

App Inventor – 7ο Μάθημα (Αριθμομηχανή)

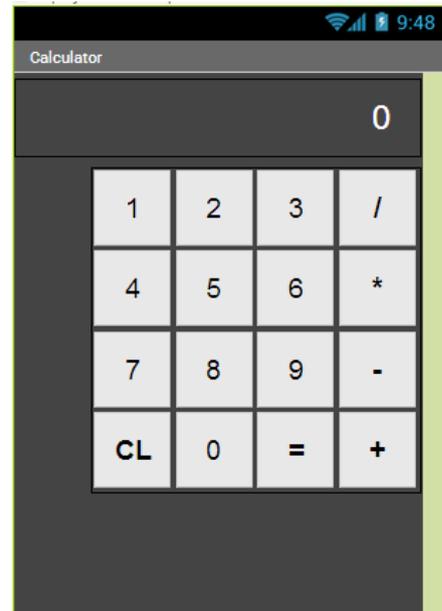
✓ Υποπρόγραμμα – διαδικασία (procedure)

Θα κατασκευάσουμε μια εφαρμογή αριθμομηχανής που θα εκτελεί όλες τις βασικές πράξεις

Βήμα 1: Εισαγωγή εφαρμογής από αρχείο

1. Στο <http://ai2.appinventor.mit.edu/> ξεκινάμε εισάγοντας το Project *Calculator.aia* (**Projects – Import project (.aia) from my computer**).

Το παραπάνω έτοιμο project περιλαμβάνει το γραφικό περιβάλλον της εφαρμογής που αποτελείται από μια ετικέτα, στην οποία θα εμφανίζονται οι αριθμοί και το αποτέλεσμα της πράξης, καθώς και από 16 κουμπιά. Αυτά περιλαμβάνουν τα ψηφία από 0 - 9, τις 4 βασικές αριθμητικές πράξεις, το ίσον (=) για την εκτέλεση της πράξης και το κουμπί CL που θα «καθαρίζει» την αριθμομηχανή σε περίπτωση που θέλουμε να ακυρώσουμε έναν υπολογισμό.



Στο Blocks παρατηρούμε ότι υπάρχει και ένα μέρος του κώδικα. Ο κώδικας αυτός περιλαμβάνει όλα τα γεγονότα που θα εκκινούν μετά το πάτημα κάθε κουμπιού χωρίς όμως τις ενέργειες που θα γίνονται.



Επιπλέον υπάρχουν τρεις εντολές αρχικοποίησης μεταβλητών. Η μεταβλητή **number** θα αποθηκεύει τον αριθμό που «πληκτρολογεί» ο χρήστης εκείνη τη στιγμή στην αριθμομηχανή. Η μεταβλητή **result** θα αποθηκεύει τον πρώτο αριθμό που πληκτρολόγησε ο χρήστης καθώς και το αποτέλεσμα της πράξης. Τέλος η μεταβλητή **action** θα αποθηκεύει το σύμβολο της πράξης που ζήτησε ο χρήστης.

initialize global **number** to " "

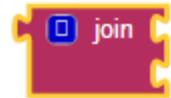
initialize global **result** to 0

initialize global **action** to " "

Βήμα 2: Διαβάζοντας τον αριθμό που πληκτρολογείται

2. Θα υλοποιήσουμε μια διαδικασία, ονομάστε την *makeNumber*, που θα παίρνει σαν παράμετρο τον αριθμό του ψηφίου που «πατήθηκε» από τον χρήστη και θα το ενώνει με τον υπόλοιπο αριθμό που έχει ήδη πληκτρολογηθεί. Για παράδειγμα, αν ο χρήστης έχει ήδη πληκτρολογήσει το ψηφίο 9 και στη συνέχεια πατήσει το ψηφίο 5, η διαδικασία μας θα πρέπει:

- α. Να τα ενώνει (δηλαδή να φτιάχνει το 95), χρησιμοποιώντας την εντολή **join**.
- β. Να αποθηκεύει το αποτέλεσμα στην μεταβλητή *number*.
- γ. Να εμφανίζει στην ετικέτα τον καινούριο αριθμό που σχηματίστηκε.



```

to makeNumber x
do
  set MainLabel . Text to ""
  set global number to join get global number
  get x
  set MainLabel . Text to get global number
  
```

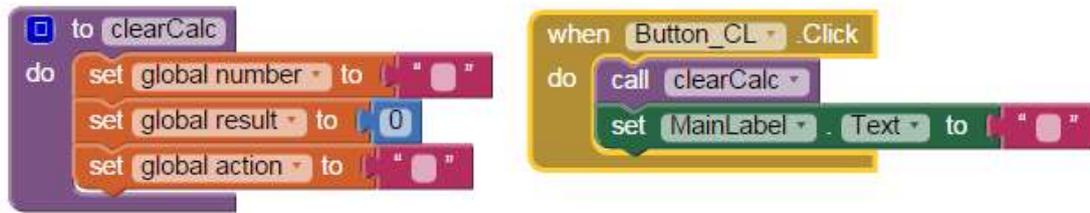
3. Τέλος, αφού ολοκληρώσουμε τη διαδικασία θα πρέπει να την καλέσουμε κατάλληλα όταν ο χρήστης αγγίζει κάποιο από τα αριθμητικά ψηφία. Υλοποιήστε τις προσθήκες του επόμενου στιγμιότυπου κάνοντας χρήση και της **Duplicate** για διευκόλυνσή σας.



4. Δοκιμάστε τι έχετε κάνει έως αυτή τη στιγμή, και αν το έχετε κάνει σωστά, στη συσκευή σας με τη βοήθεια του AI Companion, **Connect - AI Companion**.

Βήμα 3: Καθαρίζοντας την αριθμομηχανή

5. Θα υλοποιήσουμε μια απλή διαδικασία (ονομάστε την *clearCalc*) που θα «καθαρίζει» τα δεδομένα που έχουν πληκτρολογηθεί στην αριθμομηχανή. Συγκεκριμένα, θα θέτει ξανά τις αρχικές τιμές των 3 μεταβλητών.
6. Αφού ολοκληρώσουμε την υλοποίηση της διαδικασίας, θα την καλέσουμε όταν πατηθεί το κουμπί καθαρισμού CL. Επίσης, θα θέτουμε στο κείμενο της ετικέτας της αριθμομηχανής το 0.

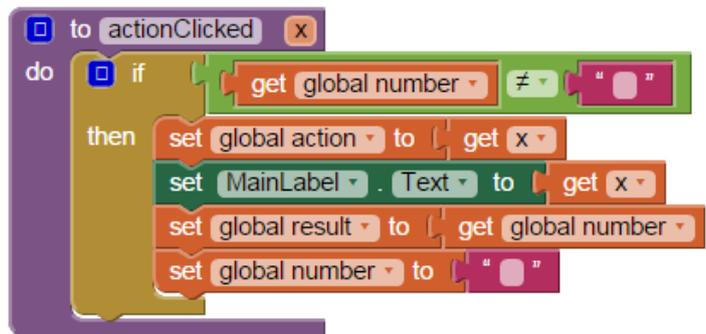


7. Δοκιμάστε στη συσκευή σας εάν έχετε κάνει σωστά αυτό το βήμα για να προχωρήσετε στο επόμενο.

Βήμα 4: Διαβάζοντας την αριθμητική πράξη

8. Η διαδικασία που θα υλοποιήσουμε σε αυτό το βήμα (ονομάστε την *actionClicked*) θα ενεργοποιείται όταν ο χρήστης αγγίζει ένα από τα κουμπιά με τις αριθμητικές πράξεις. Θα δέχεται σαν παράμετρο το σύμβολο της πράξης που επέλεξε ο χρήστης και στη συνέχεια θα εκτελεί τις παρακάτω ενέργειες:

- α. Θα ελέγχει αν η μεταβλητή *number* έχει κάποια τιμή διαφορετική από το κενό αλφαριθμητικό «». Αυτός ο έλεγχος είναι απαραίτητος, ώστε να μην επιτρέπουμε στον χρήστη να επιλέγει αριθμητική πράξη πριν πληκτρολογήσει κάποιον αριθμό. Στην περίπτωση που η μεταβλητή *number* έχει τιμή:
- β. Θα αποθηκεύει το σύμβολο της πράξης που επιλέχθηκε στην μεταβλητή *action*.
- γ. Θα θέτει το κείμενο της ετικέτας ίσο με το σύμβολο της πράξης που επιλέχθηκε.
- δ. Θα αποθηκεύει την τιμή της μεταβλητής *number* στην μεταβλητή *result*.
- ε. Θα θέτει την τιμή της μεταβλητής *number* ίση με το κενό αλφαριθμητικό «».



9. Τέλος, αφού ολοκληρώσουμε τη διαδικασία θα πρέπει να την καλέσουμε κατάλληλα όταν ο χρήστης αγγίζει κάποιο από τα κουμπιά των αριθμητικών πράξεων.



10. Δοκιμάστε στη συσκευή σας εάν έχετε υλοποιήσει σωστά την παραπάνω λειτουργία.

Βήμα 7: Υπολογίζοντας το αποτέλεσμα

11. Η τελευταία και πιο σημαντική διαδικασία που θα φτιάξουμε (ονομάστε την *calcResult*) θα ενεργοποιείται όταν ο χρήστης αγγίζει το κουμπί $=$. Οι ενέργειες που εκτελεί η διαδικασία είναι:

- α. Θα ελέγχει αν η μεταβλητή *result* έχει τιμή διαφορετική από το 0 και ταυτόχρονα η μεταβλητή *number* τιμή διαφορετική από το κενό αλφαριθμητικό. Αυτό είναι απαραίτητο προκειμένου να μην γίνονται πράξεις χωρίς να έχει πληκτρολογήσει ο χρήστης και τους δύο αριθμούς. Σε περίπτωση που ο παραπάνω έλεγχος είναι εντάξει, τότε η εφαρμογή:
- β. Θα ελέγχει ποια από τις 4 πράξεις ζήτησε ο χρήστης. Ανάλογα θα ενημερώνει την τιμή της μεταβλητής *result*.
- γ. Θα θέτει το κείμενο της ετικέτας ίσο με την τιμή της μεταβλητής *result*.
- δ. Θα καλεί τη διαδικασία που καθαρίζει την αριθμομηχανή.

```

to calcResult
do
  if (get global result ≠ 0 and get global number ≠ "")
  then
    if (get global action = "+")
    then set global result to (get global result + get global number)
    else if (get global action = "-")
    then set global result to (get global result - get global number)
    else if (get global action = "*")
    then set global result to (get global result × get global number)
    else set global result to (get global result / get global number)
  set MainLabel . Text to get global result
  call clearCalc

```

12. Η παραπάνω διαδικασία θα καλείται μόλις ο χρήστης θα κάνει click στο κουμπί «=».
13. Δοκιμάστε την εφαρμογή σας.

```

when Button_ison .Click
do call calcResult

```

Πηγή: <http://www.sepchiou.gr/>