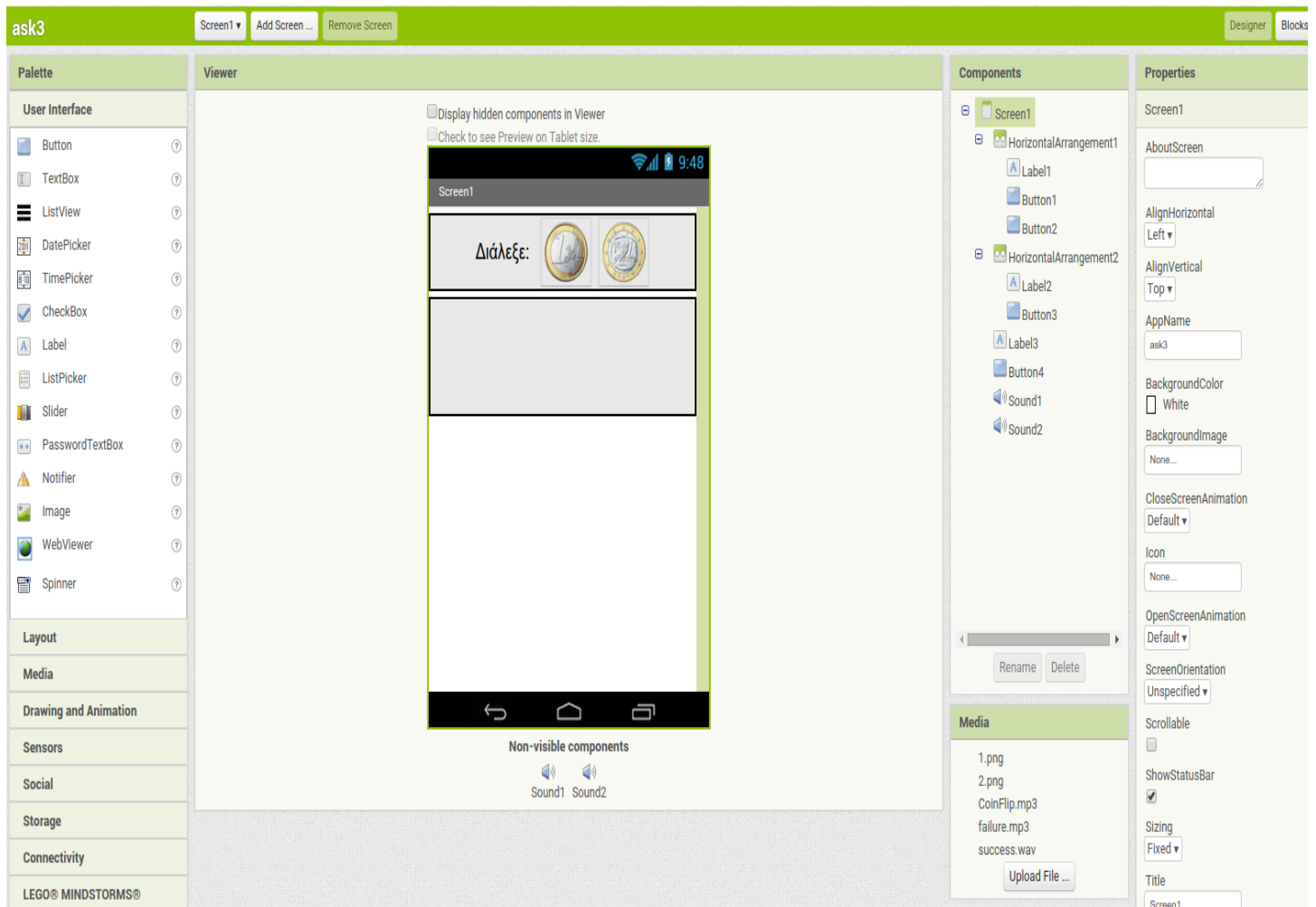


ΣΗΜΕΙΩΣΕΙΣ APPINVENTOR

DESIGNER



Στο χώρο της εργαλειοθήκης “**Palette**” του Appinventor υπάρχουν 9 κατηγορίες αντικειμένων που μπορούν να χρησιμοποιηθούν στην εκάστοτε εφαρμογής μας και είναι οι: **1) User Interface, 2) Layout, 3) Media, 4) Drawing and Animation, 5) Sensors, 6) Social, 7) Storage, 8) Connectivity, 9) Lego MindStorms**. Στις ασκήσεις μας θα χρησιμοποιήσουμε λίγα βασικά αντικείμενα από τις 6 πρώτες κατηγορίες.

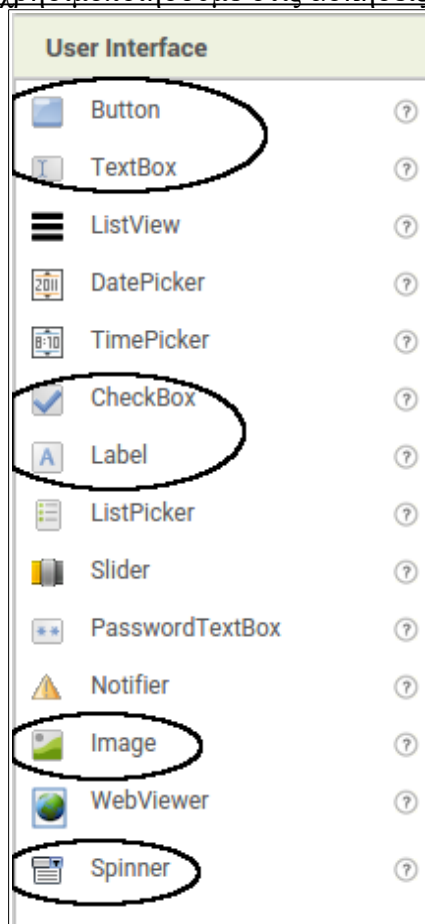
Όταν βρισκόμαστε σε κατάσταση σχεδίασης (Designer), τα αντικείμενα της εργαλειοθήκης μπορούν να συρθούν στο χώρο “**Viewer**” (ουσιαστικά δηλαδή στην οθόνη του κινητού μας) κι έτσι να χρησιμοποιηθούν στην εφαρμογή που κατασκευάζουμε.

Τα αντικείμενα που τοποθετούνται στην περιοχή Viewer εμφανίζονται και στην περιοχή “**Components**” σε μια δενδροειδή μορφή. Η μορφή είναι δενδροειδής μιας και ορισμένα αντικείμενα εμπεριέχονται σε άλλα και σίγουρα όλα εμπεριέχονται στο αντικείμενο Screen1 (ή Screen2, ... αν έχουμε πολλαπλές οθόνες).

Στην περιοχή “**Media**” εμφανίζονται όλα τα εξωτερικά αρχεία (ήχοι και εικόνες κυρίως) που απαιτεί η εφαρμογή μας και τα οποία έχουμε “ανεβάσει” εμείς οι ίδιοι επιλέγοντας το κουμπί Upload File.

Τέλος στην περιοχή “**Properties**” εμφανίζονται όλες οι ιδιότητες του αντικειμένου που είναι επιλεγμένο και οι αρχικές τιμές που αυτές έχουν κατά τη σχεδίαση της εφαρμογής. Σημειωτέον ότι οι τιμές των ιδιοτήτων αυτών πιθανότατα αλλάζουν με τη χρήση block κώδικα κατά την εκτέλεση της εφαρμογής.

Αντικείμενα της κατηγορίας **User Interface**. Κυκλωμένα είναι αυτά που πιθανότατα θα χρησιμοποιήσουμε στις ασκήσεις μας. Γι' αυτά παρακάτω φαίνονται βασικές τους ιδιότητες-γεγονότα:



Button

Αντικείμενο-Κουμπί με την ικανότητα να ανιχνεύει κλικ. Πολλές πτυχές της εμφάνισης ενός κουμπιού μπορούν να αλλάξουν, καθώς επίσης και το κατά πόσον αυτό μπορεί να πατηθεί ή όχι.

Properties - Ιδιότητες (και τιμές που μπορούν να πάρουν):

Text (οποιοδήποτε κείμενο σε " " που αναγράφεται στο κουμπί),

Enabled (true-false, δείχνει αν μπορεί να πατηθεί ή όχι)

Visible (true-false, δείχνει αν είναι ορατό ή όχι)

Width, Height (ακέραια τιμή που εκφράζει pixels πλάτους-ύψους)

Image (όνομα αρχείου μέσα σε " " της εικόνας που φαίνεται πάνω στο κουμπί)

FontSize (ακέραια τιμή που δείχνει μέγεθος του font γραμμάτων του κειμένου του κουμπιού)

Όλες οι παραπάνω ιδιότητες τίθενται με χρήση αντίστοιχων εντολών Set και οι τιμές που μπορούν να πάρουν μπορεί να βρίσκονται και σε αντίστοιχων τύπων μεταβλητές.

Π.χ.: Set Button1.Text to "Πάτησε με!"

Set Button1.Image to get global img (το img είναι μεταβλητή)

Set Button1.Visible to false (κάνει το Button1 αόρατο)

Γεγονότα (που σχετίζονται με Buttons):

Όταν κλικαρισθεί (πατηθεί και αφαιρεθεί)	Όταν πατηθεί παρατεταμένα κι ύστερα αφαιρεθεί	Την ώρα που είναι πατημένο	Την ώρα που αφήνεται
when Button1 .Click do	when Button1 .LongClick do	when Button1 .TouchDown do	when Button1 .TouchUp do

Label

Μια ετικέτα εμφανίζει ένα κομμάτι κειμένου στην οθόνη του κινητού, το οποίο καθορίζεται από την ιδιότητα Text. Άλλες βασικές ιδιότητες που μπορούν να ρυθμιστούν σε Designer ή Blocks Editor, ελέγχουν την εμφάνιση και τη θέση του κειμένου.

Properties - Ιδιότητες (και τιμές που μπορούν να πάρουν):

Text (οποιοδήποτε κείμενο σε " " το οποίο δείχνει η ετικέτα),

Visible (true-false, δείχνει αν είναι ορατή ή όχι)

Width, Height (ακέραια τιμή που εκφράζει pixels πλάτους-ύψους)

FontSize (ακέραια τιμή που δείχνει μέγεθος του font γραμμάτων του κειμένου της ετικέτας)

TextColor, BackgroundColor (χρώμα, βλ. Built-in στοιχεία Appinventor)

Όλες οι παραπάνω ιδιότητες τίθενται με χρήση αντίστοιχων εντολών Set και οι τιμές που μπορούν να πάρουν μπορεί να βρίσκονται και σε αντίστοιχων τύπων μεταβλητές.

Π.χ.: Set Label1.Text to "Πάτησε με!"

Set Label1.FontSize to 20

Γεγονότα (που σχετίζονται με Labels) δεν υπάρχουν...

Image

Αντικείμενο για την εμφάνιση εικόνων. Η εικόνα που εμφανίζεται, καθώς και άλλες πτυχές της εμφάνισης της εικόνας μπορούν να προσδιοριστούν στον Designer ή στον Blocks Editor.

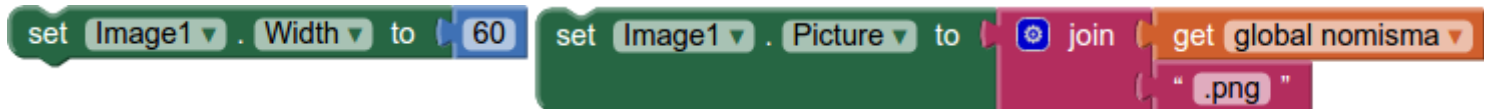
Properties - Ιδιότητες (και τιμές που μπορούν να πάρουν):

Picture (όνομα αρχείου μέσα σε " " της εικόνας που εμφανίζεται),

Visible (true-false, δείχνει αν είναι ορατή ή όχι)

Width, Height (ακέραια τιμή που εκφράζει pixels πλάτους-ύψους)

Όλες οι παραπάνω ιδιότητες τίθενται με χρήση αντίστοιχων εντολών Set και οι τιμές που μπορούν να πάρουν μπορεί να βρίσκονται και σε αντίστοιχων τύπων μεταβλητές, π.χ.:



Γεγονότα (που σχετίζονται με Image) δεν υπάρχουν.

TextBox

Ένα πλαίσιο κειμένου για το χρήστη για την εισαγωγή κειμένου. Η αρχική ή εισαγόμενη από τον χρήστη τιμή κειμένου βρίσκεται στην ιδιότητα Text. Αν είναι κενό, η ιδιότητα Hint, η οποία εμφανίζεται ως αχνό κείμενο στο πλαίσιο, μπορεί να παρέχει στο χρήστη με οδηγίες για το τι πρέπει να πληκτρολογήσετε. Η ιδιότητα NumbersOnly περιορίζει το πληκτρολόγιο για να δέχεται μόνο αριθμό ως είσοδο. Άλλες ιδιότητες επηρεάζουν την εμφάνιση του πλαισίου κειμένου (TextAlignment, BackgroundColor, κλπ) και αν μπορεί να χρησιμοποιηθεί (Enabled).

Properties - Ιδιότητες (και τιμές που μπορούν να πάρουν):

Text (οποιοδήποτε κείμενο δίνει ο προγραμματιστής από Designer ή Blocks ή ο χρήστης),

Hint (οποιοδήποτε κείμενο βάζει ο προγραμματιστής που αχνοφαίνεται ως υπόδειξη για το τι πρέπει να συμπληρωθεί στο πλαίσιο κειμένου από το χρήστη),

Enabled (true-false, δείχνει αν μπορεί να συμπληρωθεί ή όχι)

Visible (true-false, δείχνει αν είναι ορατό ή όχι)

Width, Height (ακέραια τιμή που εκφράζει pixels πλάτους-ύψους)

FontSize (ακέραια τιμή που δείχνει μέγεθος του font γραμμάτων του κειμένου του κουμπιού)

TextColor, BackgroundColor (χρώμα, βλ. Built-in στοιχεία Appinventor)

NumbersOnly (true-false, δείχνει αν δέχεται μόνο αριθμούς ή οποιοδήποτε κείμενο)

Όλες οι παραπάνω ιδιότητες τίθενται με χρήση αντίστοιχων εντολών Set και οι τιμές που μπορούν να πάρουν μπορεί να βρίσκονται και σε αντίστοιχων τύπων μεταβλητές.

Π.χ.: Set Button1.NumbersOnly to true

Γεγονότα (που σχετίζονται με TextBox): **When TextBox1.GotFocus do**, **When TextBox1.LostFocus do** (δε θα τα χρησιμοποιήσουμε στις απλές μας ασκήσεις)

Διαδικασίες (που σχετίζονται με TextBox): **call TextBox1.HideKeyboard**, **call TextBox1.RequestFocus** (δε θα τα χρησιμοποιήσουμε στις απλές μας ασκήσεις)

Spinner

Ένα αντικείμενο spinner εμφανίζει τη δυνατότητα επιλογής από μια λίστα στοιχείων (με αναδυόμενο παράθυρο όταν πατηθεί). Τα στοιχεία αυτά μπορούν να τεθούν στο Designer ή στο Blocks ορίζοντας την ιδιότητα `ElementsFromString` ως μια ακολουθία επιλογών χωριζόμενων με κόμμα. π.χ. "επιλογή1,επιλογή2,επιλογή3".

Properties - Ιδιότητες (και τιμές που μπορούν να πάρουν):

ElementsFromString (οποιοδήποτε κείμενο ως ακολουθία επιλογών χωριζόμενων με "," δίνει ο προγραμματιστής από Designer ή Blocks),

Prompt (οποιοδήποτε κείμενο βάζει ο προγραμματιστής που φαίνεται ως υπόδειξη για το τι πρέπει να κάνει ο χρήστης ή τι εμπεριέχει η λίστα επιλογών από την οποία πρέπει να επιλέξει)

SelectionIndex (οποιοσδήποτε ακέραιος μεταξύ του 1 και του πλήθους των επιλογών της λίστας. Τίθεται από τον προγραμματιστή στο Blocks ή από το χρήστη όταν επιλέγει μεταξύ των στοιχείων της λίστας).

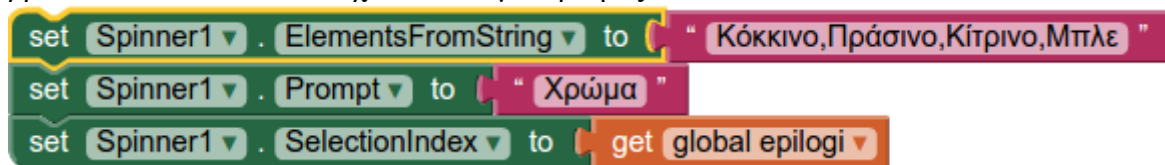
Selection (οποιοδήποτε κείμενο αντιστοιχεί σε μια από τις επιλογές της λίστας στοιχείων. Τίθεται από τον προγραμματιστή στο Designer ή στο Blocks, ή από τον χρήστη όταν επιλέγει μεταξύ των στοιχείων της λίστας).

Visible (true-false, δείχνει αν είναι ορατό ή όχι)

Width ή Height (ακέραια τιμή που εκφράζει pixels πλάτους-ύψους)

Όλες οι παραπάνω ιδιότητες τίθενται με χρήση αντίστοιχων εντολών Set και οι τιμές που μπορούν να πάρουν μπορεί να βρίσκονται και σε αντίστοιχων τύπων μεταβλητές.

Π.χ.:



Γεγονότα (που σχετίζονται με Spinner): **When Spinner1.AfterSelecting Selection do** (δε θα το χρησιμοποιήσουμε στις απλές μας ασκήσεις)

Διαδικασίες (που σχετίζονται με Spinner): **call Spinner1.DisplayDropdown** (εμφανίζει το αναδυόμενο παράθυρο)

CheckBox

Κουτάκι που δημιουργεί ένα γεγονός όταν ο χρήστης κάνει κλικ πάνω του. Υπάρχουν πολλές ιδιότητες που επηρεάζουν την εμφάνισή του, που μπορούν να ρυθμιστούν σε Designer ή Blocks Editor.

Properties - Ιδιότητες (και τιμές που μπορούν να πάρουν):

Checked (true-false, δείχνει αν είναι κλικαρισμένο ή όχι)

Text (οποιοδήποτε κείμενο δίνει ο προγραμματιστής από Designer ή Blocks για το CheckBox),

Enabled (true-false, δείχνει αν μπορεί να κλικαριστεί ή όχι)

Visible (true-false, δείχνει αν είναι ορατό ή όχι)

Width, Height (ακέραια τιμή που εκφράζει pixels πλάτους-ύψους)

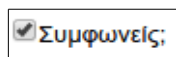
FontSize (ακέραια τιμή που δείχνει μέγεθος του font γραμμάτων του κειμένου του κουμπιού)

TextColor, BackgroundColor (χρώμα, βλ. Built-in στοιχεία Appinventor)

Όλες οι παραπάνω ιδιότητες τίθενται με χρήση αντίστοιχων εντολών Set και οι τιμές που μπορούν να πάρουν μπορεί να βρίσκονται και σε αντίστοιχων τύπων μεταβλητές.

Π.χ.: Set CheckBox1.Text to "Συμφωνείς;"

Set CheckBox1.Checked to true



Γεγονότα (που σχετίζονται με CheckBox): **When CheckBox1.Changed do, When CheckBox1.GotFocus do, When CheckBox1.LostFocus do** (Το 2ο και 3ο δε θα τα χρησιμοποιήσουμε στις απλές μας ασκήσεις)

Αντικείμενα της κατηγορίας **Layout**. Κυκλωμένα είναι αυτά που πιθανότατα θα χρησιμοποιήσουμε στις ασκήσεις μας. Γι' αυτά παρακάτω φαίνονται βασικές τους ιδιότητες:

Layout	HorizontalArrangement
 TableArrangement	?
 VerticalArrangement	?
 HorizontalArrangement	?

Στοιχείο μορφοποίησης – περιοχή στην οποία ό,τι αντικείμενο τοποθετείται εμφανίζεται από τα αριστερά προς τα δεξιά. Τη χρησιμοποιούμε για να τοποθετήσουμε μέσα της buttons, images, labels κλπ σε οριζόντια διάταξη.

Στην περιοχή components εμφανίζεται ως γονικό αντικείμενο (parent) όσων αντικειμένων τοποθετηθούν μέσα σε αυτή.

Properties - Ιδιότητες (και τιμές που μπορούν να πάρουν):

AlignHorizontal ή **AlignVertical** (ακέραιος 1-3. 1=Αριστερή ή πάνω στοίχιση αντικειμένων, 2=Στο κέντρο, 3=Δεξιά ή κάτω στοίχιση),

Visible (true-false, δείχνει αν είναι ορατή ή όχι αυτή και τα αντικείμενα που περιέχει)

Width, Height (ακέραια τιμή που εκφράζει pixels πλάτους-ύψους)

Image (όνομα αρχείου μέσα σε " " της εικόνας που φαίνεται ως background στην περιοχή)

BackgroundColor (χρώμα, βλ. Built-in στοιχεία Appinventor)

Όλες οι παραπάνω ιδιότητες τίθενται με χρήση αντίστοιχων εντολών Set και οι τιμές που μπορούν να πάρουν μπορεί να βρίσκονται και σε αντίστοιχων τύπων μεταβλητές.

Π.χ.: Set HorizontalArrangement1.AlignHorizontal to 2 (στοιχίζει οριζόντια τα αντικείμενα της στο κέντρο)

VerticalArrangement

Στοιχείο μορφοποίησης – περιοχή στην οποία ό,τι αντικείμενο τοποθετείται εμφανίζεται από πάνω προς τα κάτω. Τη χρησιμοποιούμε για να τοποθετήσουμε μέσα της buttons, images, labels κλπ σε κάθετη διάταξη. Στην περιοχή components εμφανίζεται ως γονικό αντικείμενο (parent) όσων αντικειμένων τοποθετηθούν μέσα σε αυτή.

Properties - Ιδιότητες (ίδιες με την Horizontal Arrangement και τίθενται με αντίστοιχες εντολές Set):

Αντικείμενα της κατηγορίας **Media**. Κυκλωμένα είναι αυτά που πιθανότατα θα χρησιμοποιήσουμε στις ασκήσεις μας. Γι' αυτά παρακάτω φαίνονται βασικές τους ιδιότητες:

Media	Sound
 Player	?
 Sound	?
 TextToSpeech	?
 YandexTranslate	?
 VideoPlayer	?
 Camcorder	?
 Camera	?
 SpeechRecognizer	?
 ImagePicker	?
 SoundRecorder	?

Ενα αντικείμενο πολυμέσων που παίζει αρχεία ήχου και προαιρετικά δονείται για τόσα χιλιοστά του δευτερολέπτου όσα καθορίζονται στο Blocks. Το όνομα του αρχείου ήχου που θα παίζει πρέπει να καθοριστεί με την ιδιότητα Source είτε στο Designer ή στο Blocks. Είναι καλύτερο για μικρά αρχεία ήχου, όπως ηχητικά εφέ. Για μεγαλύτερου χρονικού διαστήματος ήχους, όπως τραγούδια να χρησιμοποιείτε το αντικείμενο Player.

Properties - Ιδιότητες (και τιμές που μπορούν να πάρουν):

Source (οποιοδήποτε κείμενο μέσα σε " " ή μεταβλητή με κείμενο που παραπέμπει σε αρχείο ήχου π.χ. mp3, wav που είναι φορτωμένο στην εφαρμογή)

MinimumInterval (ακέραιος που εκφράζει τα χιλιοστά του δευτερολέπτου που πρέπει να περάσουν για να μπορεί να ξανακουστεί ήχος που αντιστοιχεί στο ίδιο αντικείμενο ήχου)

Οι παραπάνω ιδιότητες τίθενται με χρήση αντίστοιχων εντολών Set και οι τιμές που μπορούν να πάρουν μπορεί να βρίσκονται και σε αντίστοιχων τύπων μεταβλητές.

Π.χ.: Set Sound1.Source to "success.mp3"

Set Sound1.MinimumInterval to 1000 (πρέπει να περάσει 1sec για να ξανακληθεί η διαδικασία Sound1.Play)

Διαδικασίες (που σχετίζονται με Sound1):

call Sound1.Play (παίζει τον ήχο που αντιστοιχεί στο αντικείμενο Sound1)

call Sound1.Stop (σταματάει να παίζει τον ήχο που αντιστοιχεί στο αντικείμενο Sound1)

call Sound1.Pause (σταματάει προσωρινά τον ήχο που αντιστοιχεί στο αντικείμενο Sound1)

call Sound1.Resume (ξαναρχίζει τον ήχο που αντιστοιχεί στο αντικείμενο Sound1 απ' όπου είχε σταματήσει με κλήση διαδικασίας Sound1.Pause)

call Sound1.Vibrate (παίρνει σαν είσοδο ακέραιο που εκφράζει msec και δονεί το κινητό για αντίστοιχο χρονικό διάστημα κατά το παίξιμο του ήχου που αντιστοιχεί στο Sound1)

Player

Αντικείμενο πολυμέσων που παίζει ήχους και ελέγχει τη δόνηση του κινητού. Το όνομα του αρχείου ήχου που θα παίζει πρέπει να καθοριστεί με την ιδιότητα Source είτε στο Designer ή στο Blocks. Αυτό το αντικείμενο είναι καλύτερο για μεγάλα αρχεία ήχου, όπως τραγούδια, ενώ το αντικείμενο Sound είναι πιο αποδοτικό για μικρά αρχεία, όπως τα ηχητικά εφέ.

Properties - Ιδιότητες (και τιμές που μπορούν να πάρουν):

Source (οποιοδήποτε κείμενο μέσα σε " " ή μεταβλητή με κείμενο που παραπέμπει σε αρχείο ήχου π.χ. mp3, wav που είναι φορτωμένο στην εφαρμογή)

Volume (ακέραιος 0-100 που εκφράζει την ένταση του ήχου)

Loop (true-false, καθορίζει αν ο ήχος θα αρχίσει από την αρχή όταν ολοκληρωθεί το παίξιμό του)

Οι παραπάνω ιδιότητες τίθενται με χρήση αντίστοιχων εντολών Set και οι τιμές που μπορούν να πάρουν μπορεί να βρίσκονται και σε αντίστοιχων τύπων μεταβλητές.

π.χ. Set Player1.Loop to true (θα παίζει κατ' επανάληψη τον ήχο)

Γεγονότα (που σχετίζονται με Player): **When Player1.Completed do** (Γεγονός που συμβαίνει όταν ολοκληρώνεται το παίξιμο του ήχου)

Διαδικασίες (που σχετίζονται με Player):

call Player1.Start (παίζει τον ήχο που αντιστοιχεί στο αντικείμενο Player1)

call Player1.Stop (σταματάει να παίζει τον ήχο που αντιστοιχεί στο αντικείμενο Player1)

call Player1.Pause (σταματάει προσωρινά τον ήχο που αντιστοιχεί στο αντικείμενο Player1)

call Player1.Vibrate (παίρνει σαν είσοδο ακέραιο που εκφράζει msec και δονεί το κινητό για αντίστοιχο χρονικό διάστημα κατά το παίξιμο του ήχου που αντιστοιχεί στο Player1)

Αντικείμενα της κατηγορίας **Drawing and Animation**. Κυκλωμένα είναι αυτά που πιθανότατα θα χρησιμοποιήσουμε στις ασκήσεις μας. Γι' αυτά παρακάτω φαίνονται βασικές τους ιδιότητες:

Drawing and Animation	Canvas
 Canvas	Αντικείμενο που αντιστοιχεί σε ένα διδιάστατο ορθογώνιο χώρο όπου μπορεί κάποιος να ζωγραφίσει και στοιχεία εικόνων μπορούν να κινηθούν. Κάθε σημείο του Canvas προσδιορίζεται από ένα ζευγάρι τιμών (x, y):
 ImageSprite	το x αντιστοιχεί στον αριθμό των pixels από την αριστερή ακμή του Canvas
 Ball	το y αντιστοιχεί στον αριθμό των pixels από την πάνω ακμή του Canvas.

Υπάρχουν γεγονότα που υποδεικνύουν πότε και που έχει πατηθεί ένας Canvas ή ένα ImageSprite ή Ball έχει συρθεί. Επίσης υπάρχουν μέθοδοι για σχεδιασμό σημείων, γραμμών και κύκλων.

Properties - Ιδιότητες (και τιμές που μπορούν να πάρουν):

BackgroundColor - PaintColor (χρώμα, βλ. *Built-in* στοιχεία. Καθορίζει το χρώμα φόντου του Canvas και το χρώμα που θα χρησιμοποιηθεί για ζωγραφική σε αυτόν),

BackgroundImage (όνομα αρχείου μέσα σε " " της εικόνας που φαίνεται ως φόντο στον Canvas)

Visible (true-false, δείχνει αν είναι ορατός ή όχι)

Width, Height (ακέραια τιμή που εκφράζει pixels πλάτους-ύψους)

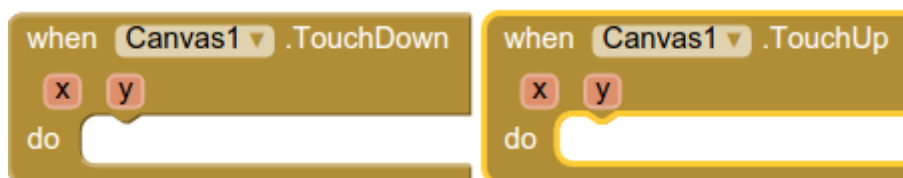
LineWidth (ακέραια τιμή που υποδεικνύει το πάχος γραμμής που σχεδιάζεται με αντίστοιχη διαδικασία στον Canvas)

FontSize (ακέραια τιμή που υποδεικνύει το μέγεθος του font του κειμένου που σχεδιάζεται στον Canvas)

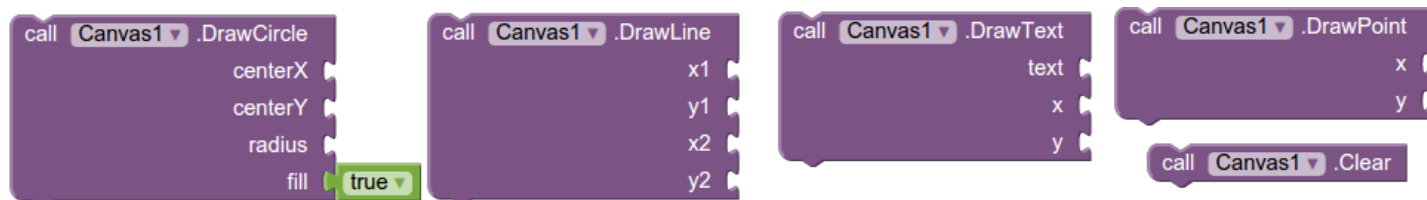
Όλες οι παραπάνω ιδιότητες τίθενται με χρήση αντίστοιχων εντολών Set και οι τιμές που μπορούν να πάρουν μπορεί να βρίσκονται και σε αντίστοιχων τύπων μεταβλητές.

Π.χ.: Set Canvas1.BackgroundImage to "Grass.jpg"

Γεγονότα (που σχετίζονται με Canvas και χρησιμοποιήσαμε στην άσκηση AngryDog). Οι εντολές που θα τοποθετούνται μέσα σε αυτά τα γεγονότα μπορούν να χρησιμοποιήσουν τις τιμές των x, y όπου πατήθηκε ή "ξε-πατήθηκε" ο καμβάς (get x, get y)



Διαδικασίες (που σχετίζονται με Canvas):



Χρησιμοποιούνται για να σχεδιάσουμε με το δάκτυλο πάνω στον Canvas κύκλους, γραμμές, κείμενο ή σημείο στα σημεία όπου πατήθηκε το δάκτυλό μας ή μεταξύ των σημείων που πατήθηκε και σύρθηκε το δάκτυλό μας. Επίσης για καθαρισμό του Canvas.

Πιθανότατα δε θα προλάβουμε να κάνουμε άσκηση όπου να τις χρησιμοποιήσουμε στα πλαίσια του μαθήματος.

ImageSprite

Αντικείμενο που μπορεί να τοποθετηθεί μέσα σε έναν Canvas όπου μπορεί να αλληλεπιδράσει σε πατήματα και συρσίματα του δακτύλου ή με άλλα ImageSprites ή Balls καθώς και με τις άκρες του Canvas. κινούμενο βάσει των τιμών των ιδιοτήτων του. Το τι εικόνα θα έχει εξαρτάται την ιδιότητά του Picture .

Properties - Ιδιότητες (και τιμές που μπορούν να πάρουν):

Picture (όνομα αρχείου μέσα σε " " της εικόνας που έχει το ImageSprite)

Visible (true-false, δείχνει αν είναι ορατό ή όχι)

Width, Height (ακέραια τιμή που εκφράζει pixels πλάτους-ύψους)

Speed (ακέραιος που δείχνει την ταχύτητα με την οποία κινείται το ImageSprite, εκφράζει τα pixels που θα μετακινηθεί σε χρόνο που υποδεικνύεται από την ιδιότητα Interval)

Interval (ακέραιος που υποδεικνύει κάθε πόσο χρονικό διάστημα σε msec ανανεώνεται η θέση του)

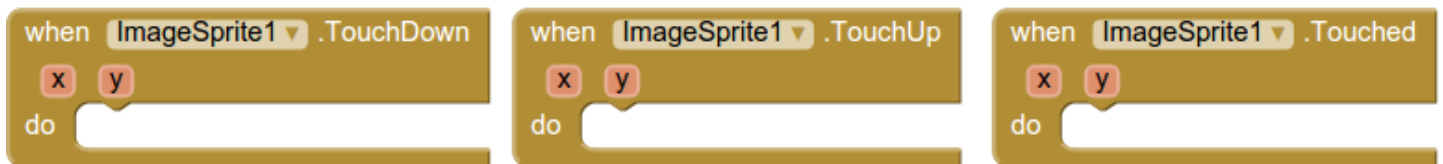
Heading(ακέραιος που υποδεικνύει γωνία σε μοίρες που σχηματίζει η κατεύθυνση κίνησής του με τον x'x)

X ή Y ή Z (υποδεικνύει τη θέση του στον Canvas)

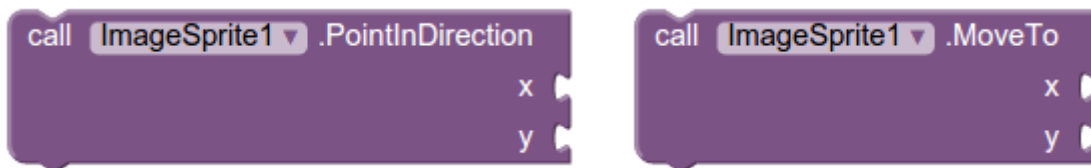
Όλες οι παραπάνω ιδιότητες τίθενται με χρήση αντίστοιχων εντολών Set και οι τιμές που μπορούν να πάρουν μπορεί να βρίσκονται και σε αντίστοιχων τύπων μεταβλητές.

Π.χ.: Set ImageSprite1.Picture to "SleepyDog.png"

Γεγονότα (που σχετίζονται με ImageSprite). Οι εντολές που θα τοποθετούνται μέσα σε αυτά τα γεγονότα μπορούν να χρησιμοποιήσουν τις τιμές των x, y όπου πατήθηκε ή "ξε-πατήθηκε" το ImageSprite (get x, get y)



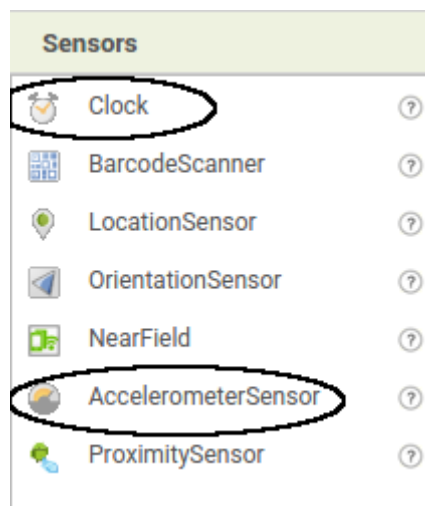
Διαδικασίες (που σχετίζονται με ImageSprite, την 1η τη χρησιμοποιήσαμε στην άσκηση AngryDog):



Η πρώτη διαδικασία κάνει το ImageSprite1 να κατευθυνθεί προς την κατεύθυνση των x,y που ορίζονται.

Η δεύτερη διαδικασία μετακινεί αυτόματα το ImageSprite1 στη θέση x,y που ορίζεται.

Αντικείμενα της κατηγορίας **Sensors**. Κυκλωμένα είναι αυτά που πιθανότατα θα χρησιμοποιήσουμε στις ασκήσεις μας. Γι' αυτά παρακάτω φαίνονται βασικές τους ιδιότητες:



AccelerometerSensor

Μη ορατό αντικείμενο – αισθητήρας που μπορεί να εντοπίσει ταρακουνήματα του κινητού και να μετρήσει την επιτάχυνσή του στις 3 διαστάσεις.

Properties - Ιδιότητες (και τιμές που μπορούν να πάρουν):

Enabled (true-false, δείχνει αν είναι ενεργοποιημένος ο αισθητήρας)

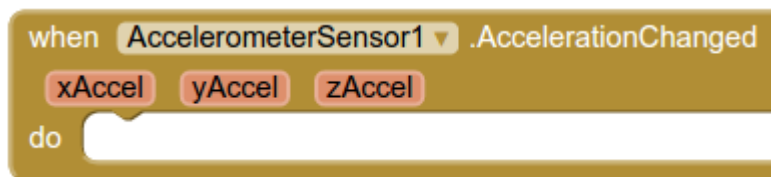
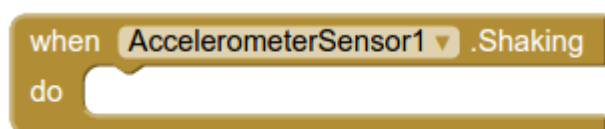
MinimumInterval (ελάχιστο χρον.διάστημα σε msec μεταξύ ταρακουνημάτων που μπορεί να ανιχνεύσει)

Sensitivity (ακέραιος 1-3. 1=μικρή ευαισθησία 2=μέτρια 3=μεγάλη)

Xaccel, Yaccel, Zaccel (αυτές οι ιδιότητες τίθενται αυτόματα από τον αισθητήρα, δεν μπορεί να τις πειράξει ο προγραμματιστής. Xaccel θετικό όταν

το κινητό γέρνει δεξιά ή αρνητικό όταν γέρνει αριστερά, Yaccel θετικό όταν το πίσω μέρος του κινητού γέρνει προς τα πάνω ή αρνητικό όταν το μπροστά μέρος γέρνει προς τα πάνω, Zaccel -9,8 όταν κινητό αφημένο “ανάσκελα” σε επίπεδο ή +9,8 όταν είναι αφημένο “μπρούμυτα”)

Γεγονότα (που σχετίζονται με AccelerometerSensor). Στην άσκηση με τα ζάρια χρησιμοποιήσαμε το πρώτο.



Clock

Μη ορατό αντικείμενο που παρέχει τη δυνατότητα μετρήσεων χρόνου με τη χρήση του εσωτερικού ρολογιού του κινητού. Ουσιαστικά πυροδοτεί χρονικά γεγονότα (χτύπους ρολογιού) σε καθοριζόμενα χρονικά διαστήματα για την εκτέλεση υπολογισμών και χειρισμών που σχετίζονται με το χρόνο.

Properties - Ιδιότητες (και τιμές που μπορούν να πάρουν):

TimerEnabled (true-false, δείχνει αν είναι ενεργοποιημένο το ρολόι ή όχι)

TimerInterval (χρονικό διάστημα σε msec μεταξύ κτύπων-γεγονότων του ρολογιού)

TimerAlwaysFires (true-false, δείχνει αν το ρολόι παράγει γεγονότα ακόμα κι όταν η εφαρμογή δεν φαίνεται στην οθόνη του κινητού – αλλά προφανώς εκτελείται, ή όχι)

Γεγονός (που σχετίζονται με Clock).



Διαδικασίες (που σχετίζονται με Clock είναι πάρα πολλές αλλά δε θα δούμε στις ασκήσεις μας)

Αντικείμενα της κατηγορίας **Social**. θα χρησιμοποιήσουμε στην άσκηση5 αλλά δεν θα έχουμε να τα ξέρουμε.

BLOCKS



Στην περιοχή των Blocks – στον Blocks Editor – υπάρχουν εκτός των σχετικών blocks των αντικειμένων που έχουμε εισάγει στην εφαρμογή μας από το Designer και ενσωματωμένα στοιχεία που πιθανόν να χρειαστούν κατά τη σύνθεση του block κώδικα της εφαρμογής.

Στα ενσωματωμένα στοιχεία συναντάμε:

Control blocks: στοιχεία για τον έλεγχο ροής της εφαρμογής, όπως στοιχεία επιλογής (if), επανάληψης (for ή while), χειρισμού οθονών (open another ή close screen) κλπ.

Logic blocks: στοιχεία για λογικές πράξεις (and, or, not), καθώς και λογικές σταθερές (true, false)

Math blocks: στοιχεία για αριθμητικές πράξεις ("+", "-", "x", "/", "^", "modulo of ") και πράξεις σύγκρισης ("=", ">", "≥", "<", "≤", "≠"), συναρτήσεις ("square root", "absolute", "neg", "log", "e^", "round", "ceiling", "floor", "sin" -ημίτονο-, "cos" -συνημίτονο-, "tan" -εφαπτομένη-, γεννήτριες τυχαίων αριθμών ("random integer from 1 to 100")

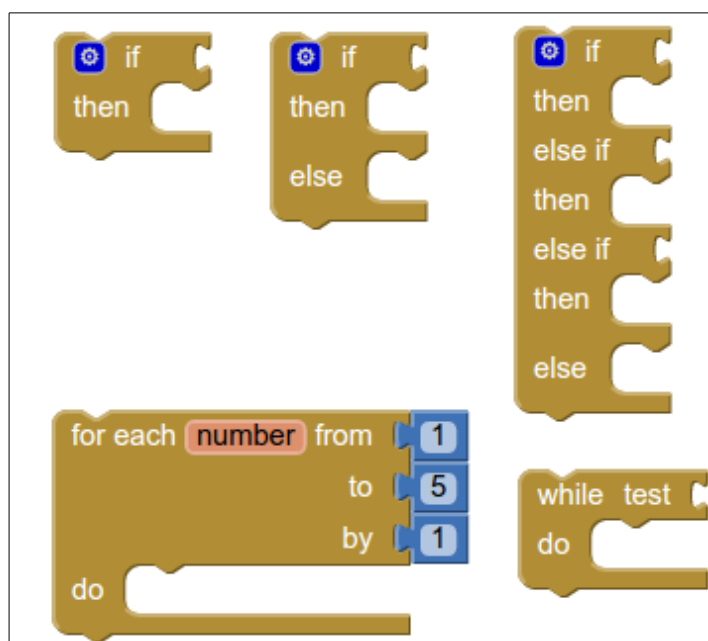
Text blocks: στοιχεία για σύγκριση κειμένων ("<", "=", ">"), συνένωση κειμένων ("join"), εύρεσης και αντικατάστασης κειμένου ("contains text", "replace all text", "starts at text"), εύρεσης μήκους κειμένου ("length").

Color blocks: block για κάθε χρώμα καθώς και block για δημιουργία χρώματος ως RGB σύνθεση ("make color").

Variables: blocks για την αρχικοποίηση μεταβλητών ("initialize global name to"), εκχώρηση τιμής ("set ... to") και ανάγνωση τιμής ("get ...").

Procedures: blocks για τη δημιουργία διαδικασιών που δεν επιστρέφουν ή επιστρέφουν τιμές.

Control blocks



Στις ασκήσεις μας θα χρησιμοποιήσουμε κυρίως τα 3 πάνω blocks (το 2ο και 3ο παράγονται από το 1ο πατώντας το μπλε τετράγωνο).

Αντιπροσωπεύουν τις δομές επιλογής στον προγραμματισμό:

Απλή Επιλογή, Σύνθετη Επιλογή, Πολλαπλή επιλογή.

Απλή Επιλογή If ... then ...: ελέγχεται μια συνθήκη (λογική έκφραση που κατασκευάζεται από Logic και Math blocks με τιμή-αποτέλεσμα true ή false) και αν η τιμή της είναι true εκτελούνται οι εντολές που περιέχονται στην περιοχή του then.

Σύνθετη Επιλογή If ... then ... else ... : ελέγχεται μια συνθήκη και αν η τιμή της είναι true εκτελούνται οι εντολές που περιέχονται στην περιοχή του then, αλλιώς οι εντολές που περιέχονται στην περιοχή του else.

Πολλαπλή Επιλογή If ... then ... elseif ... then else ... : ελέγχεται η πρώτη συνθήκη και αν είναι true εκτελούνται οι εντολές που περιέχονται στην περιοχή του πρώτου then, ενώ αν είναι false εξετάζεται η δεύτερη συνθήκη. Αν εκείνη είναι true εκτελούνται οι εντολές στην περιοχή του δεύτερου then, ενώ αν είναι false εξετάζεται η τρίτη συνθήκη κοκ. Αν δεν ισχύει καμιά από τις ελεγχόμενες συνθήκες εκτελούνται οι εντολές else. Η 4η και 5η δομή ελέγχου είναι δομές επανάληψης και χρησιμοποιούνται για να εκτελέσουμε εντολές προκαθορισμένο αριθμό φορών (for) ή όσο η συνθήκη test είναι true (while). Πιθανότατα δε θα δούμε φέτος δομή επανάληψης.

Logic blocks



Τα 2 πρώτα blocks αντιπροσωπεύουν τις 2 λογικές τιμές-σταθερές true και false

Τα 3ο, 4ο, 5ο αντιπροσωπεύουν τους τελεστές των λεγόμενων λογικών πράξεων not, and, or

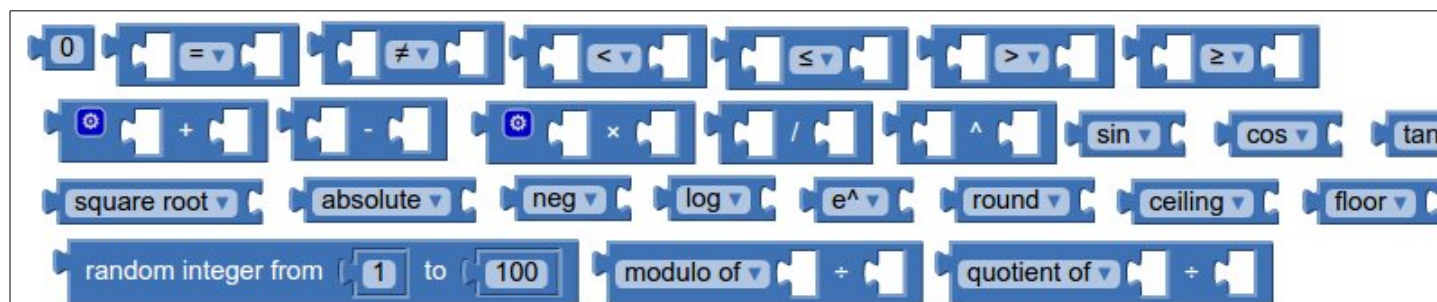
παράδειγμα λογικής πράξης $5 > 4$ and $3 < 2$. Αυτή η πράξη έχει αποτέλεσμα false αφού δεν ισχύουν και οι 2 συγκρίσεις ταυτόχρονα.

παράδειγμα λογικής πράξης $5 > 4$ or $3 < 2$. Αυτή η πράξη έχει αποτέλεσμα true αφού ισχύει έστω μία από τις συγκρίσεις.

παράδειγμα λογικής πράξης not ($5 \neq 4$). Αυτή η πράξη έχει αποτέλεσμα false αφού το not αντιστρέφει το αποτέλεσμα της παρένθεσης $5 \neq 4$ που είναι true, άρα το αποτέλεσμα είναι false.

Το 6ο block αντιπροσωπεύει τον έλεγχο αν 2 πράγματα είναι ή δεν είναι ίσα.

Math Blocks



Το 1ο block αντιπροσωπεύει μια οποιαδήποτε αριθμητική σταθερά (ακέραια ή δεκαδική)

Τα 2ο – 7ο αντιπροσωπεύουν πράξεις σύγκρισης.

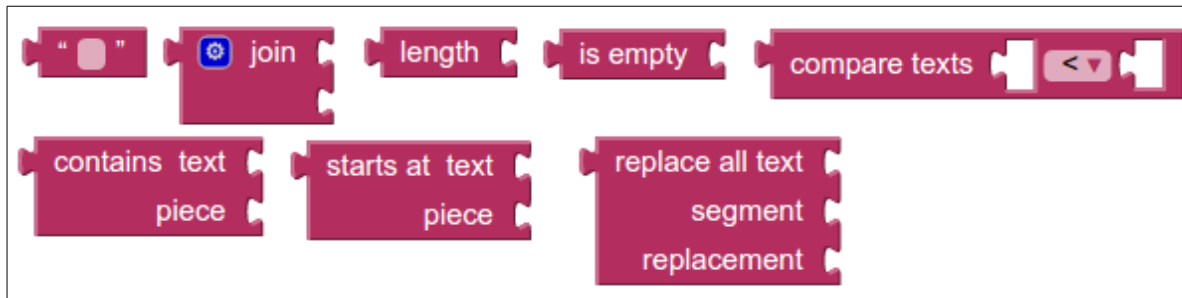
Τα 8ο – 12 αντιπροσωπεύουν αριθμητικές πράξεις. Συγκεκριμένα το 12ο την ύψωση σε δύναμη

Τα 13ο – 15ο αντιπροσωπεύουν συνηθισμένες τριγωνομετρικές συναρτήσεις

Τα 16ο – 23ο αντιπροσωπεύουν συνηθισμένες μαθηματικές συναρτήσεις

Το 24ο γεννήτρια τυχαίων ακεραίων αριθμών. Το 22ο την πράξη υπόλοιπο. Το 23ο την πράξη πηλίκο.

Text Blocks



To 1o block αντιπροσωπεύει μια οποιαδήποτε αλφαριθμητική σταθερά δηλ. σταθερά κείμενου

To 2o block συνενώνει 2 ή περισσότερες αλφαριθμητικές σταθερές σε μία.

To 3o block εξετάζει μια μεταβλητή ή σταθερά που περιέχει κείμενο και δίνει το μήκος του κειμένου

To 4o block εξετάζει μια μεταβλητή κι επιστρέφει true αν δεν περιέχει καθόλου κείμενο ή false αν περιέχει

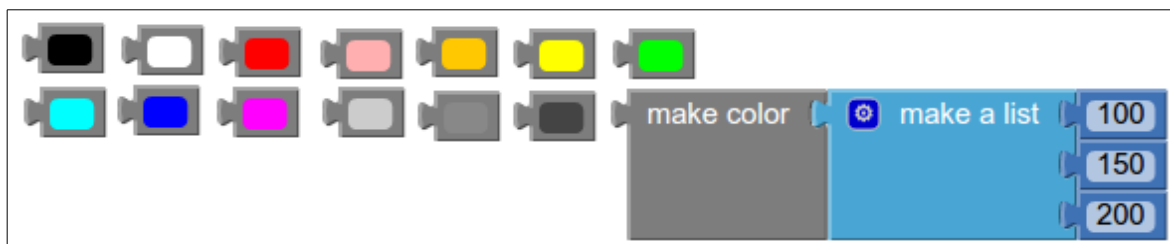
To 5o block συγκρίνει λεξικογραφικά μεταβλητές ή σταθερές κειμένου κι επιστρέφει true ή false

To 6o block εξετάζει αν μια ακολουθία piece κειμένου υπάρχει μέσα στο κείμενο text.

To 7o block επιστρέφει τη θέση μέσα στο text που εμφανίζεται για πρώτη φορά η ακολουθία κειμένου piece.

To 8o block αντικαθιστά τις ακολουθίες segment στο text με το replacement

Color Blocks



Κάθε block αντιπροσωπεύει ένα χρώμα, εκτός το τελευταίο που αντιπροσωπεύει τη δυνατότητα μας να φτιάξουμε οποιοδήποτε χρώμα ως μια σύνθεση-list 3 αποχρώσεων των βασικών χρωμάτων : του κόκκινου R(0-255), του πράσινου G(0-255) και του μπλε B(0-255).

Variable Blocks

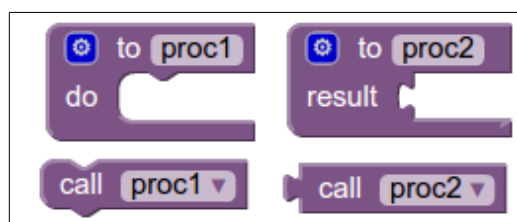


To 1o block αντιπροσωπεύει την δημιουργία και αρχικοποίηση μιας μεταβλητής δηλαδή την απόδοση σε αυτήν μιας αρχικής τιμής κατά την εκκίνηση της εφαρμογής. Ένα τέτοιο block εμφανίζεται αυτόνομο στον block κώδικά μας και δεν συνδέεται με άλλα blocks γεγονότων ή πράξεων κλπ...

To 2o block ουσιαστικά διαβάζει την τιμή που έχει μια μεταβλητή – εδώ η zari1 – στη μνήμη

To 3o block εκχωρεί , αποθηκεύει στη μεταβλητή – εδώ στη zari1 – μια τιμή.

Procedure Blocks



To 1o και 2o block μας δίνουν τη δυνατότητα να φτιάξουμε διαδικασίες που δεν επιστρέφουν ή επιστρέφουν τιμές κατά την κλήση τους αντίστοιχα.

Τα 3o και 4o block καλούν αντίστοιχα αυτές τις διαδικασίες.