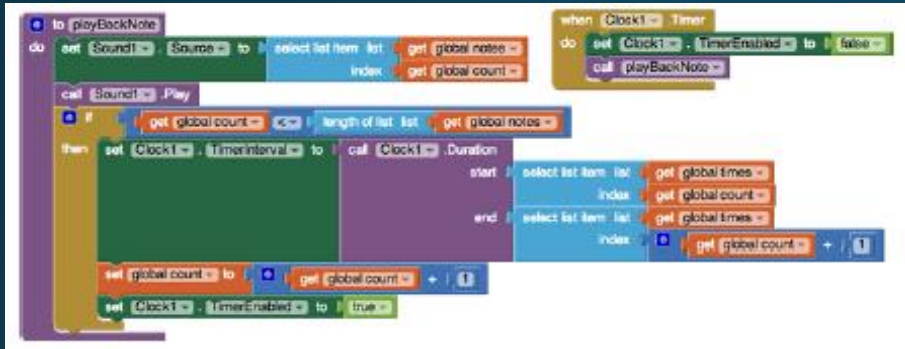
```
initialize global count to 0

when PlayButton .Click
do
  if length of list list >= get global notes > 0
  then
    set global count to 1
    call PlayBackNote

to PlayBackNote
do
  set Sound1 . Source to select list item list
  index get global notes
  get global count
  call Sound1 .Play
  if get global count <= length of list list > get global notes
  then
    set global count to get global count + 1
    call PlayBackNote
```

- Για την αναπαραγωγή προσθέτουμε τον ακόλουθο κώδικα
- Παρατηρήστε ότι στο τέλος της διαδικασίας, αν δεν έχουμε φτάσει στο τέλος της λίστας, καλούμε ξανά την ίδια διαδικασία! (αναδρομή)

Βήμα 9



- Αν και η διαδικασία είναι σωστή δεν έχει τις κατάλληλες καθυστερήσεις για να μην αναπαράγεται η μια νότα 'πάνω' στην άλλη
- Για αυτό μετά την αναπαραγωγή μιας νότας, ελέγχουμε τη λίστα global times και ανάλογα με τη χρονική απόσταση με την επόμενη νότα ενεργοποιούμε το timer.
- Όταν περάσει αυτό το διάστημα απενεργοποιούμε το timer και ξανακαλούμε τη διαδικασία playBackNote

Βήμα 10

```
when Clock1 Timer
do
  call UpdateLadybug
  call UpdateAphid
  call UpdateFrog

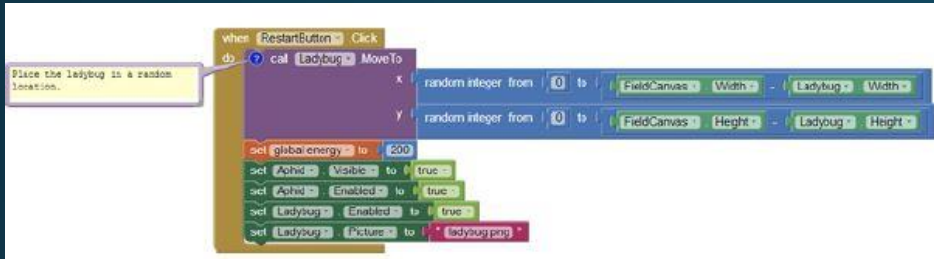
to UpdateFrog
do
  if random fraction < 0.1
  then
    set Frog Heading to atan2
      y: Frog Y - Ladybug Y
      x: Ladybug X - Frog X
  end
end
```

10% of the time, change the frog's direction to straight toward the ladybug.

```
when Ladybug CollidedWith
other
do
  if get other = Aphid and Aphid Visible
  then
    call EatAphid
  end
  if get other = Frog
  then
    set global energy to 0
    call DisplayEnergy
    call GameOver
  end
end
```

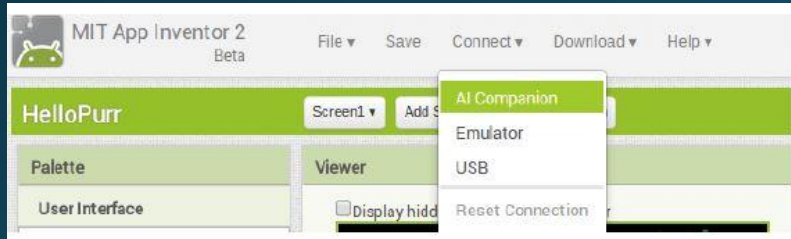
- Θα προσθέσουμε και ένα βάτραχο
- Imagesprite->Frog (Speed->1, Interval->10)
- Φτιάχνουμε διαδικασία και για το βάτραχο
- Με πιθανότητα 10% ο βάτραχος θα μετακινείται προς το σημείο που είναι η Ladybug
- Αντίστοιχα, επικαιροποιούμε το CollidedWith για την περίπτωση που θα υπάρχει επαφή βάτραχου, Ladybug

Βήμα 11



- Αν κάνουμε επανεκκίνηση έχει νόημα να εμφανιστεί η LadyBug ακριβώς στο σημείο που ήταν πριν (πάνω στο βάτραχο πιθανά);
- Θα τροποποιήσουμε τη σχετική διαδικασία ώστε να λαμβάνει το γεγονός αυτό υπόψη
- Θα την εμφανίζουμε σε μια τυχαία θέση
- Βέβαια παραμένει η (μικρή) πιθανότητα να βγει στο ίδιο σημείο
- Προσθέστε γεγονότα ήχου (πχ δόνηση όταν τρώει η Ladybug ένα έντομο ή όταν τρώγεται από τον βάτραχο)

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 Xylophone (block)

The image displays a collection of Scratch code blocks for a xylophone application. The code is organized into several functional sections:

- Initialization:** Three global variables are initialized: `times` to an empty list, `notes` to an empty list, and `count` to 0.
- Screen Initialization:** A loop sets the source of eight sound objects (`Sound1` through `Sound8`) to wav files numbered 1 through 8.
- Button Clicks:** Eight separate blocks, each triggered by a button click (`Button1` through `Button8`), call the `PlayNote` function with a specific note number (1-8).
- PlayNote Function:** A function that takes a `number` parameter. It sets the source of `Sound1` to a wav file based on the number, adds the note name and time to global lists, and starts a timer.
- PlayBackNote Function:** A function that plays a note from the global lists. It sets the source of `Sound1` to the selected note, plays it, and updates the timer and global count.
- Reset and Control:** A `ResetButton` click event resets the global lists and times. A `PlayButton` click event checks if the list is empty and increments the count if so.

Τι μάθαμε;

- Με ένα αντικείμενο `sound` παίζουμε διαφορετικούς ήχους. Θα πρέπει όμως στη διαδικασία αρχικοποίησης (`initialization`) να τους δηλώνουμε
- Με τις λίστες καταγράφουμε μια ακολουθία γεγονότων
- Καταγράφουμε από το ρολόι την ώρα. Χρησιμοποιώντας τη διαφορά των χρόνων εκτέλεσης σε 2 διαδοχικές νότες, καθορίζουμε και την καθυστέρηση στην αναπαραγωγή
- Κλήση διαδικασία από τον εαυτό της

Πιθανές επεκτάσεις;

- Τι θα γίνει αν πατήσουμε reset κατά τη διαδικασία της αναπαραγωγής; Πως μπορούμε να το αποτρέψουμε;
- Τι θα γίνει αν πατήσουμε πάλι play ενώ αναπαράγουμε ήδη τις νότες;
- Παροχή οπτικής ανάδρασης για την νότα που παίζουμε (αλλάζουμε το χρώμα του πλήκτρου πχ)
- Προαποθηκευμένα τραγούδια (θα πρέπει να 'γεμίζουμε' τη λίστα με μια ακολουθία)
- Τι θα γίνει αν ξεχαστεί ο χρήστης για πολλή ώρα χωρίς να πατήσει κάτι; Η αναπαραγωγή θα έχει ένα τεράστιο κενό. Μπορούμε να το περιορίσουμε με το max)

Βιβλιογραφία

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

Τέλος Ενότητας